2
Lógica Fuzzy

2.1. Introdução

A lógica fuzzy é uma extensão da lógica booleana, introduzida pelo Dr. Lofti Zadeh da Universidade da Califórnia / Berkeley no ano 1965. Foi desenvolvida para expressar o conceito de verdade parcial, de maneira que se possam determinar valores entre o limite "completamente verdadeiro" e "completamente falso". Isto significa que um valor lógico difuso é um valor qualquer no intervalo de 0 a 1. A lógica fuzzy torna-se importante na medida em que o mundo em que vivemos não é constituído por fatos absolutamente verdadeiros ou falsos.

Embora as técnicas de controle possam ser implementadas por modelos matemáticos determinísticos, as implementações baseada na lógica fuzzy freqüentemente têm um melhor desempenho, pelos seguintes pontos [7]:

- As estratégias de controle fuzzy imitam um comportamento baseado em regras vindas da experiência do especialista, em vez de um controle explicitamente restrito a modelos matemáticos como equações diferenciais. Portanto, é robusto em sistemas não-lineares sem requerer um modelo matemático.
- O controle fuzzy abrange um grande número de entradas, muito das quais são apenas para condições especiais. Portanto, algumas condições excepcionais (tais como alarmes) podem ser implementadas com um menor esforço computacional e flexível a modificações.
- A implementação de produtos comerciais baseados em estratégias de controle fuzzy destinadas ao mercado devem ser de custo mais baixo, freqüentemente mais eficiente e facilmente implementável em microprocessadores, em comparação a estratégias de controle

convencionais. Isto é devido a uma menor codificação e tempo computacional de execução.

A lógica fuzzy tem a capacidade de incorporar a forma humana de pensar em sistemas de controle. Dessa forma, o controlador fuzzy comporta-se conforme o raciocínio que o especialista utiliza para inferir as regras, baseadas nas informações que eles já conhecem.

A lógica fuzzy é uma variação da lógica booleana, que só apresenta os valores de “0” e “1”, sem nenhum termo médio. Entretanto, na lógica fuzzy, uma premissa pode assumir valores de pertinência (grau de verdade) intermediários. Assim, é possível descrever grandezas imprecisas como: altura (alto, baixo), quantidade (muito, razoável, pouco), idade (jovem, velho), etc. Em sentido mais amplo, a lógica fuzzy é quase sinônimo da teoria de conjuntos fuzzy, uma teoria na qual os elementos têm um grau parcial de pertinência [8].

2.2. Conjuntos Fuzzy

Na teoria clássica de conjuntos, o conceito de pertinência de um elemento a um conjunto é bem definido, de maneira que para um conjunto A em um universo U, o elemento simplesmente pertence ou não pertence àquele conjunto, como se mostra na seguinte função característica [9]:

$$f_A(x) = \begin{cases} 1 & \text{Se e somente se } \; x \in A \\ 0 & \text{Se e somente se } \; x \in A \end{cases} \tag{2.1}$$

Em um sentido mais amplo, Zadeh propôs a generalização da função característica, de modo que ela possa assumir infinitos valores no intervalo [0,1] e representado por pares ordenados.

$$A = \{\mu_A(x)/x\} \quad x \in U \tag{2.2}$$

Onde: $\mu_A(x)$: Representa o grau de pertinência de x com o conjunto A

A : Conjunto fuzzy.

x : A variável de interesse.

U : Universo de discurso.

Além disso, um elemento pode pertencer a mais de um conjunto fuzzy, com diferente grau de pertinência.

Na Figura 2.1 (a) pode-se observar que se um elemento x for movido em direção aos limites do conjunto A (cor amarela), no ponto de cruzamento ocorrerá repentinamente um degrau no comportamento de sua pertinência, inicialmente de membro para não membro. Também, o grau de pertinência nos limites é indeterminado.

Por outro lado, a lógica fuzzy pode perceber as variações ocorridas nos pontos transição de uma cor para outro. Os conjuntos (faixa de cores) são facilmente representáveis através de linguagem fuzzy (Figura 2.1 b). As funções de pertinência fuzzy podem representar a variação gradual nas tonalidades. Por exemplo, o mesmo elemento x possui um grau de pertinência 0,8 na função de pertinência de cor amarela, e grau de pertinência 0,2 na função de cor vermelha. Na Figura 2.1, o eixo "X" representa o universo de discurso do elemento x e o eixo "Y" representa o grau de pertinência definido entre 0 e 1.

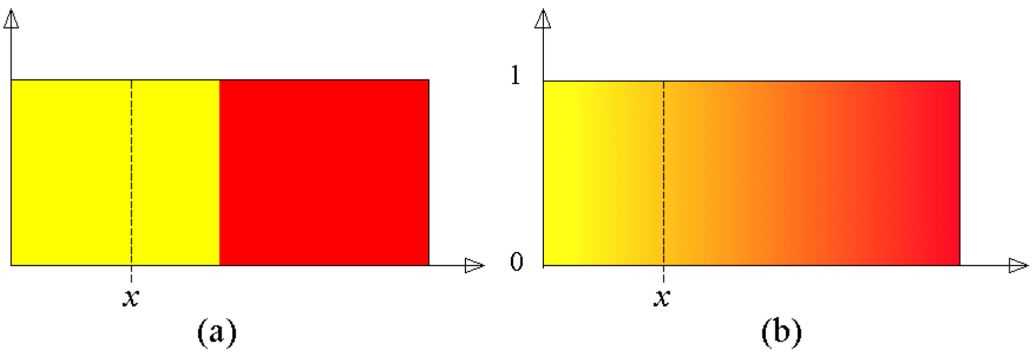

Figura 2.1. (a) lógica booleana (b) lógica fuzzy.

Neste momento, é importante definir o conceito de grau de pertinência. Ele é definido por um número no intervalo de [0,1] que determina "quanto" uma variável pertence a um determinado conjunto. Na lógica booleana, somente existem dois graus de pertinência: 0% se não pertence e 100% se pertence ao conjunto. No entanto, na lógica fuzzy existe uma faixa de valores entre 0% e 100%. O grau de pertinência é descrito pela Equação 2.2 [10].

2.2.1. Variáveis lingüísticas

A *variável lingüística* é uma variável que possui valores que não são números, mas sim palavras ou frases na linguagem natural. Elas são os nomes dos conjuntos fuzzy, os quais são representados por meio de funções de pertinência. Por exemplo, a *velocidade* de um pistão hidráulico pode ser uma variável lingüística com valores: *baixa*, *média* e *alta*, conforme mostrado na Figura 2.2.

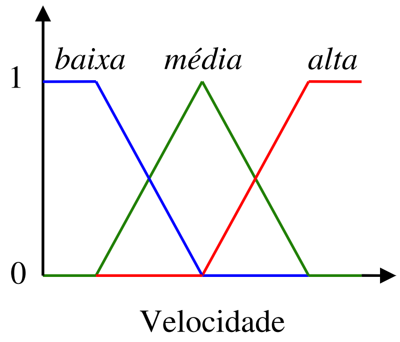

Figura 2.2. Variável lingüística

As variáveis lingüísticas têm a função de fornecer uma forma sistemática para as descrições aproximadas dos fenômenos complexos ou mal definidas, utilizando um tipo de descrição lingüística similar ao empregado pelos seres humanos. Isto permite o tratamento de sistemas muito complexos para serem analisados através de cálculos matemáticos.

Generalizando, as variáveis lingüísticas podem ser sentenças com um formato de linguagem padrão ou especificada pelo usuário, construídas a partir de termos primários (alto, baixo, grande, etc.), de modificadores (muito, pouco, levemente, etc.), de delimitadores (acima, abaixo, etc.) e conectores lógicos (não, e, e ou). Uma variável lingüística é formalmente caracterizada pela quíntupla (N, T(N), U, G, M), onde:

N : Nome da variável.

T(N) : Conjunto de nomes dos valores lingüísticos de N.

U : Universo de discurso.

G : Regra sintática para gerar os valores de N como uma composição de termos de T(N).

M : Regra semântica, para associar a cada valor gerado por G um conjunto fuzzy em U.

2.2.2. Função de Pertinência.

A função de pertinência (*FP*) é uma função numérica que atribui valores de pertinência fuzzy para valores discretos de uma variável, em seu universo de discurso [8]. Estas funções de pertinência podem ter diferentes formas, as quais dependem do critério utilizado pelo especialista para representar o contexto em que serão utilizadas. Por exemplo, considere-se a variável lingüística idade (de pessoas), constituídas pelos termos lingüísticos; T(idade) = {jovem, media, velho} correspondentes aos conjuntos fuzzy A, B e C, e cada uma definido por uma função de pertinência. Uma possível representação das funções de pertinência é mostrada na Figura 2.3.

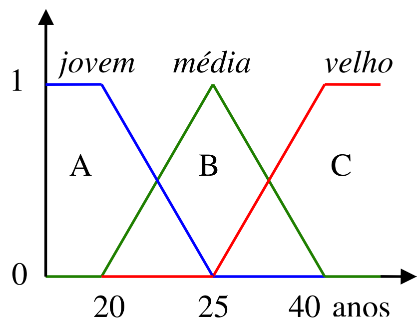

Figura 2.3. Funções de pertinência.

Na figura acima, a idade até 20 anos apresenta um grau de pertinência igual a 1 no conjunto A, à medida que a idade aumenta o grau de pertinência neste conjunto decresce; para uma idade de 25 anos é totalmente incompatível no conjunto A e compatível no conjunto B, e uma idade acima de 40 anos apresenta grau de pertinência diferente de 0 em C.

Observe-se que, na Figura 2.3, o contexto é importante na definição das funções de pertinência e de sua distribuição ao longo de seu universo de discurso, as quais são determinadas pelas noções que podem ter diferentes pessoas do contexto. Além disso, a forma (triangular, trapezoidal, gaussiana, etc.) das funções de pertinência pode ser definida baseada na experiência e perspectiva do especialista. Na prática, as formas escolhidas podem ser alteradas em função dos resultados obtidos.

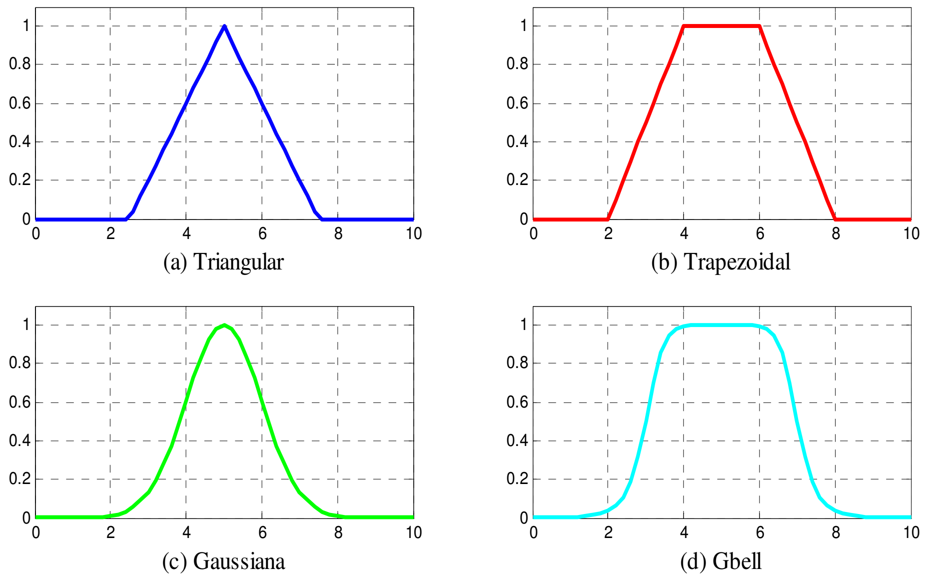

Figura 2.4. Funções de pertinência: (a) Triangular (b) Trapezoidal (c) Sigmoidal (d) função de Bell.

Algumas recomendações práticas podem ser mencionadas como segue [7].

- O número de funções de pertinência por variável deve estar entre 2 e 7; com um maior número de conjuntos, se tem uma maior precisão, mas a demanda computacional também é maior; e para valores muito maiores, não há melhorias significativas.
- Os formatos mais freqüentemente encontrados são os triangulares e trapezoidais, pela facilidade de serem gerados. Em aplicações onde se requer um desempenho suave de importância crítica, podem ser usadas funções tipo sigmóides e *spline*.
- Outro fator que afeta a precisão é o grau de superposição das funções de pertinência; experimentalmente foram determinados como adequados valores na faixa de 25 % até 75%.

2.2.3. Operações entre Conjuntos Fuzzy

As operações que podem ser feitas com conjuntos fuzzy são baseadas na teoria clássica de conjuntos. Algumas das operações elementares mais utilizadas são: união, interseção, e complemento, as quais são definidas pelas equações (2.3), (2.4) e (2.5) respectivamente e mostradas na Figura 2.4 [9, 11, 12].

União : $\mu_{(A \cap B)}(x) = \max(\mu_{(A)}(x), \mu_{(B)}(x))$ (2.3)

Interseção : $\mu_{(A \cap B)}(x) = \min(\mu_{(A)}(x), \mu_{(B)}(x))$ (2.4)

Complemento : $\mu_{(\bar{A})}(x) = 1 - \mu_{(A)}(x)$ (2.5)

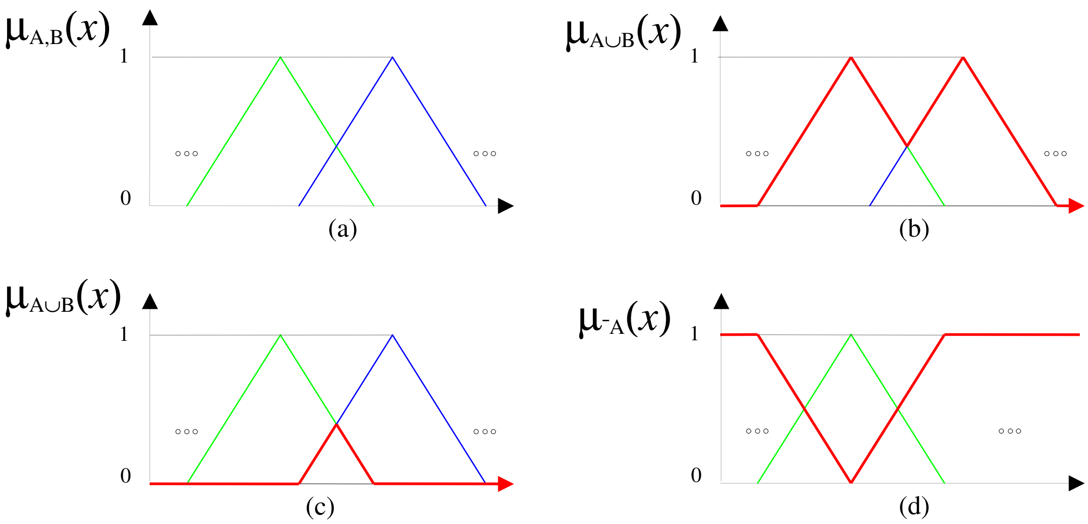

Figura 2.4. Operações entre conjuntos fuzzy: (a) Conjuntos A e B, (b) União, (c) Interseção e (d) Complemento. [2].

Embora a união e interseção possam ser descritas também por meio de outros operadores, Zadeh os representou com os operadores *min* e *max*. Alem disso, sugeriu a soma algébrica para a união fuzzy e o produto algébrico para a interseção fuzzy, como se apresenta na Equação (2.6) e (2.7); posteriormente, com o objetivo de generalizar, foram definidos operadores baseados no conceito de norma triangular (*norma-t*) e co-norma triangular (*co-norma-t* ou *norma-s*).

Soma Algébrica : $\mu_{(A \cup B)}(x) = \mu_{(A)}(x) + \mu_{(B)}(x) - \mu_{(A)}(x).\mu_{(B)}(x)$ (2.6)

Produto Algébrico : $\mu_{(A \cap B)}(x) = \mu_{(A)}(x).\mu_{(B)}(x)$ (2.7)

2.3. Sistema de Inferência Fuzzy

O sistema fuzzy é um modelo geral que permite a identificação dos módulos que compõem tal sistema, fornecendo assim a idéia do fluxo de informação dentro do mesmo. Basicamente ele é compostos por 3 etapas, como é mostrado na Figura 2.5, onde estão definidas as funções de cada uma das etapas.

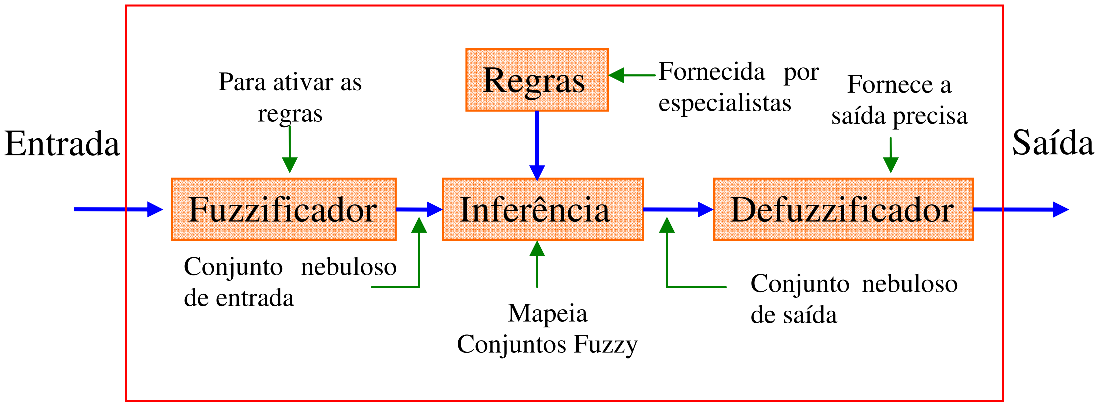

Figura 2.5. Sistema de Inferência Fuzzy.

Na figura acima, considera-se a entrada não fuzzy, e ela é resultado de medições na maioria das aplicações práticas. Portanto, é necessário efetuar-se a conversão destas entradas em uma representação conhecida como conjuntos fuzzy, o que se denomina fuzzificação. Além disso, nesta etapa ocorrem as ativações das regras para as diferentes situações.

Na segunda etapa, estabelece-se a base de regras, como a relação das variáveis de entrada e saída, as quais são obtidas pelo conhecimento e pela experiência do especialista da aplicação. Uma vez obtido o conjunto fuzzy de saída resultante do processo de inferência, é necessário efetuar a interpretação dessa informação, pois nas aplicações práticas são requeridas saídas precisas, o que é realizado na etapa de defuzzificação [9].

2.3.1.Fuzzificação

A fuzzificação é a conversão das entradas exatas (número reais) para o domínio fuzzy. O fuzzificador atribui valores lingüísticos (graus de pertinência) empregando funções de pertinência às variáveis de entrada. Isto se considera como uma etapa de pré-processamento dos sinais de entrada, reduzindo o numero de valores a ser processado o que significa um menor esforço computacional.

2.3.2. Regras e Inferência Fuzzy

Regra fuzzy são implicações lógicas que relacionam os conjuntos fuzzy de entrada com os de saída. Geralmente são fornecidas por um especialista, em forma de sentenças lingüísticas, constituindo um aspecto fundamental no desempenho de um sistema de inferência fuzzy, como mostrado abaixo.

Se x è A **e** y é B, **então** z é C.

Onde, A e B são os conjuntos fuzzy de entrada, relativos à parte conhecida como antecedentes ou premissas, enquanto C é o conjunto fuzzy de saída, relativo à parte conhecida como conseqüente ou conclusão [8]. Estas regras podem ser definidas previamente ou alternativamente geradas automaticamente a partir de uma base de dados.

Na etapa de inferência, ocorrem as operações dos conjuntos propriamente ditas, como a combinação dos *antecedentes* das regras do tipo *SE – ENTÃO* , gerando o conjunto de saída fuzzy. Na Figura 2.6, apresenta-se o processo de inferência fuzzy.

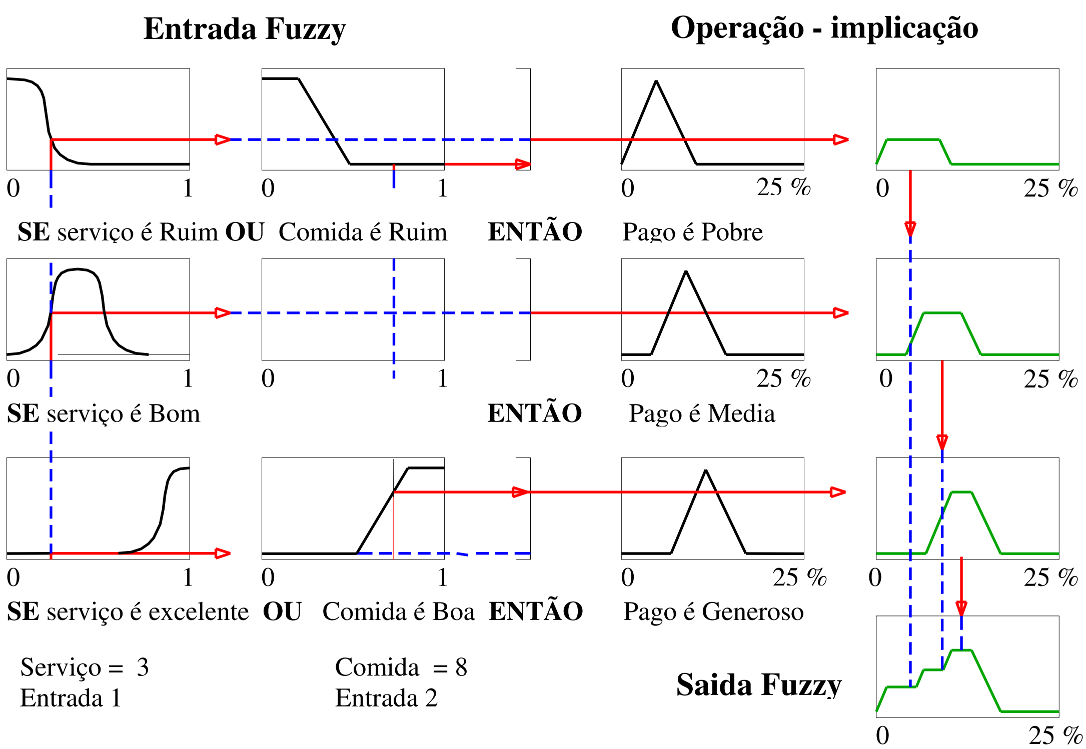

Figura 2.6. Regra e inferência fuzzy [8].

Na figura acima, mostra-se como ocorrem às operações na etapa de inferência; a entrada exata é transformada em entradas fuzzy, e com esses valores calculam-se as operações - implicação e finalmente obtém-se a saída fuzzy.

2.3.3. Defuzzificação.

No processo de defuzzificação, é efetuada a interpretação do conjunto fuzzy de saída inferida pelas regras, com o objetivo de obter um valor numérico. Isto se faz necessário porque em aplicações práticas são requeridas saídas precisas com algum significado físico. Por exemplo, no controle de uma planta utilizando um controlador fuzzy, este deve fornecer à planta sinais precisos, já que apresentar um conjunto fuzzy na entrada da planta não teria significado algum.

Na prática, o método de defuzzificação mais popular é o centro de Área (C-o-A), o qual retorna o centro da área debaixo a curva, como se mostra na Figura 2.7. Outros métodos utilizados são: Centro do Máximo (C-o-M), Média do Máximo (M-o-M) e o Maior do Máximos [13].

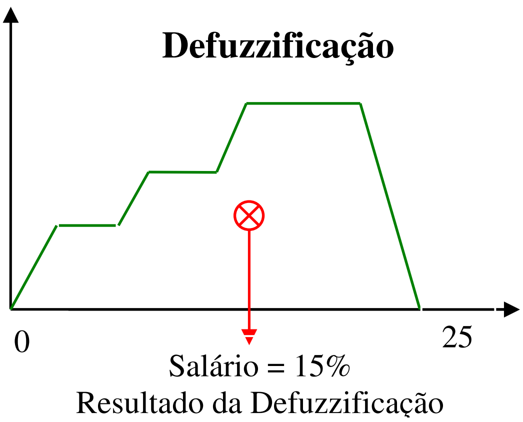

Figura 2.7. Processo de Defuzzificação.

2.4. Tipos de Sistema Fuzzy

Na literatura existem vários modelos de sistemas fuzzy, nos quais geralmente os antecedentes das regras estão formados por conjuntos fuzzy (proposições lingüísticas), e a diferença entre os modelos se dá no "conseqüente" das regras. Algum dos modelos mais conhecidos são o modelo de Mandani e o modelo de Tagaki-Sugeno.

2.4.1. Modelo de Mandani

O modelo de Mandani utiliza conjuntos fuzzy nos "conseqüentes" das regras fuzzy. Neste modelo, a saída da etapa de inferência é representada por um conjunto fuzzy, que é o resultado da agregação das saídas inferida por cada uma das regras, a qual na seguinte etapa gera uma saída exata utilizando um dos métodos de defuzzificação já mencionados.

A característica básica do modelo tipo Mandani é o fato que tanto os antecedentes como os conseqüentes são mapeados com conjuntos fuzzy. Por exemplo, uma regra típica num modelo Mandani é da seguinte forma:

SE *Erro* é "Grande" **E** a *Derivada de Erro* é "Pequena" **Então** *Torque* é "Alto".

No caso em que cada um das regras de inferência tenha mais de uma variável de entrada, é necessário aplicar uma técnica de agregação dos conjuntos antecedentes, que neste caso geralmente é dada pelas t-normas (*min* e *produto*). Além disso, nas aplicações práticas têm-se N regras na etapa de inferência, das quais são gerados N conjuntos conseqüentes, um por cada regra. Para obter o conjunto final de saída da etapa de inferência, é feita a composição dos N conjuntos conseqüentes utilizando a s-norma (*max*) [7- 10].

2.4.2. Modelo de Takagi-Sugeno

No modelo de Takagi-Sugeno (TS), o conseqüente de cada regra é representado em função das variáveis de entrada; e a saída final de todas as regras é determinada pela média ponderada das saídas geradas por cada um das regras. Nesse caso, os coeficientes de ponderação são definidos pelos graus de ativação das respectivas regras.

A fuzzificação das entradas com a aplicação dos operadores fuzzy (operação dos antecedentes) é feita de igual forma que no modelo Mandani, com a diferença que a saída é uma função linear ou constante. Uma regra típica de um modelo Sugeno é da seguinte forma:

SE $Erro = x$ **e** a $Derivada\ de\ Erro = y$ **então** *Torque* é $\tau_i = a.x + b.y + c$.

onde τ_i é o valor de saída de cada um das regras. Além disso, a partir do produto da operação do antecedente de cada regra, obtém-se um peso ω_i, o qual determina o fator de influência de cada regra no resultado final. Por exemplo, na regra mostrada acima, com o tipo de operador *AND*, com $Erro = x$ **e** $Derivada\ de\ Erro = y$**,** o fator de influência é:

$$\omega_i = AND(f(x), g(y)) \tag{2.8}$$

onde f e g são as funções de pertinência para as entradas *Erro* e *Derivada de Erro.* A saída final é determinada pela equação:

$$out = \frac{\sum_{j=1}^{n} w_i.\tau_i}{\sum_{j=1}^{n} w_i} \tag{2.9}$$

2.5. Vantagens e Desvantagens dos Sistemas Fuzzy

As principais vantagens e desvantagens dos sistemas fuzzy, apresentadas em aplicações práticas, são as seguintes [11]:

2.5.1. Vantagens dos Sistemas Fuzzy

- A capacidade de controlar sistemas com muitas variáveis de saída utilizando um só controlador fuzzy, com um bom desempenho.
- A facilidade de utilizar expressões utilizadas na linguagem natural na elaboração das proposições lingüísticas.
- A habilidade de controlar processos com característica não-linear e de alta ordem, na qual a determinação do modelo matemático e o controle clássico do sistema são muito complexos.
- A facilidade de implementar técnicas de controle baseadas na experiência de um especialista e em aspectos intuitivos, utilizando proposições lingüísticas (regras) e entradas imprecisas.

2.5.2. Desvantagens dos Sistemas Fuzzy

Algumas das limitações que apresentam os sistemas fuzzy são as seguintes:

- A dificuldade de análise de aspectos de optimalidade, estabilidade e robustez.
- A influência da grande quantidade de parâmetros na configuração geralmente feita pelo usuário, algumas das quais são: número de funções de pertinência de cada variável, número de regras, seleção dos métodos de implicação e agregação, método de defuzzificação, assim como os parâmetros de cada função de pertinência.
- Geralmente a precisão do sistema fuzzy é limitada pela experiência do especialista na configuração dos parâmetros, a qual é determinada pelo conhecimento do processo pelo especialista.

No capítulo seguinte apresentam-se os fundamentos teóricos das redes neurais artificiais, o modelo de sua célula básica, sua arquitetura e os algoritmos de aprendizagem.