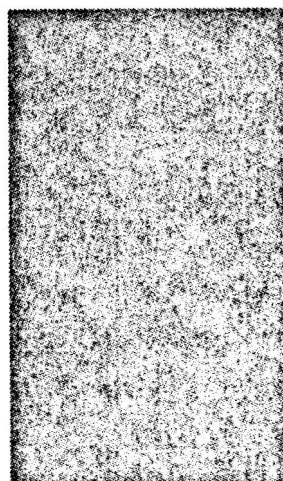


Waldemar Bonventi Jr. ()*

***Lógica “Fuzzy”:
Fundamentos e Aplicabilidade***

(*) Mestre em Física dos Materiais - Modelagem Computacional pela Universidade de São Paulo - USP. Doutorado em Física pela mesma Universidade. Professor nos cursos de Análise de Sistemas e Administração na Universidade de Sorocaba - UNISO.



RESUMO

O artigo apresenta a Lógica Difusa ("Fuzzy Logic") e seu desenvolvimento, surgindo inicialmente como uma das lógicas não-clássicas e, posteriormente, sendo empregada na modelagem e controle de sistemas complexos. Ênfase particular é dada na implementação da metodologia de controle, através de um exemplo prático. É discutida a aplicabilidade do método em função do comportamento do sistema, sendo apontados caminhos para soluções em sistemas estáticos.

ABSTRACT

We introduce here Fuzzy Logic and its development, which has emerged as a non-classic logic and has been employed afterwards in complex systems modeling and control. Special emphasis is given to methodologies of control building, through a practical example. We also discuss the applicability of the method according to the system behavior and point out ways of investigating steady-state systems.

Introdução

A Lógica Nebulosa ou Difusa ("Fuzzy Logic") é basicamente uma lógica multivalente que permite valores intermediários entre afirmações "booleanas" convencionais (sim/não, verdadeiro/falso, branco/preto, alto/baixo, velho/novo, etc.). Podem-se incluir estados como **quase alto** ou **seminovo**. Afirmações estritamente lingüísticas como **bem quente** ou **quase frio** podem ser formuladas matematicamente e processadas por computadores, tornando o relacionamento de dados mais "humanizado". As bases da Lógica Difusa foram, inicialmente, propostas na década de 20 pelo matemático Lukaciewicz (também inventor da notação polonesa reversa). Em 1965, Lotfi A. Zadeh, professor de Ciência da Computação na Universidade da Califórnia, em Berkeley, codificou e expandiu o trabalho de Lukaciewicz, aplicando-o a sistemas de controle,^{1 2} o qual, posteriormente, ganhou espaço no ambiente industrial.

A Lógica Difusa, hoje, é estudada dentro das áreas de Inteligência Artificial que, juntamente com as Redes Neurais e os Sistemas Especialistas, formam uma nova abordagem da descrição do comportamento dos sistemas complexos, e que rapidamente se tornou bem sucedida no desenvolvimento de sistemas de controle sofisticados. Também auxiliou na implementação de prescrições de controle muito complexas, de modo que a modelagem de sistemas a serem controlados simplificou-se extremamente, assim como sua manutenção. O "hardware" que suporta essas implementações também se simplificou acentuadamente, acolhendo um conjunto de regras de controle muito mais simples. O suporte fornecido estendeu-se também a sistemas especialistas em geral e de apoio à decisão, com a decomposição de regras complexas de controle e decisão em unidades constituídas por sentenças lógicas simples.

Na realidade, a Lógica Difusa não é a melhor implementação para resolver todas as situações. Gerentes de projeto, arquitetos de sistemas e engenheiros de "software" continuam a se perguntar: "quais tipos de projetos e implementações podem-se beneficiar desta tecnologia? Qual a melhor metodologia para desenvolver e testar sistemas que a empregam?"

¹ ZADEH, L.A. Fuzzy Sets. *Information Control*, vol.8, p.338-353, 1965.

² ZADEH, L.A., Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics*, vol.3, p.28-44, 1973.

A Lógica Difusa tem sido um bom método para representar processos analógicos em um computador digital. A razão inicial é puramente intrínseca ao ambiente: um processo analógico refere-se a fenômenos (ex.: sinais elétricos) de comportamento contínuo no tempo. O computador digital, por sua natureza, funciona com sinais elétricos discretos no tempo, e em cada intervalo (frações de segundo) é definida a existência ("bit" 1) ou não ("bit" 0) do sinal. Portanto, todos os processos que ocorrem no computador digital são de natureza discreta no tempo. Frequentemente, um processo relacionado a um fenômeno contínuo nem sempre é facilmente dividido em segmentos discretos, tornando difícil a modelagem lógico-matemática do sistema a ser controlado.

Um exemplo de sistema dinâmico é o sistema de freios ABS (anti-locking braking system) de um automóvel. As regras de controle de um sistema, como esse, incluem a velocidade do automóvel, a pressão exercida na frenagem, a temperatura dos materiais envolvidos (que altera o coeficiente de atrito), o intervalo de aplicação dos freios. Todas essas variáveis são contínuas no tempo, e os valores que as mesmas assumem ao longo da operação do sistema necessitam ser abrangidos e corretamente interpretados pelo desenvolvedor.

A variável temperatura pode apresentar uma gama de estados descritivos: *frio, morno, quente, gelado*, etc. A região de transição entre um estado e outro não é precisamente definida. A noção de "quente" (e qual o valor em °C respectivo) difere, na interpretação de vários especialistas, de acordo com diferentes sistemas. Essa subjetividade apresenta profundas implicações na modelagem de sistemas contínuos, e o cerne de seu poder está na flexibilidade da Lógica Difusa. Com ela, as regras de controle são escritas em termos de idéias imprecisas (difusas) que representam o estado de cada variável. Voltando ao exemplo anterior, uma regra difusa sobre a ação de frenagem pode ser estabelecida na forma:

"Se a temperatura do freio for morna e a velocidade não muito rápida, então a pressão do freio será ligeiramente diminuída". (1)

O equivalente do controlador lógico programável corresponde a:

"Se a temperatura do freio for maior que 280°C e a velocidade menor que 45 km/h, então a pressão do freio será 190 kPa". (2)

A regra difusa é mais natural à nossa linguagem e funciona, nesse exemplo, para uma ampla variedade de veículos, temperaturas, velocidades. O valor que será atribuído às variáveis **morna** ou **muito rápido** depende da característica do sistema (veículo + sistema de freio) que se está modelando. Se especificarmos que **frio** corresponde às temperaturas de 15 a 45°C, **morno** de 45°C a 75°C e **quente** acima de 75°C, estaremos aplicando o que denominamos conjuntos difusos ("fuzzy sets").

Freqüentemente, uma afirmação lógica difusa como a (1) pode representar uma equação matemática que descreva o modelo de controle do sistema, às vezes, complicada demais para ser implementada em um microprocessador que integre o sistema eletrônico do automóvel. As regras difusas, combinadas com os conjuntos difusos, criam "condições de contorno" geradoras da função de controle, mesmo quando o comportamento matemático do sistema não for completamente conhecido.

Em que casos se torna apropriado o emprego de lógica difusa?

1. quando uma ou mais variáveis de controle forem contínuas;
2. quando o modelo matemático do processo não existir;
3. quando o sistema é muito complexo para ser descrito;
4. ou muito complexo para ser calculado em tempo real;
5. ou envolve o uso de muita memória eletrônica;
6. quando o ruído eletrônico do ambiente externo sobre o sinal é tanto que necessite o uso de sensores baratos e um tanto insensíveis e/ou microprocessadores simples e baratos;
7. quando houver um especialista que possa especificar as regras de comportamento do sistema e os conjuntos difusos que representem as características de cada variável.

Em que casos **não** se torna apropriado o emprego de lógica difusa?³

1. quando as teorias de controle convencionais fornecem um resultado satisfatório acerca do sistema;
2. quando existe um modelo matemático adequado e solúvel (mesmo com o emprego de cálculo numérico);
3. quando o problema for insolúvel...

³ BAUER, Peter; NOUAK, Stephan; WINKLER, Roman; A brief course in Fuzzy Logic and Fuzzy Control, Version: 1.2. Internet homepage at <ftp://ftp.flll.unilinz.ac.at/pub/info>, December 4, 1996.

O modelo de controle difuso

A idéia básica em controle difuso é modelar as ações a partir de um conhecimento especialista, retratado na construção dos conjuntos difusos, como acima. A modelagem deixa de ser lógico-matemática para ser lógico-comportamental do sistema.

A estrutura de um processo controlado por lógica difusa está representada a seguir.

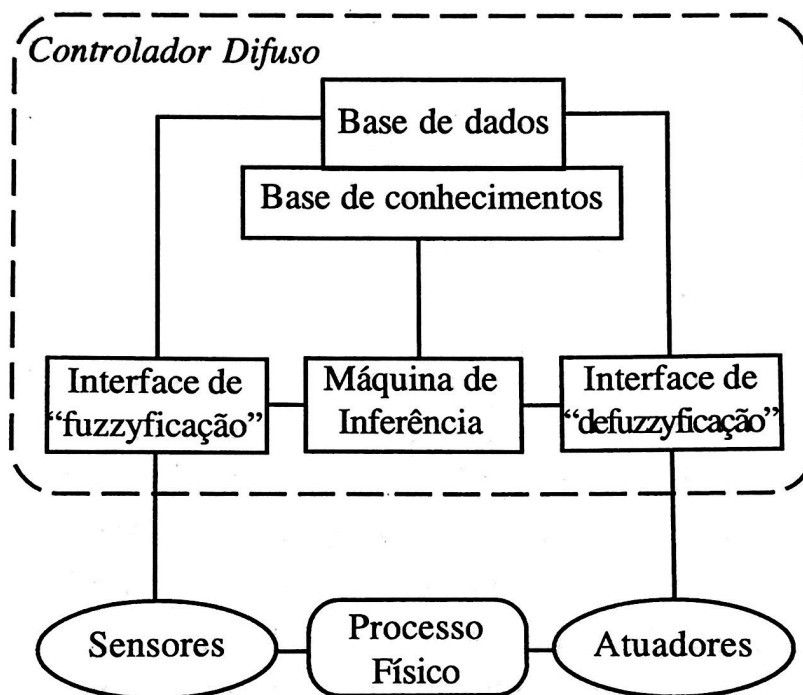


Figura 1: esquema modular de um controlador difuso.

A interface de “*fuzzyficação*” toma os valores das variáveis de entrada, normalizando-os para os valores de pertinência do *universo de discurso* (conjunto difuso), transformando-os em *instâncias de variáveis lingüísticas*. A *base de conhecimento* engloba uma série de regras (afirmações lógicas) que caracterizam a estratégia de controle e suas metas. A *base de dados* armazena as definições dos universos de discurso, as partições difusas dos espaços de entrada e saída e as definições das funções de pertinência. A *máquina* (ou procedimento) *de inferência* processa os dados difusos, aplicando-os às regras lógicas para obter as ações de controle sobre o sistema. A *interface de “defuzzyficação”* transforma a ação obtida, expressa em linguagem natural, em valor numérico no espaço analógico das variáveis de saída.

Os componentes do modelo de controle convencional e do difuso são muito parecidos, apenas pelo fato de que no difuso existem os "fuzzyficadores", que convertem as entradas analógicas ou digitais em representação difusa, e os "defuzzyficadores" que fazem a conversão inversa [fig. 1]. Em seguida, os valores de entrada já "fuzzyficados" são submetidos às regras do "banco de conhecimentos" ("knowledge base") sobre ações a serem executadas sobre o sistema. A execução dessas regras produz um novo conjunto difuso representando a saída ou ação do sistema de controle. Em sistemas físicos, os valores de saída destes são usados como entrada do controlador que, por sua vez, corrige os sinais de entrada sobre o sistema físico, ajustando seu estado.

Em contrapartida, um controlador convencional baseia-se em um rigoroso modelo matemático, descrevendo o comportamento físico do sistema, em geral, um conjunto de equações diferenciais acopladas e, se possível, linearizadas. A solução destas produz um modelo funcional do sistema, cujo objetivo é obter seus valores de equilíbrio (os quais variam com o tempo, a cada novo estado do sistema). Na prática, a modelagem matemática é trabalhosa e requer aproximações na impossibilidade de sua solução exata.

Controladores baseados em lógica difusa são mais fáceis de se prototipar e implementar, mais simples de se verificar e descrever, e podem ser mantidos e estendidos com grande acurácia em pouco tempo, devido à sua confiabilidade na descrição das regras sobre o banco de conhecimentos, criando as condições propícias que Lotfi Zadeh, o pai da Lógica Difusa, denominou "máquina com alto quociente de inteligência".

Metodologia de implementação.⁴

Modelos difusos tendem a seguir sempre o mesmo ciclo de desenvolvimento [figura 1], iniciando com o projeto conceitual em papel, até os testes finais de modelagem e simulação auxiliados por computador. Tanto a compreensão do comportamento dinâmico do sistema quanto a sua descrição em termos de teoria de controle e posterior conversão ao modelo difuso são partes essenciais da modelagem. Quatro das etapas de modelagem dos módulos da figura 1 são importantes:

⁴ COX, Earl. Fuzzy Fundamentals. *IEEE Spectrum*, october 1992, p. 58-61.

1. *definir o modelo funcional e características operacionais do sistema.* Mesmo não conhecendo o modelo matemático deste, deve ser descrito em termos de entrada-processo-saída. O fluxo de dados no sistema deve ser identificado pelo projetista, bem como as transformações que ocorrem nele.

2. *definir as “superfícies de controle”.* Cada variável de controle e sua solução no modelo difuso são decompostas em regiões difusas. Estas recebem rótulos únicos, onde o conjunto difuso representa semanticamente o conceito associado aos rótulos criados [gráfico 2, adiante]. O número de rótulos a uma variável deve ser um ímpar entre cinco e nove, e cada um deles deve se sobrepor um pouco ao vizinho, permitindo uma superfície de controle mais suave (cerca de 10 a 50% de superposição). Devido à definição de conjunto difuso, a soma dos valores das funções superpostas em um determinado valor da abscissa não deve ser maior do que um. Finalmente, a densidade do conjunto deve ser maior em torno do ponto de controle ótimo.

3. *definir o comportamento das superfícies de controle.* Escrever as regras de controle em forma de sentenças lógicas: “Se *proposição 1* e/ou *proposição 2*, então *proposição 3*”, onde cada proposição é do tipo “ x é Y ”, “ x não é Y ”, x sendo variável escalar (grandeza física) e Y o conjunto difuso associado à variável.

O número de regras de controle necessárias ao sistema é diretamente proporcional ao número de variáveis de controle. Se o freio ABS exigir duas variáveis de entrada, a velocidade do veículo e a temperatura do disco, e cada uma delas estiver dividida em cinco regiões difusas, há 25 possíveis combinações, dispostas como elementos de uma matriz (a matriz contendo as regras de controle). Porém, na situação real, talvez nem todas as 25 regras sejam aplicáveis, pois algumas delas podem representar um estado que nunca é atingido pelo sistema. Essas regras constituem a base de conhecimento, a remoção de alguma delas significa subtrair “conhecimento” do sistema. Se o sistema precisar ser modificado mais tarde, as regras existentes se tornam todas importantes, e até novas regras deverão ser acrescentadas.

4. *selecionar o método de “defuzzyficação”.* Conversão da ação de controle obtida em um conjunto difuso, como consequência das regras de 3. Há várias maneiras de se converter a solução gráfica do conjunto difuso em um valor analógico no espaço métrico da variável de saída, sendo as mais comuns o

máximo composto e o **centro de massa da figura**. Enquanto este obtém um resultado que é contribuição de todas as regras aplicadas (alterando-se suavemente sobre a superfície de controle), o máximo composto produz um resultado mais sensível à regra de maior peso no espaço de soluções.

Tendo construído o modelo de controle difuso, inicia-se o processo de simulação do mesmo. Em paralelo, pode-se compará-lo com modelos conhecidos e previamente testados, para validação dos resultados. Caso não satisfaçam, devem-se alterar algumas das regras da base de conhecimento ou alterar o domínio das regiões difusas nas variáveis do sistema afetadas.

Formalização da lógica

Apresentaremos aqui apenas as idéias básicas necessárias para a implementação e discussão da Lógica Difusa em sistemas de controle; no entanto, existe uma complexa base formal de construção da lógica e que justifica seu emprego.

Construção dos conjuntos difusos representando variáveis físicas.

Na teoria clássica dos conjuntos, um elemento pertence a um (ou mais) conjunto(s) ou não. Logo, sendo um elemento particular $x \in U$ (universo de elementos), em respeito a um conjunto $A \subseteq U$, apresenta um grau de pertinência $\mu_A(x)$ dado por:

$$\mu_A(x) = \begin{cases} 1 & \text{se } x \in A \\ 0 & \text{se } x \notin A \end{cases} \quad (3)$$

A função $\mu_A(x): U \rightarrow \{0,1\}$ é chamada de função característica na Teoria Clássica de Conjuntos. Se considerarmos $\sim A$ o conjunto complemento de A , e que $x \in A$, então $x \notin \sim A$, levando à lei aristotélica da Não-contradição: $A \cap \sim A = \emptyset$

Zadeh propôs uma caracterização mais ampla, sugerindo que alguns elementos podem ser mais membros de um conjunto do que de outros. O valor da função característica (ou *função pertinência*) pode assumir qualquer valor entre 0 e 1 (... 0-100% pertence a...).

Como exemplo, podemos construir funções de pertinência para a temperatura do freio, de acordo com a representação gráfica seguinte:

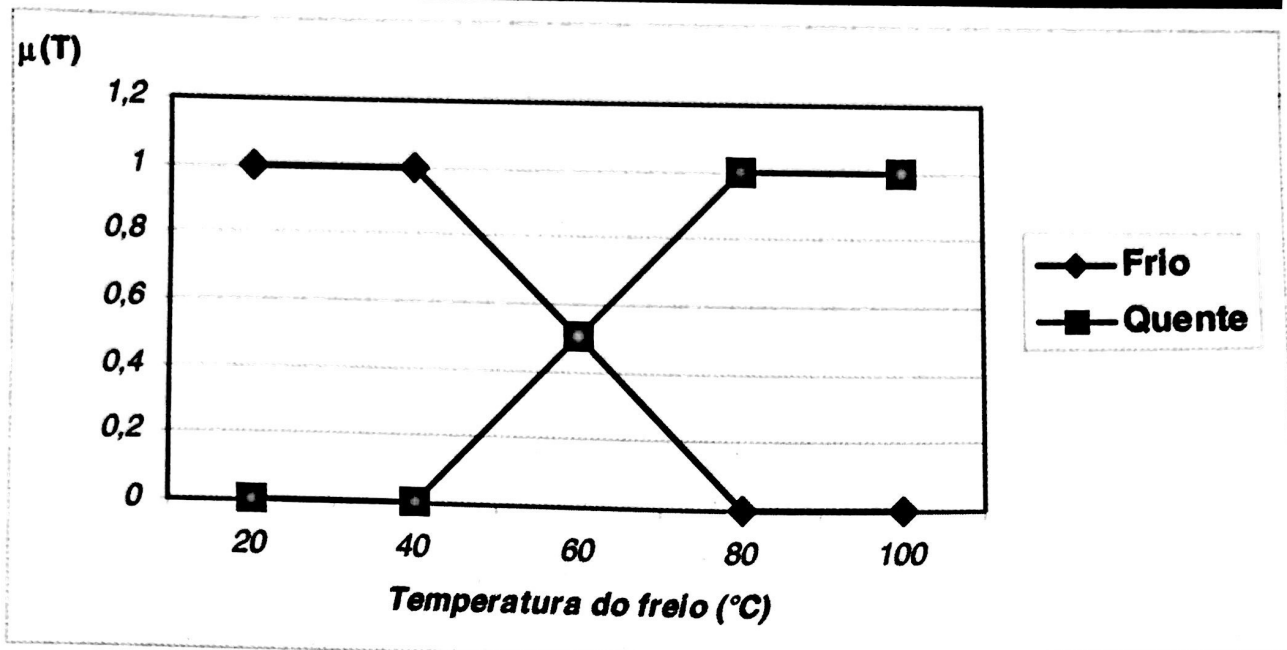


Gráfico 1: Conjuntos difusos "frio" e "quente" e grau de pertinência em função da temperatura.

Para expressar conceitos e medidas, é muito comum o uso de linguagem natural que, no caso da temperatura dado acima, pode-se refiná-la com o intuito de um controle mais acurado do sistema:

Temperatura = { muito frio, frio, morno, quente, fervendo }

O conjunto difuso respectivo poderia ser, como exemplo:

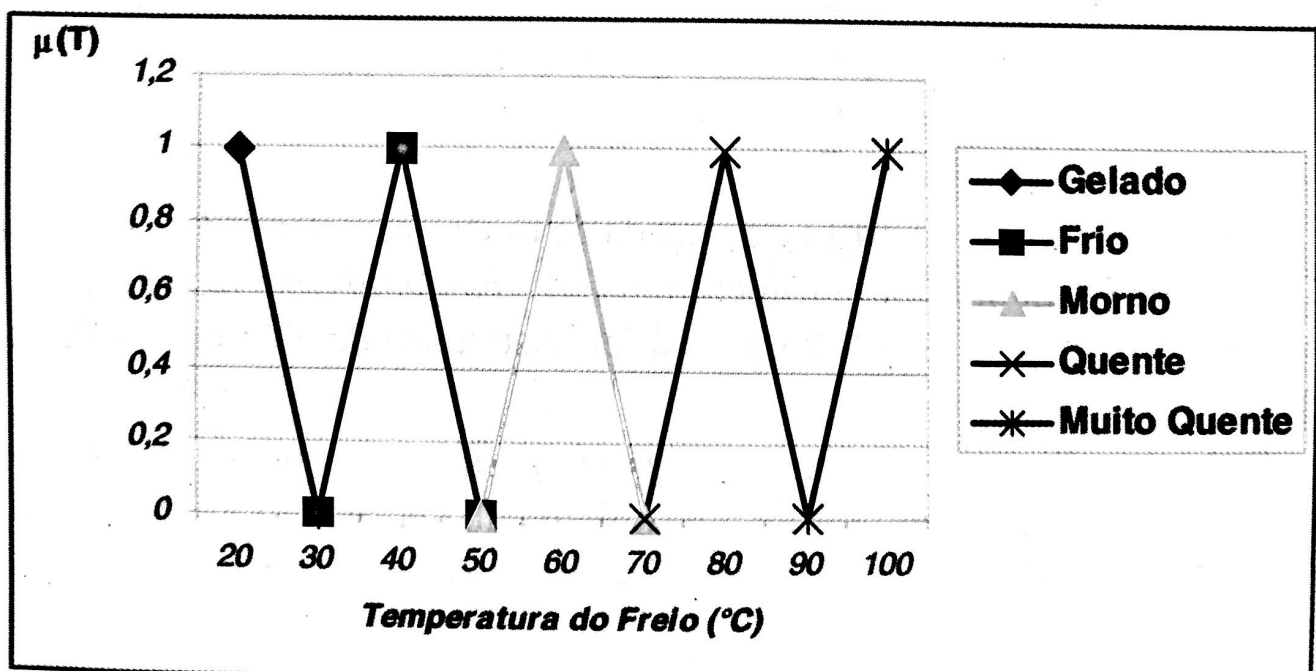


Gráfico 2: Conjuntos difusos sobre a função da temperatura, com maior refinamento.

com os limites de temperatura para cada função $\mu_{\text{conceito lingüístico}}(T)$ determinados pelo projetista do sistema. Note-se que os mesmos valores de T limites para cada uma das funções $\mu(T)$ seriam alterados, se o sistema fosse uma caldeira de fundição, em vez de um motor de automóvel. Para seres humanos, certamente que $T=20^{\circ}\text{C}$ não corresponde ao conceito *gelado*, como estabelecido no gráfico anterior.

A construção dos conjuntos difusos exemplificados acima, com as funções de pertinência determinando as regiões de temperatura para cada conceito lingüístico, implica que, para um determinado valor T_p (ou a variável que se quer representar) a soma das funções $\mu(T_p)$ não deve exceder 1. Um outro exemplo: se o universo de discurso foi construído com as variáveis lingüísticas **alto** e **baixo**, a lógica booleana afirma que Fulano é alto (portanto não-baixo), a lógica difusa afirma que Fulano pode ser considerado 60% alto e 40% baixo (para uma altura de, digamos, 1,70m), mas **jamais** a soma das funções lingüísticas relativas a 1,70m deve ultrapassar 100%. A consequência direta desta implicação é que as funções μ_{alto} e μ_{baixo} estão superpostas no valor 1,70m. Em nosso exemplo anterior de temperatura, não há superposição entre as funções μ , o que se torna perigoso no caso $T=90^{\circ}\text{C}$. Ali não há nenhuma representação difusa (todas as $\mu=0$), de modo que o controlador se perderia, pois não consegue aplicar nenhuma regra de inferência, usando uma variável lingüística. Uma superposição segura, que tornasse o controlador suavizado, seria como a da figura seguinte:

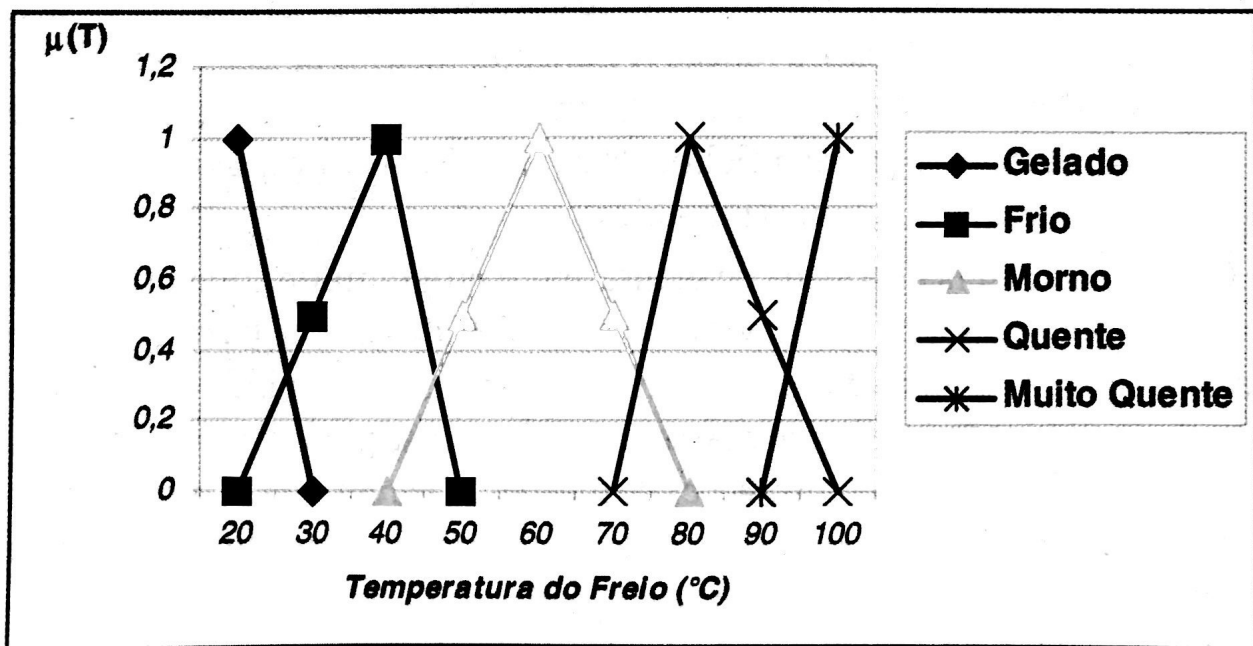


Gráfico 3: Conjuntos difusos apresentando superposição.

Note que as superposições entre as funções de pertinência não são regulares (os gráficos são assimétricos), mas isto depende de como o comportamento e a caracterização do sistema são avaliados pelo projetista-especialista.

Formas das funções de pertinência

Observa-se nos gráficos que as funções de pertinência que representam cada estado do sistema (ex. morno) são triangulares, isto é,

$$\mu_{\text{MORNO}}(T) = \begin{cases} 0 & \text{se } T < 40^{\circ}\text{C} \\ 0,05.T - 2 & \text{se } 40 \leq T < 60^{\circ}\text{C} \\ -0,05.T + 4 & \text{se } 60 \leq T < 80^{\circ}\text{C} \\ 0 & \text{se } T > 80^{\circ}\text{C} \end{cases} \quad (4)$$

e assim por diante para cada estado.

Dependendo do conceito que a variável lingüística venha a transmitir, o formato da função $\mu(x)$ pode ser simétrico ou não ou, ainda, crescente/decrescente. Por exemplo, a função $\mu_{\text{quente}}(T)$ representada no [gráfico 1] é composta de duas funções constantes (zero para $T < 40^{\circ}\text{C}$, um para $T > 80^{\circ}\text{C}$) e uma reta para T entre 40 e 80°C . Ela também poderia ser construída simplesmente a partir de uma reta crescente, do ponto $(20; 0)$ ao ponto $(100; 1)$. Ou, uma função *degrau* (cujo gráfico é uma “escada”) ou, ainda, a função *sigmóide*. Funções simétricas, além da triangular, são a *trapezoidal*, a *gaussiana* (ou normal), a *uniforme* (que corresponde à definição clássica ou booleana de pertinência), a função Π (junção simétrica de duas sigmóides: crescente e decrescente), a função *beta*, ou quaisquer outras, regulares ou não, que procurem representar fielmente a relação entre a variável lingüística difusa e a variável analógica numérica.

Quando houver necessidade de se inserir um programa de computador em um microprocessador programável (como as EPROMs), onde há restrições de memória e os programas devem ser os menores possível - e ainda escritos em linguagens de baixo nível, como o Assembly - devem-se preferir representações matemáticas lineares (ou composições de funções lineares). Assim, uma função de pertinência descrita por uma função gaussiana pode ser substituída por uma triangular ou trapezoidal, com o formato devidamente ajustado, sem perda significativa de precisão.

Regras de inferência - ações do controlador difuso

1) Uma regra de controle difuso do tipo se (x é A) então (y é B) ou, resumidamente $A \rightarrow B$, é inicialmente transformada em uma relação difusa $R_{A \rightarrow B}$, obedecendo ao formalismo dos operadores difusos de implicação⁵:

$$\mu_{A \rightarrow B}(x, y) = \min(\neg \mu_A(x), \mu_B(y)); \quad x \in X, y \in Y \quad (5)$$

2) Complemento difuso:

$$\mu_A^c(x) = 1 - \mu_A(x) \quad \forall x \in U \quad (6)$$

3) Operador e :

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)); \quad \forall x \in U \quad (7)$$

4) Operador ou :

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)); \quad \forall x \in U \quad (8)$$

5) Inclusão:

$$A \subset B \Leftrightarrow \mu_A(x) \leq \mu_B(x); \quad \forall x \in U \quad (9)$$

6) Relação entre conjuntos A e B :

$$R(A, B) = \{(x, y) \in A \times B / xRy\} \quad (10)$$

onde a função de pertinência do produto $A \times B$ é dada por

$$\mu_{A \times B}(x, y) = \max(\mu_A(x), \mu_B(y)); \quad \forall x, y \in U \quad (11)$$

R é caracterizado pela função de pertinência $\mu_R(x, y)$ que representa o "grau de relacionamento" entre os elementos x e y . Exemplo de graus de relacionamento entre duas pessoas e seus respectivos cães pode ser representado matricialmente por

⁵ LEE, C.C., Fuzzy Logic in control systems: Fuzzy Logic, Part I and II. **IEEE Transactions on Systems, Man and Cybernetics**, vol.20, part 2, p. 404-435, mar/abr 1990.

$$A = \{\text{Antônio, Dique}\}, \quad B = \{\text{Joana, Filó}\}$$

$$R = \begin{matrix} & \begin{matrix} \text{Joana} & \text{Filó} \end{matrix} \\ \begin{matrix} \text{Antônio} \\ \text{Dique} \end{matrix} & \begin{bmatrix} 0.8 & 0.3 \\ 0.2 & 0.6 \end{bmatrix} \end{matrix}$$

“Defuzzyficação”: transformação de uma regra de inferência, cujo resultado pertence a um conjunto difuso, em um valor analógico da variável correspondente.

Os estudos realizados com as aplicações difusas sugerem dois principais métodos para obter o resultado mensurável sobre a(s) variável(is) de saída:

- 1) *média dos máximos* das funções de pertinência envolvidas na obtenção do resultado;
- 2) *mediana (ou centro de massa)* da figura geométrica obtida pelo truncamento das funções de pertinência em suas ordenadas.

A demonstração prática dessas prescrições está nas aplicações tomadas como exemplo, na seção a seguir.

Aplicações:

Controle de sistemas dinâmicos

Turbina a vapor⁶

Um tipo de sistema de comportamento não linear e de modelagem complexa é o termodinâmico. O primeiro exemplo é uma turbina a vapor, em que são envolvidas as variáveis de entrada pressão e temperatura, e a variável de saída é a velocidade de acionamento da válvula da turbina.

Interface de “fuzzyficação”:

Cada variável é decomposta em regiões de controle difusas, de acordo com a prescrição de um especialista. A modelagem inicial poderia ser como a descrita nos gráficos seguintes:

⁶ Cox, Earl; *The Fuzzy Systems Handbook*, Boston, Academic Press Inc., 1994

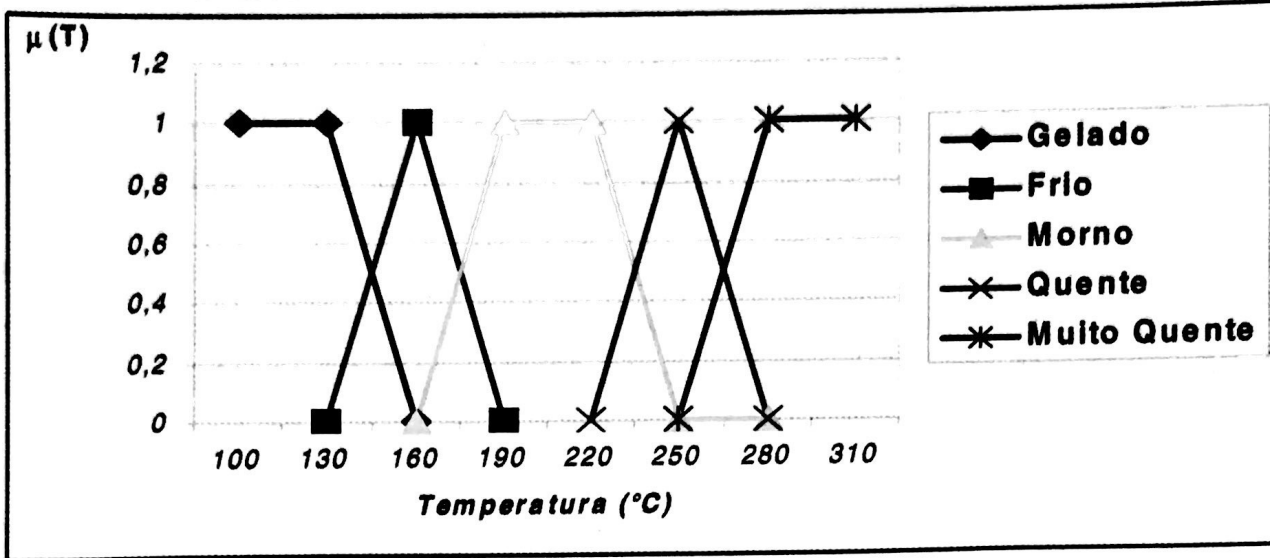


Gráfico 4: Conjuntos difusos em função da temperatura, na turbina a vapor.

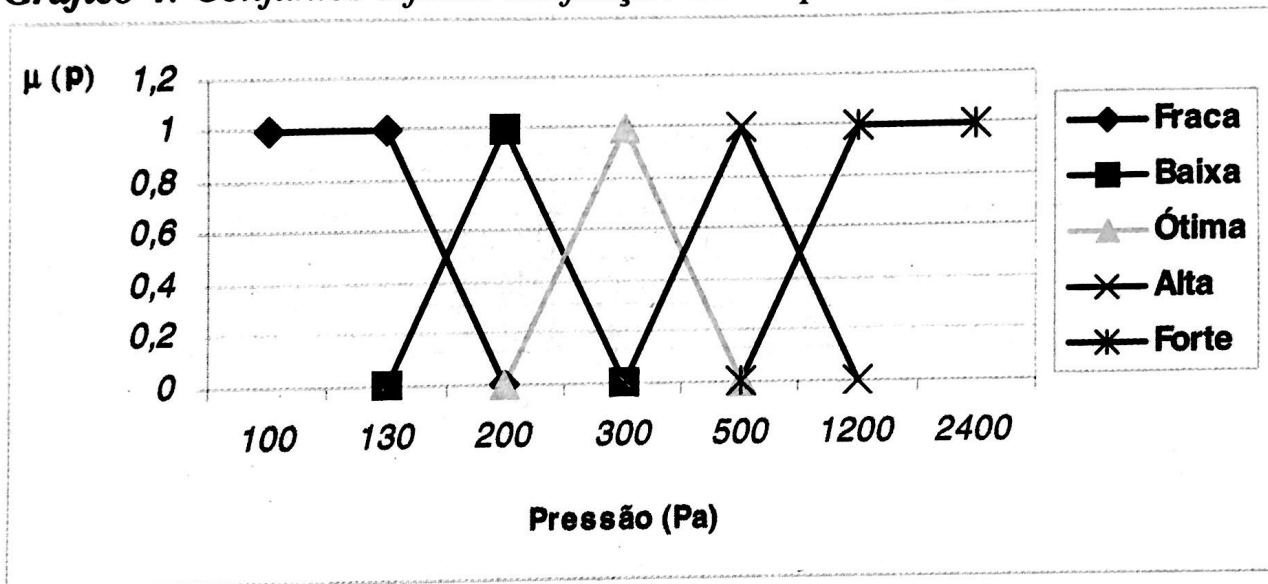


Gráfico 5: Conjuntos difusos em função da pressão do vapor.

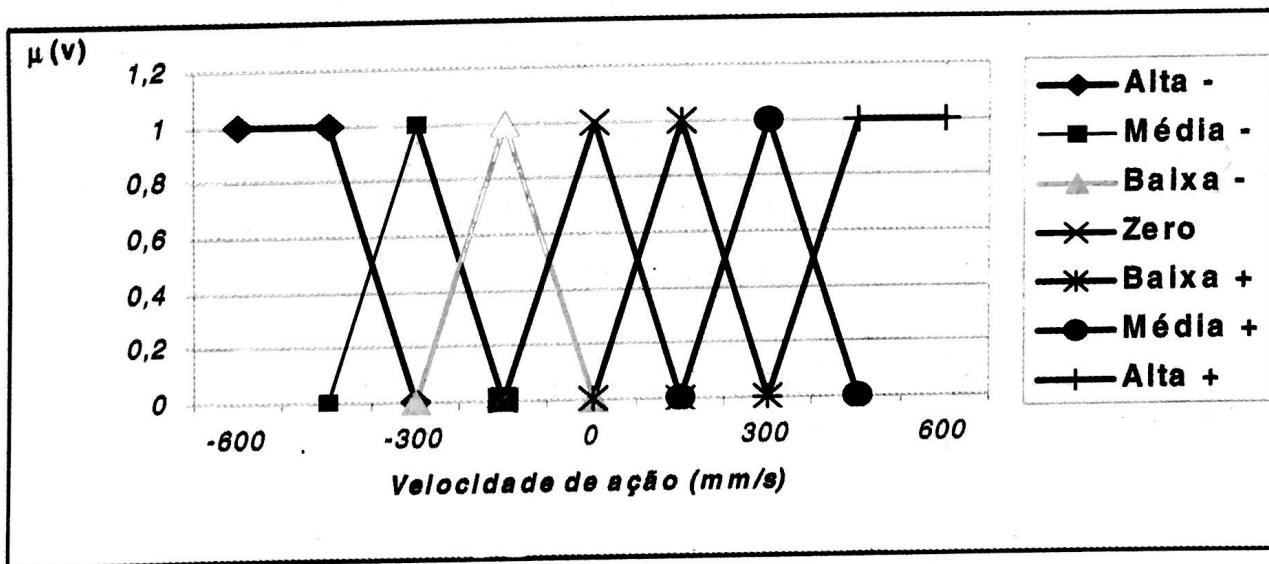


Gráfico 6: Conjuntos difusos em função da velocidade da turbina a vapor.

Regras de inferência:

Sentenças lógicas descrevendo a ação desempenhada pelo sistema de controle, pela combinação dos estados difusos definidos anteriormente.

1. Se a temperatura for **fria** e a pressão for **fraca**, então a velocidade é **média +**
2. Se a temperatura for **fria** e a pressão for **baixa**, então a velocidade é **baixa +**
3. Se a temperatura for **fria** e a pressão for **ótima**, então a velocidade é **zero**
4. ...

E assim por diante. O conjunto de regras pode ser representada pela matriz seguinte:

<i>Temperatura</i> <i>Pressão</i>	Gelado	Frio	Morno	Quente	Muito quente
<i>Velocidade</i> ↘					
Fraca		Média +	Baixa +	Zero	Baixa -
Baixa		Baixa +	Zero	Baixa -	Média -
Ótima	Baixa +	Zero	Baixa -	Média -	Alta -
Alta	Zero	Baixa -	Média -	Alta -	
Forte	Baixa -	Média -	Alta -		

onde notamos que algumas regras não estão definidas porque **seguramente** o especialista-projetista considerou estados que o sistema **jamais** atingirá.

“Defuzzyficação - a resposta do controlador:

Esta etapa é a que deve transformar a(s) solução(ões) difusa(s) obtida(s) pelas regras de inferência (e seus valores de pertinência) em valores numéricos na(s) variável(eis) de saída - que é única neste exemplo.

Considere-se que, em um determinado instante, os valores das variáveis do sistema são $T = 240^{\circ}\text{C}$ e $p = 220 \text{ Pa}$. No [gráfico x], os valores de pertinência são ambíguos, pois abrangem duas funções:

$$T = 240^{\circ}\text{C} \Rightarrow \mu_{\text{MORNO}} = 0,35 \quad \text{e} \quad \mu_{\text{QUENTE}} = 0,65$$

$$p = 500 \text{ Pa} \Rightarrow \mu_{\text{ALTA}} = 1,00$$

Devem-se aplicar as regras de inferência para todas as combinações ocorridas:

Se a temperatura for morna e a pressão for alta, então a velocidade é média -
Se a temperatura for quente e a pressão for alta, então a velocidade é alta -

Como as regras apresentam o operador e, devemos aplicá-lo segundo a Lógica Difusa:

$$\mu_{MÉDIA-}(x) = \mu_{MORNA \cap ALTA}(x) = \min(\mu_{MORNA}(x), \mu_{ALTA}(x)) = \min(0,35; 1,00) = 0,35 \quad (12)$$

$$\mu_{ALTA-}(x) = \mu_{QUENTE \cap ALTA}(x) = \min(\mu_{QUENTE}(x), \mu_{ALTA}(x)) = \min(0,65; 1,00) = 0,65 \quad (13)$$

Mas, qual das regras aponta para a solução? A velocidade será média- e alta-? Qual será o resultado em mm/s? Sendo a solução uma combinação das regras obtidas, truncamos as funções de pertinência obtidas nos seus valores e aplicamos a união dos conjuntos difusos truncados média- e alta-, conforme gráfico seguinte:

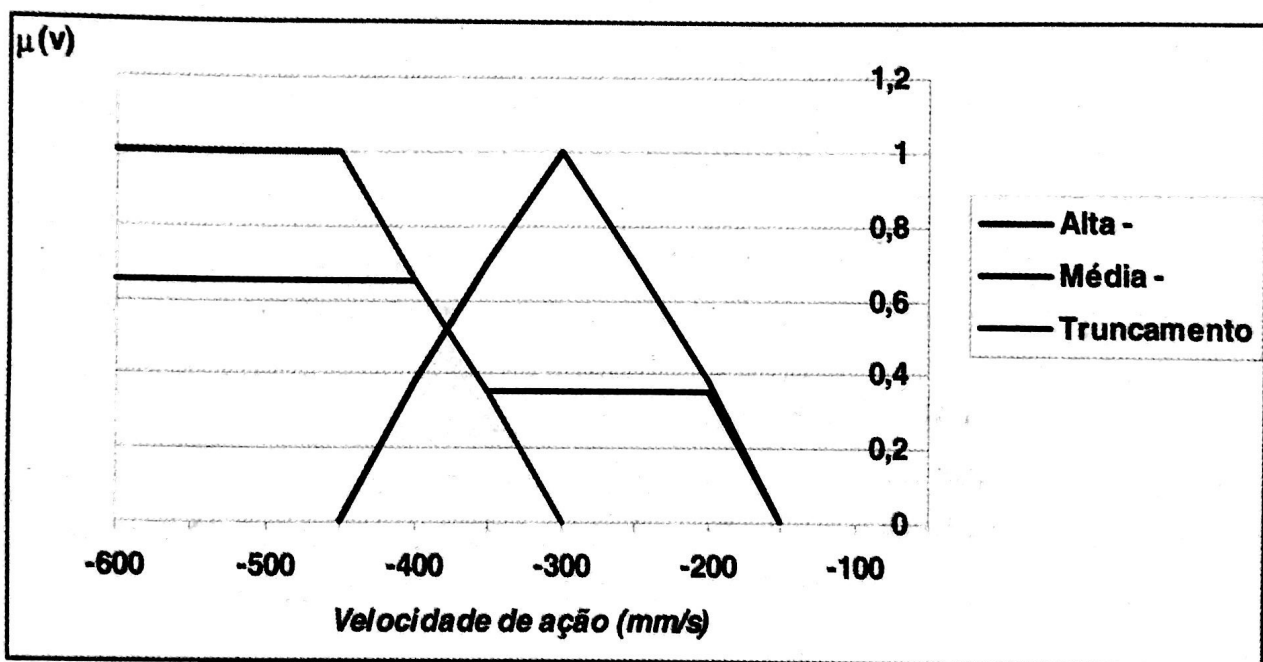


Gráfico 7: União dos conjuntos difusos obtidos pela aplicação de regras simultâneas.

O conjunto união é representado pela área sob as linhas de truncamento 1 e 2, no gráfico 7 acima. Para “defuzzyficar” esse conjunto, isto é, obter o valor da velocidade em mm/s, são usuais dois métodos matemáticos:

- 1) **Cálculo do centro de gravidade** da área de truncamento, que corresponde a uma velocidade de -430 mm/s (projeção no eixo correspondente).

- 2) **Média dos máximos** de cada função difusa obtida que, em nosso caso, resulta $((-500)+(-300))/2 = -400$ mm/s

Embora os dois métodos geralmente divirjam, ao serem aplicados, o sistema é dinâmico. A ação efetuada pela válvula, ao atingir a velocidade determinada acima pelo controlador, levará o sistema a outro ponto de equilíbrio, provocando a alteração dos valores de temperatura e pressão. Esses novos valores serão aplicados ao controlador difuso, passando novamente por todo o processo descrito acima e chegando a um novo valor da velocidade de ação da válvula. O controle difuso é acionado ciclicamente, atuando no sistema a intervalos regulares de tempo.

Sistemas Lineares

A aplicação de regras lógicas difusas também tem sido validada em sistemas unidimensionais simples, que podem ser modelados matematicamente, mas cuja função descritiva é muito complicada para ser empregada em microprocessadores programáveis. A facilidade de implementação das regras difusas, bem como as etapas de “fuzzyficação” e “defuzzyficação” se tornam factíveis em ambientes onde a simplicidade seja a norma. Os resultados obtidos com o controlador difuso, comparados com o modelo de controle tradicional (PID, por exemplo), são bem próximos, o que se obtém com uma calibração mais meticulosa das funções de pertinência e das regras de inferência.

Sistemas em que o controlador difuso pode ser aplicado com sucesso encontram-se citados nos trabalhos de Isotani, Albuquerque, Muratore e Bonventi⁷ e também Isotani, Albuquerque, Muratore e Brasil Filho⁸.

Controle de sistemas estáticos

O controle difuso de um sistema estático, cujo ponto de equilíbrio seja necessário alcançar, e que é desconhecido e ainda não modelável matematicamen-

⁷ ISOTANI, Sadao; ALBUQUERQUE, Antonio R.P.L.; MURATORE, Marcio; BONVENTI JR., Waldemar; A 1-D Pattern Recognition System Based on Fuzzy Logic. Submetido para publicação.

⁸ ISOTANI, Sadao; ALBUQUERQUE, Antonio R.P.L.; MURATORE, Marcio; BRASIL FILHO, Nilson. A recursive spline-based algorithm for sensor calibration design. Submetido para publicação.

te, aplica-se da mesma forma descrita na seção anterior, com todas as etapas de projeto, partes do controlador difuso e metodologias.

O problema com que se depara é o fato de ser um sistema estático, isto é, uma vez obtido(s) o(s) valor(es) da(s) variável(is) de saída e encontrado um ponto de equilíbrio ou acomodação, o sistema não será levado a um novo ponto de equilíbrio devido às alterações destas variáveis. O sistema ficará "morto".

Para solucionar o impasse, alguns métodos são propostos, adicionando novas metodologias e módulos ao controlador difuso. Um deles baseia-se no modelo de decisões de Bellman e Zadeh⁹, sobre o qual foi derivado um procedimento de otimização aplicado a sistemas estacionários¹⁰.

Uma outra solução a ser implementada é causar uma pequena perturbação no sistema estacionário, cada vez que o controlador difuso encontra um novo ponto de equilíbrio para ele. Espera-se que sucessivas perturbações façam o sistema convergir para um ponto de equilíbrio ótimo. Tal investigação encontra-se atualmente em curso em um projeto sob minha execução, tratando-se de um sistema óptico cujas aberrações deverão ser minimizadas¹¹.

Aplicações reais em curso.

Apresentaremos, a seguir, uma lista (não esgotada, naturalmente) de aplicações correntes da Lógica Difusa:

1. controle automático de comportas em barragens de usinas hidroelétricas (Tokio Electric Power);

⁷ ISOTANI, Sadao; ALBUQUERQUE, Antonio R.P.L. MURATORE, Marcio; BONVENTI JR., Waldemar; A 1-D Pattern Recognition System Based on Fuzzy Logic. Submetido para publicação.

⁸ ISOTANI, Sadao; ALBUQUERQUE, Antonio R.P.L.; MURATORE, Marcio; BRASIL FILHO, Nilson. A recursive spline-based algorithm for sensor calibration design. Submetido para publicação.

⁹ BELLMAN, R. and ZADEH, L.A. Decision making in a Fuzzy Environment. **Management Science**, vol.17, p.144-64, 1970.

¹⁰ SIDAOU, H. and ISOTANI, S.; Steady-State optimization of Fuzzy Modeled Systems. **Comunicação Interna**, Escola Politécnica da USP, 1994.

¹¹ BONVENTI JR., Waldemar. Software para otimização de Sistemas Ópticas utilizando-se Lógica Fuzzy. Projeto de Pesquisa na Universidade de Sorocaba, 1997.

2. controle simplificado de robôs (Hirota, Fuji electric, Toshiba, Omron);
3. mira de câmaras em transmissões esportivas (Omron);
4. avaliação de controle de estoque (Yamaichi, Hitachi);
5. controle de flutuações de temperatura em sistemas de ar condicionado (Mitsubishi, Sharp);
6. controle de motores de automóveis (Nissan);
7. controle de rotas de automóveis (Nissan, Subaru);
8. otimização de aplicações em controle industrial (Apronix, Omron, Meiden, Sha, Micom, Mitsubishi, Nisshin-Denki, Oku Electronics);
9. posicionamento de camadas na construção de semicondutores (Canon);
10. planejamento e otimização de horários de tráfego de ônibus (Toshiba, Nippon-System, Keihan-Express);
11. sistema de arquivos de documentos (Mitsubishi Electric);
12. sistema de previsão de terremotos (Inst. de Sismologia, Japão);
13. diagnóstico de câncer (Kawasaki Medical School);
14. em combinação com Redes Neurais (Matsushita);
15. reconhecimento de símbolos manuscritos em computadores de bolso (Sony);
16. reconhecimento de motivos em pinturas e imagens com câmaras de vídeo (Canon, Minolta);
17. controle automático de aspiradores de pó, com reconhecimento das condições da superfície (Matsushita);
18. controle de iluminação de fundo para câmaras de vídeo (Sanyo);
19. compensação de vibrações em câmeras de vídeo (Matsushita);
20. controle de máquinas de lavar em um único botão (Matsushita, Hitachi);
21. reconhecimento de escrita à mão, objetos, vozes (CSK, Hitachi, Hosai University, Ricoh);
22. auxílio a vôos de helicóptero (Sugeno);
23. simulação de procedimentos legais (Meihi Gakuin Univ, Nagoya Univ.);
24. projeto de programas para processos industriais (Apronix, Harima, Ishikawajima-OC Engineering);
25. controle de velocidade e temperatura em maquinaria de forja em aço (Kawasaki Steel, New-Nippon Steel, NKK);
26. controle de sistemas de metrôs, para aumento de conforto da viagem e economia de potência (Hitachi);

27. consumo otimizado de combustível em automóveis (NOK, Nippon Denki Tools);
28. sensibilidade melhorada e eficiência em controle de elevadores (Fujitec, Hitachi, Toshiba);
29. auxiliar o aumento de segurança em reatores nucleares (Hitachi, Bernard, Nuclear Fuel division).

Um futuro difuso?

Sistemas baseados em Inteligência Artificial processam simbolicamente o conhecimento estruturado. Sistemas Neurais processam numericamente conhecimento desestruturado, varrendo bases de dados. Sistemas Difusos processam numericamente conhecimento estruturado. Atualmente, os sistemas difusos desenvolvidos misturam Sistemas Especialistas com Redes Neurais. Devido a esta estruturação, uma estimativa do sistema livre de modelagem tornou-se possível, permitindo que os sistemas difusos processem tanto a informação vaga quanto a informação precisa, simultaneamente.

Podemos crer que a Lógica Difusa, bem como as Redes Neurais e os Sistemas Especialistas, tornar-se-ão os componentes mais importantes dos programas de computador da próxima geração. Parte desses componentes já existem, e muitos problemas ainda difíceis serão solucionados por alguns deles, senão por todos em conjunto. Esses componentes não competem entre si, mas se complementam, numa cooperação. Esforços dos especialistas, atualmente, visam não somente a aperfeiçoá-los, mas a uni-los harmoniosamente no interior de um sistema computacional.

REFERÊNCIAS BIBLIOGRÁFICAS

1. BAUER, Peter; NOUAK, Stephan; WINKLER, Roman. A brief course in Fuzzy Logic and Fuzzy Control, Version: 1.2, **Internet homepage at <ftp://ftp.flll.uni-linz.ac.at/pub/info>**, December 4, 1996.
2. BELLMAN, R. and ZADEH, L.A. Decision Making in a Fuzzy Environment, **Management Science**, vol.17, p.144-64, 1970.
3. BONVENTI Jr., Waldemar. Software para Otimização de Sistemas Ópticas Utilizando-se Lógica Fuzzy, **Projeto de Pesquisa na Universidade de Sorocaba**, 1997.

-
4. COX, Earl; Fuzzy Fundamentals. **IEEE Spectrum**, October 1992, p. 58-61.
 5. ___ **The Fuzzy Systems Handbook** Boston : Academic Press Inc., 1994
 6. LEE, C.C., Fuzzy Logic in Control Systems: Fuzzy Logic Part I and II. **IEEE Transactions on Systems, Man and Cybernetics**, vol.20, part 2, p. 404-435, mar/abr 1990
 7. ISOTANI, Sadao; ALBUQUERQUE, Antonio R.P.L.; MURATORE, Marcio; BONVENTI Jr., Waldemar. A 1-D Pattern Recognition System Based on Fuzzy Logic, submetido para publicação.
 8. ISOTANI, Sadao; ALBUQUERQUE, Antonio R.P.L.; MURATORE, Marcio; BRASIL FILHO, Nilson. A Recursive Spline-Based Algorithm for Sensor Calibration Design, submetido para publicação.
 9. SIDAOU, H. and ISOTANI, S.; Steady-State Optimization of Fuzzy Modeled Systems. **Comunicação Interna**, Escola Politécnica da USP, 1994
 10. ZADEH, L.A., Fuzzy Sets. **Information Control**, vol.8, p.338-353, 1965.
 11. ___ Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. **IEEE Transactions on Systems, Man and Cybernetics**, vol.3, p.28-44, 1973.