

ANÁLISE COMPARATIVA DOS MÉTODOS MAMDANI E SUGENO PARA UM CONTROLADOR DE LUMINOSIDADE

Adailson Pereira de Melo Filho¹, Marcus Vinicius Silverio Costa²

Resumo: O seguinte trabalho tem como proposta auxiliar na compreensão da lógica *fuzzy*, através de uma análise comparativa entre dois métodos, sendo eles o de Mamdani e Sugeno, em controladores de luminosidade que utilizam a teoria de conjuntos *fuzzy*, desenvolvida por L.A. Zadeh em 1965. As comparações realizadas, foram possíveis, somente após um grande estudo a respeito dos métodos de inferência *fuzzy*, através de pesquisas em revistas científicas e também utilizando outros trabalhos acadêmicos como base para o desenvolvimento dos controladores. Utilizando a ferramenta do *software* MATLAB, chamada *fuzzy logic toolbox*, para a criação dos sistemas de controle. Foi desenvolvido dois controladores, em que, um deles foi feito através do método de Mamdani e o outro o de Sugeno. Durante a pesquisa, foram realizados cálculos para identificar as diferenças percentuais entre os controladores, usando o controlador do tipo Mamdani como referência. Após a realização dos cálculos, observou-se que o controlador do tipo Mamdani ofereceu uma resposta mais suave, também foi possível notar que, para níveis de luminosidade mais baixos, a diferença percentual entre os dois controladores é relativamente pequena, mas, conforme os valores de luminosidade iam aumentando, essa diferença se tornava cada vez mais visível. Com isso foi concluído que a utilização da lógica *fuzzy*, é uma ótima alternativa para sistemas de controle e que, conhecer a diferença entre os métodos de inferência, é importante para auxiliar na escolha das aplicações dos métodos.

Palavras-chave: Análise; Mamdani; Sugeno; *Fuzzy*; Luminosidade.

1. INTRODUÇÃO

A lógica nebulosa ou lógica *fuzzy*, foi desenvolvida por L.A. Zadeh em 1965. Segundo [1], a lógica *fuzzy* é uma técnica que integra o modo humano de pensar em um sistema de controle. Um controlador *fuzzy* pode ser desenvolvido para agir de acordo com o raciocínio dedutivo, que é o processo utilizado pelas pessoas para tirar conclusões baseadas em informações previamente conhecidas. [2] disse que um conjunto *fuzzy* é uma classe de objetos com um continuum de graus de pertinência. Tal conjunto é caracterizado por uma função de pertinência (característica) que atribui a cada objeto um grau de pertinência que varia entre zero e um. Para [3], é uma metodologia para expor dados imprecisos e qualitativos, comuns na comunicação humana, em uma sequência de forma compreensível pelos computadores. Esses sistemas são ferramentas poderosas, que possibilitam a expressão de conceitos que não sejam bem definidos, nos quais usualmente é empregado um nome ou adjetivo para descrevê-los.

O trabalho de [4], será utilizado como base para realização de uma comparação entre os métodos de inferência Mamdani e Sugeno, onde o mesmo apresentou a aplicação da utilização da lógica *fuzzy* em um controlador de luminosidade. As comparações entre os métodos, podem auxiliar em uma maior compreensão a respeito de vasto assunto, que é a lógica *fuzzy*. Através de um programa computacional, o estudo tentará mostrar que, ao se conhecer as diferenças entre os métodos, a escolha a respeito de qual tipo de inferência utilizar, ajudará a melhorar a eficiência dos controladores que utilizam a lógica *fuzzy* como base.

Este presente artigo tem como objetivo realizar uma análise comparativa dos métodos Mamdani e Sugeno em controladores de luminosidade, na busca de auxiliar na compreensão das diferenças entre os dois métodos, bem como, analisar os resultados encontrados através do estudo e mostrar que, os controladores que utilizam a lógica *fuzzy* podem auxiliar os profissionais da área de controle e automação.

2. LÓGICA FUZZY

Segundo [5], a utilização de algoritmos *fuzzy* na solução de problemas de controle começou com o trabalho pioneiro de Mamdani (1973, *apud* Sugeno, 1985), influenciado pelos estudos de Zadeh. Os controladores que utilizam os métodos de Mamdani e Sugeno possuem estruturas semelhantes, mas diferem em alguns aspectos.

Nas seções subsequentes, serão detalhados os conceitos de cada um desses métodos, bem como suas características específicas. Serão exploradas as estruturas e características de cada abordagem, proporcionando

uma compreensão mais abrangente sobre a aplicação dos controladores *fuzzy* no contexto dos sistemas de controle.

2.1. Funções de pertinência

De acordo com [6], na teoria clássica dos conjuntos, um elemento ou faz parte de um conjunto ou não. Considerando um universo U e um elemento específico $x \in U$, o grau de pertencimento $\mu_A(x)$ em relação a um conjunto $A \subseteq U$ é definido na equação (1) como:

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

A função $\mu_A(x): U \rightarrow \{0,1\}$ é chamada de função característica na teoria clássica de conjuntos.

Na visão de [1] a propriedade fundamental da lógica *fuzzy* é que a função de pertinência $\mu_A(x)$ tem todos os valores dentro do intervalo $[0,1]$. Isso significa que um elemento pode ser membro parcialmente de um conjunto, indicado por um valor fracionário dentro do intervalo numérico. Desta forma, de acordo com [7], “a função de pertinência associa a cada elemento x pertencente a X um número real μ no intervalo $[0,1]$, que representa o grau de pertinência do elemento x ao conjunto A , isto é, o quanto é possível para o elemento x pertencer ao conjunto A ”.

Segundo [1], as funções de pertinência podem ser triangulares, trapezoidais ou gaussianas na figura 1 a seguir pode-se observar os tipos de funções mencionadas.

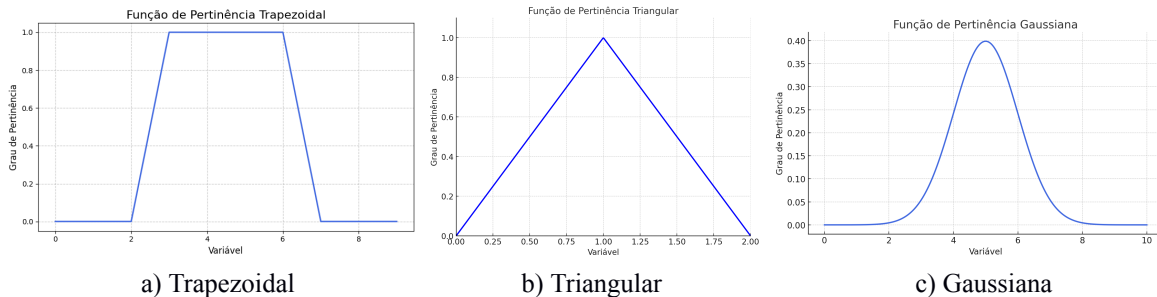


Figura 1. Tipos de funções de pertinência (Fonte: Adaptada de [8])

2.2 Variáveis linguísticas

Segundo [9], “uma variável linguística é uma variável cujos valores são definidos por conjuntos *fuzzy*. Por exemplo, a temperatura em um determinado processo pode ser considerada uma variável linguística que assume valores como baixa, média e alta.” [10] afirma que esses valores são descritos através de conjuntos *fuzzy*, representados por funções de pertinência, como ilustrado na figura 2 a seguir:

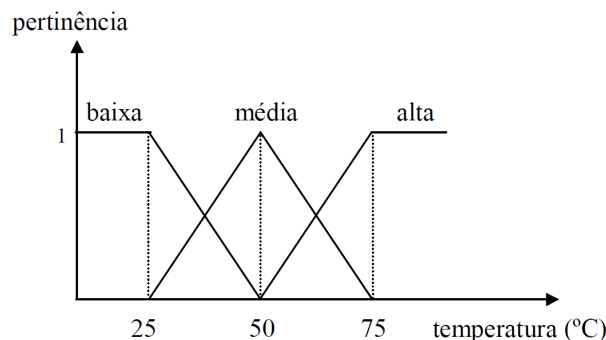


Figura 2. Funções de pertinência para a variável temperatura (fonte: [10])

2.3 Base de regras *fuzzy*

De acordo com a visão de [11], “as regras *fuzzy* são regras normais utilizadas para operar, da maneira correta, conjuntos *fuzzy*, com o intuito de obter consequentes. Para criar tais regras é preciso de um raciocínio coerente com o que se deseja manusear e obter.”

Para [1], a base de conhecimento define o modelo do sistema a ser controlado. Ela é composta por uma base de dados (funções de pertinência linguísticas) e uma base de regras *fuzzy* linguísticas. A base de dados fornecerá as definições numéricas necessárias para as funções de pertinência que são utilizadas no conjunto de regras *fuzzy*. A base de regras descreve quais são os objetivos de controle, por meio de um conjunto de regras de controle geralmente expressas de forma linguística. Por exemplo, considere a seguinte afirmação usando essa técnica: "se a temperatura está baixa, então o aquecedor aumentará a intensidade". Utilizando variáveis linguísticas, os termos baixar e aumentar podem ser definidos dessa forma.

Segundo [10], na teoria de conjuntos nebulosos, uma regra, será disparada se houver um grau de similaridade ou de compatibilidade diferente de zero entre a premissa e o antecedente da regra; o resultado será um consequente com grau de similaridade não nulo em relação ao consequente da regra.

3 INFERÊNCIAS FUZZY

Na visão de [5], existem dois tipos de controladores nos quais a ação de controle é derivada através da especificação de um conjunto de instruções, conhecidas como regras de controle *fuzzy*, isto é, de um algoritmo *fuzzy*. Essas regras seguem o princípio do GMP (*Generalized Modus Ponens*), onde uma regra específica (Ri) pode ser formulada da seguinte maneira.

Antecedente: *se x é A'*

Regra (Ri): *se x é A então y é B*

Consequente: *y é B'*

Nesse contexto, [5] argumenta que *x* representa uma variável do processo, enquanto *y* é a variável de controle. *A'* e *A* são conjuntos *fuzzy* de *x*, e *B* e *B'* são conjuntos *fuzzy* de *y*. Importante notar que *A'* e *B'* não precisam ser idênticos a *A* e *B*.

No processo de inferência, [6] destaca que os dados *fuzzy* de entrada são processados junto com as regras, a fim de inferir as ações de controle *fuzzy*, aplicando o operador de implicação *fuzzy* e as regras de inferência da lógica *fuzzy*. conforme [11], esse processo pode ser realizado utilizando modelos de inferência, que devem ser selecionados conforme o tipo de problema a ser resolvido, resultando em um melhor processamento. Na visão de [4], dentre os diversos métodos de inferência disponíveis, o método Mamdani é o mais comumente empregado.

Na visão de [4], esse tipo de inferência tem uma precisão em sistemas de controle, ou seja, projetos que possuem como base o fator de decisão e exatidão, utilizam-se desse método, enquanto técnicas como o de Sugeno, são mais utilizadas para fazer o design de sistemas. Na figura 3 pode-se observar um fluxograma com uma estrutura de sistema baseada em regras *fuzzy*.

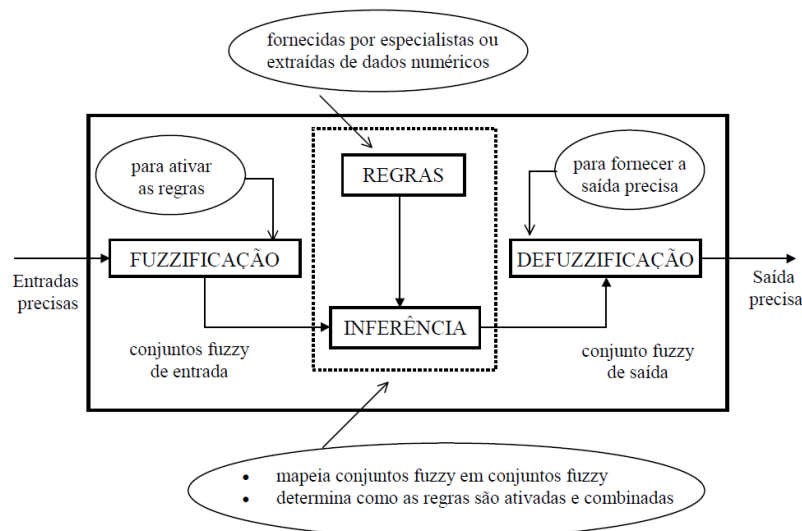


Figura 3. Estrutura de sistema baseadas em regras nebulosas. (Fonte: [10])

3.1 Controlador com inferência Mamdani

Na visão de [10] o método de inferência do tipo Mamdani utiliza uma base de regras caracterizada modelo nebuloso linguístico. Segundo [12], cada regra pode associar várias entradas a diferentes saídas. As entradas são

conectadas pelo termo lógico "E", utilizando a função 'mínimo' como mostra o seguinte exemplo: Se temperatura ALTA E ventilação BAIXA, Então Ar-condicionado LIGADO na potência MÁXIMA. Para combinar as condições de cada regra, utiliza-se o conectivo lógico "OU", utilizando a função 'máximo' que conecta as regras *fuzzy* da base de regras. O seguinte exemplo mostra como usá-lo: Se temperatura ALTA OU umidade ALTA, Então Ar-condicionado LIGADO.

De acordo com [11], este método é possivelmente o mais utilizado e foi proposto por Ebrahim Mamdani em 1975. O processo é realizado em quatro passos.

1. Fuzzificação das variáveis;
2. Avaliação das regras *fuzzy*;
3. Agregação das regras *fuzzy*;
4. Defuzzificação;

3.1.1 Fuzzificação

Para [6], “a interface de fuzzificação toma os valores das variáveis de entrada, faz um escalonamento para condicionar os valores a universos de discurso normalizados e fuzzyfica os valores, transformando números em conjuntos *fuzzy*, de modo que possam se tornar instâncias de variáveis linguísticas”. Por exemplo, se a entrada é a temperatura, ela pode ser categorizada como "baixa", "média" ou "alta". Segundo [4], neste processo são feitas as seguintes determinações:

- Analisar o problema;
- Definir quais serão as variáveis utilizadas;
- Determinar as Funções de Pertinência (FP);
- Criar as Regiões.

3.1.2. Avaliação das regras *fuzzy*

Segundo [11], após fuzzyficar as entradas, elas são aplicadas nos antecedentes para calcular o valor do consequente em cada regra. Para antecedentes compostos, são usados os operadores "e" e "ou" para que seja possível encontrar um único resultado. O operador "ou" utiliza a operação de união, onde o mesmo irá selecionar o maior grau de pertinência, enquanto o operador "e", irá aplicar a operação de interseção em que o menor grau de pertinência é selecionado. Depois que o antecedente é definido em um único valor, é necessário calcular o valor do consequente por meio de um método de correlação, em que o mais comum é o método chamado "*clipped*", onde o consequente é ajustado para o nível de valor verdade do antecedente avaliado, transferindo esse valor diretamente para o consequente da regra.

3.1.3. Agregação das regras *fuzzy*

Conforme [11] apresenta, nesta fase, todas as funções de pertinência dos consequentes de cada regra são combinadas, resultando em um único conjunto *fuzzy*. Dessa maneira, ocorre a agregação das saídas. Na figura 4 é demonstrado como esse processo é realizado.

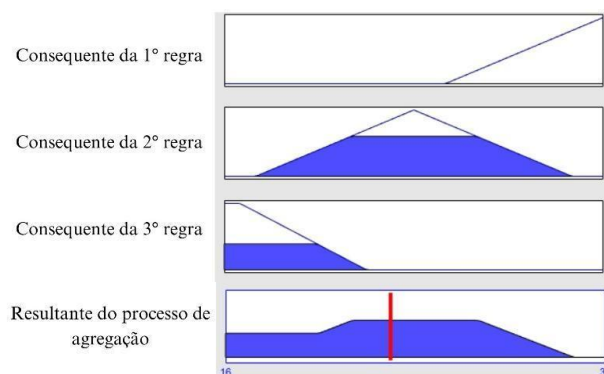


Figura 4. Consequentes das regras e o resultante dos consequentes das regras 1, 2 e 3. (Fonte: Autoria própria)

3.1.4 Defuzzificação

No processo de defuzzificação no ponto de vista de [1], consiste em obter-se um único valor discreto, utilizável numa ação de controle concreta no mundo real, a partir de valores *fuzzy* de saída obtidos. Este único valor discreto representa um compromisso entre os diferentes valores *fuzzy* contidos na saída do controlador. Para [11], o método mais comum para transformar valores nebulosos em um número claro é chamado de técnica do centroide (CDA). Basicamente, ele encontra o ponto exato onde uma linha imaginária poderia dividir o gráfico em duas partes iguais. Quanto maior for a área considerada no cálculo, menos preciso o resultado, mas o cálculo se torna mais rápido. A equação usada para calcular esse ponto pode ser representada da seguinte forma como expressão (2). [6] mostra que há também outros métodos que são utilizados para realizar a defuzzificação, como “o critério do máximo (MAX), que escolhe o ponto onde a função inferida tem seu máximo, a média dos máximos (MDM), que representa o valor médio dentre todos pontos de máximo quando existe mais de um máximo”.

$$CDA = \frac{\sum_{i=0}^n \mu(x_i) \cdot x_i}{\sum_{i=0}^n \mu(x_i)} \quad (2)$$

A figura 5 mostra um exemplo dos métodos de defuzzificação.

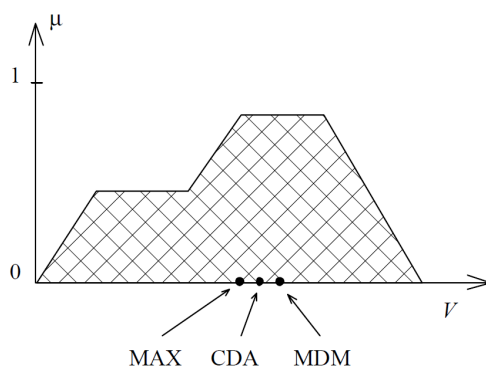


Figura 5. Métodos de defuzzificação. (Fonte: [6])

3.2 Controlador com Inferência do tipo Takagi e Sugeno

Segundo [13], Takagi e Sugeno propuseram em 1985 uma nova maneira de trabalhar com a lógica *fuzzy*, como uma alternativa ao método de Mamdani. Esse método utiliza regras *fuzzy*, mas, diferentemente do método Mamdani, o consequente é representado por funções lineares, em vez de conjuntos *fuzzy*. Esse modelo é uma melhor opção, onde os sistemas são mais complexos e com maiores dimensões, onde não podem ser solucionados de maneira mais fácil pelo método de Mamdani.

Conforme observado por [10], os modelos funcionais de Takagi e Sugeno (1985) operam de forma semelhante ao método de Mamdani, pois cada regra irá influenciar proporcionalmente ao seu grau de ativação. Porém, como os consequentes das regras, que são definidos como funções das variáveis do antecedente, a inferência resultante de uma base de regras é calculada como uma média das funções de cada regra, que é ponderada pelos seus respectivos graus de ativação. Com isso, segundo [13], o modelo (TSK) não se faz necessário realizar uma defuzzificação. Isso porque, em vez de um conjunto difuso, obtém-se um conjunto de funções lineares. Segundo [5] no método (TSK) as regras R_i podem ser definidas da seguinte maneira:

Regra (R_i): se x é A_i e y é B_i então $z = f_i(x, y)$

Então conforme [5], “O valor de z pode também ser definido como um valor constante, que pode ser interpretado como um conjunto *fuzzy* com a característica especial de apresentar um único valor com pertinência igual a um e todos os demais com pertinência zero.”. [5] ainda diz que, o uso desse tipo de conjunto possibilita a definição de regras que classificam a resposta do controlador de maneira direta, sem complicar o processo de obtenção da resposta final.

4. IMPLEMENTAÇÃO COMPUTACIONAL

Nesta seção será apresentado as etapas para o desenvolvimento do sistema de controle dos controladores, usando o método Mamdani e Sugeno, e também os processos da implementação dos controladores, mostrando a montagem do circuito de iluminação, assim como a criação dos controladores, utilizando a ferramenta *Fuzzy Logic Toolbox* do software MATLAB.

Para que seja possível realizar a implementação da lógica *fuzzy* nos controladores de luminosidade, será necessário fazer uso de uma ferramenta computacional chamada *Fuzzy Logic Toolbox*, essa ferramenta é de um software, chamado MATLAB. Para [14]. Este produto permite que seja especificado e configurado entradas, saídas, funções de pertinência e regras de sistemas de inferência *fuzzy*. A caixa de ferramentas permite o ajustar de maneira automática, funções de pertinência e regras de um sistema de inferência *fuzzy* a partir de dados. A implementação que será realizada neste trabalho é inspirada no trabalho de [4].

4.1 Funcionamento do Sistema de Controle de Iluminação

O sistema desenvolvido para controle da iluminação utiliza um sensor LDR, um Arduino Uno e o MATLAB, integrando processamento de dados com um sistema de inferência *fuzzy*.

- Sistema de Inferência *Fuzzy* no MATLAB: É o ponto central do sistema de controle é onde o modelo de inferência *fuzzy* é implementado no MATLAB. O sistema irá processar as informações do sensor de luminosidade e determinará, de maneira precisa, as configurações necessárias na iluminação do ambiente.
- Central de Processamento no MATLAB: Recebe, via comunicação USB, os dados capturados pelos sensores. Com base nas regras *fuzzy*, são geradas instruções para o ajuste da iluminação.
- Comunicação com Arduino via USB: A comunicação entre o MATLAB e o Arduino Uno é feita através de cabo USB. O MATLAB enviará os comandos para o Arduino, que, tem como função, controlar o circuito de iluminação.
- Arduino Uno: O Arduino tem como finalidade coletar os dados do sensor LDR e controlar a intensidade da iluminação.
- Sensor LDR (*Light Dependent Resistor*): Um sensor LDR é usado para monitorar a intensidade da luz no ambiente. Ele converterá a luz em sinais elétricos, que são processados pelo Arduino.
- Circuito de Iluminação: Tendo como sua funcionalidade, a regulação da corrente que é fornecida ao LED, ajustar a intensidade de acordo com as instruções do MATLAB.
- LED: É o dispositivo de saída do sistema, cuja intensidade luminosa será ajustada de acordo com a luz do ambiente.

4.2 Descrição do controlador de luminosidade

4.2.1 Variáveis de entrada e saída

A variável de entrada que descreve a luminosidade do local na qual se deseja controlar a intensidade luminosa é “luminosidade”, onde a mesma pode assumir os valores de “muito baixa”, “baixa”, “média”, “alta” e “muito alta” os conjuntos *fuzzy* associados a essa variável é mostrada na figura 6. Como o microcontrolador do Arduino Uno tem um conversor analógico-digital (ADC) de 10 bits, o que significa que a leitura analógica é limitada a uma resolução de 0 a 1023. O Range utilizado para cada variável linguística, será de 0 a 1023. Para a variável de entrada foi utilizada as funções de pertinência triangular e trapezoidal, onde os valores de cada função são mostrados na tabela 1 abaixo.

Tabela 1. Valores usados para os vértices das funções de pertinência de entrada. (Fonte: Autoria própria)

Tipo de função	Muito baixa	Baixa	Média	Alta	Muito alta
Triangular	0;0;100	-	-	-	850;1023;1023
Trapezoidal	-	50;100;250;350	250;350;500;650	550;650;750;850	-

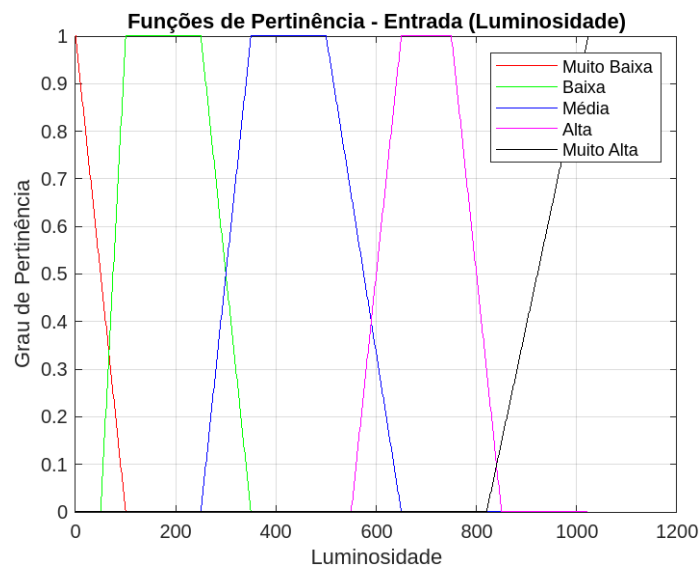


Figura 6. Funções de pertinências de entrada. (Fonte: Autoria própria)

A variável de saída “intensidade”, irá controlar a intensidade luminosa do LED, possuindo as seguintes funções de pertinência: “muito baixa”, “baixa”, “média”, “alta” e “muito alta”. Como a variável de saída do controlador Mamdani é diferente do controlador Sugeno, ela será apresentada posteriormente.

4.2.2 Base de regras

As regras adotadas para os dois controladores são mostradas na tabela 2, possuindo 5 regras ao todo.

Tabela 2. Regras adotadas para os controladores. (Fonte: Autoria própria)

Regras	Se a “ luminosidade ” é	Então a “ intensidade ” é
	...	...
1	Muito baixa	Muito alta
2	Baixa	Alta
3	Média	Média
4	Alta	Baixa
5	Muito alta	Muito baixa

4.2.3 Descrição do controlador Mamdani

Para o controlador Mamdani foi implementado funções de pertinência do tipo triangular e trapezoidal, para a variável de saída como é mostrado na figura 7. Os valores dos intervalos de cada função são mostrados na tabela 3.

Tabela 3. Valores de vértices usados para as funções de pertinência de saída. (Fonte: Autoria própria)

Tipo de função	Muito baixa	Baixa	Média	Alta	Muito alta
Triangular	0;0;1	-	-	-	0.85; 1; 1
Trapezoidal	-	0.05; 0.15; 0.25; 0.35	0.25; 0.4; 0.6; 0.7	0.6; 0.675; 0.85; 0.9	-

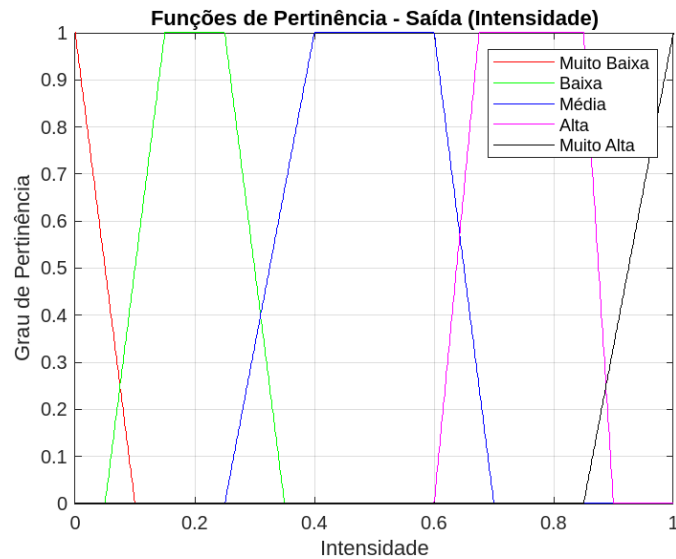


Figura 7. Funções de pertinências de saída Mamdani. (Fonte: Autoria própria)

Foi utilizado a defuzzificação, o tipo centroide, buscando uma superfície de controle suave e contínua. Na figura 8 é apresentado a base de regras e a superfície do controlador do tipo Mamdani.

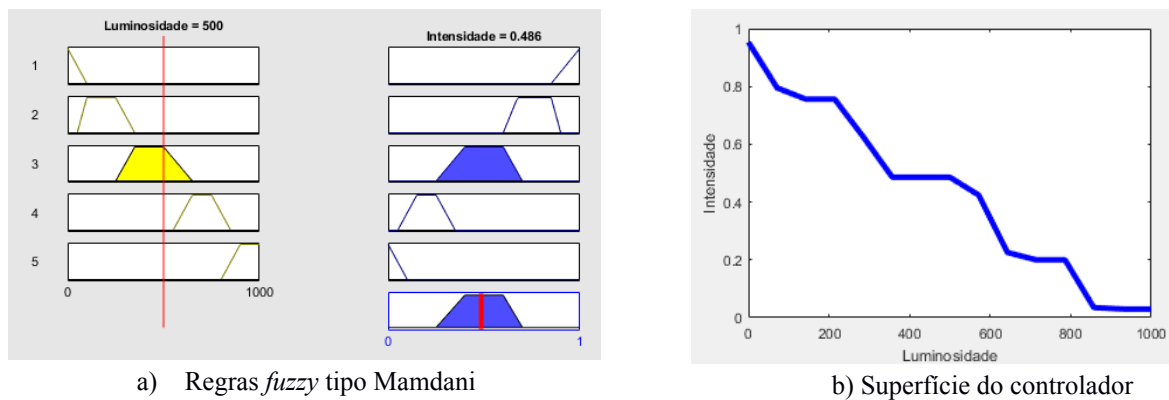


Figura 8. Base de regras e superfície do controlador Mamdani. (Fonte: Autoria própria)

4.3 Descrição do controlador Sugeno

Para o controlador do tipo Sugeno, como já foi explicado anteriormente, as saídas da base de regras são funções lineares ou constantes. Então ao invés de usar funções de pertinências foram definidos os seguintes mostrados na tabela 4 para os valores de saída de intensidade do controlador.

Tabela 4. Valores dos consequentes para a saída “intensidade”. (Fonte: Autoria própria)

Consequentes da base de regras	Valor de pertinência
Muito baixa	0.00
Baixa	0.35
Média	0.50
Alta	0.75
Muito alta	1.00

Na figura 9 a seguir, pode-se observar a base de regras utilizada para o controlador bem com a sua superfície de controle.

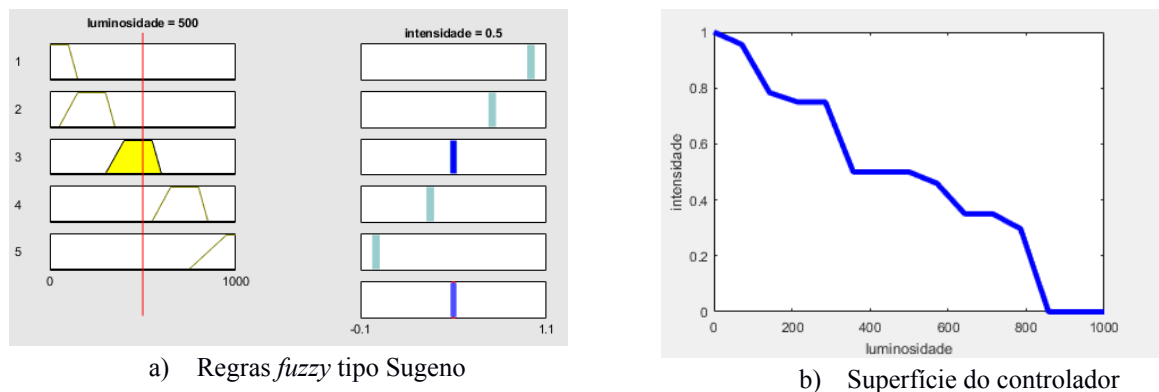


Figura 9. Base de regras e superfície do controlador Sugeno. (Fonte: Autoria própria)

4.4 Circuito de iluminação

O circuito de iluminação é onde todo o *hardware* do projeto está presente. O circuito tem como principal propósito monitorar a luz do ambiente e controlar a intensidade do LED, que está conectado a um resistor de 220Ω para limitar a corrente que o percorre, há também, um potenciômetro que pode ser usado para ajustar a tensão lida pelo nos pino analógico Arduino. Este ajuste permite maior precisão, garantindo que a leitura permaneça na faixa ideal de 0 a 5V. Mantendo assim, a leitura dos pinos analógicos do Arduino mais precisos. O Arduino será utilizado para a interação com o MATLAB, e alguns *jumpers* serão utilizados para as conexões. O LDR fará a leitura na mudança de intensidade da luz ambiente. A Figura 10 apresenta o circuito de iluminação, proporcionando uma visão mais clara de sua estrutura.

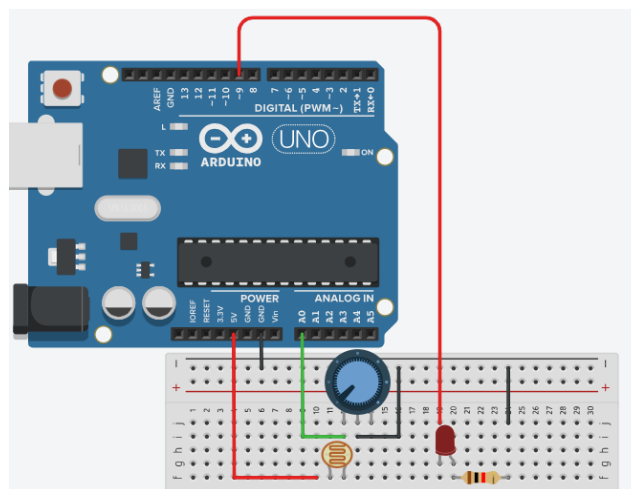


Figura 10. Circuito de iluminação. (Fonte: Autoria própria)

5. ANÁLISE DOS RESULTADOS

A comparação entre as superfícies de controle geradas pelos métodos de inferência *fuzzy* Mamdani e Sugeno é fundamental para compreender as divergências de cada abordagem na modelagem de sistemas de controle de luminosidade. Ambos os métodos utilizam regras *fuzzy* para mapear a entrada (luminosidade) em saídas (intensidade do LED), mas suas representações e resultados podem variar significativamente.

5.1 Gráficos das superfícies de controle

No gráfico mostrado na figura 11, são apresentadas as superfícies de controle resultantes dos dois métodos. A linha azul representa a intensidade calculada pelo método Mamdani, enquanto a linha vermelha representa a intensidade obtida pelo método Sugeno, há também uma linha de referência em preto, em que a mesma

representa uma situação onde o controlador estaria operando de uma forma perfeita. O gráfico das figuras apresentadas nas figuras 8 e 9, foram ajustadas para que ficassem sobrepostas e com uma suavização, como pode ser visto na figura 11.

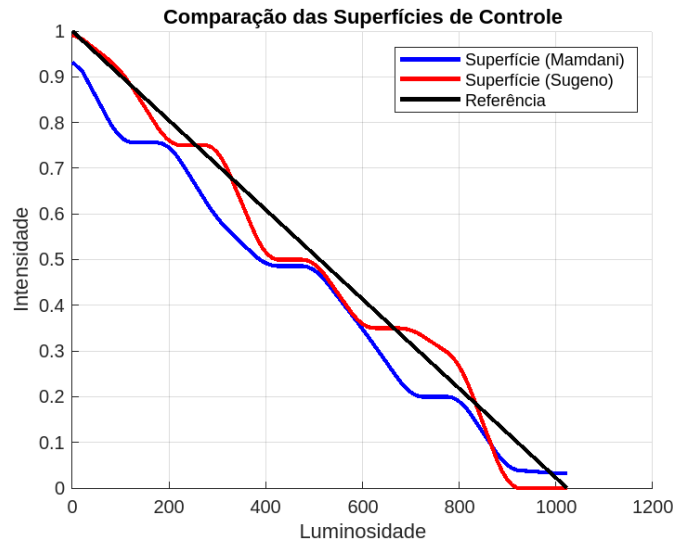


Figura 11. Superfícies dos controladores. (Fonte: Autoria própria)

As superfícies de controle dos controladores Mamdani e Sugeno se aproximam, de certa forma, do cenário ideal, em que o controlador funcionaria de maneira perfeita. A superfície de controle do controlador Mamdani é levemente mais suavizada. Em certos pontos, observa-se que, conforme a luminosidade aumenta em pequenos intervalos, a superfície do controlador Mamdani mantém uma continuidade no valor da intensidade de saída. Esse comportamento se repete por toda a superfície de controle. Outra observação é que, independentemente dos valores de luminosidade, a intensidade de saída não será zero, devido que nesse tipo de controlador, uma superfície de controle é construída a partir de regras *fuzzy* que resultam em uma resposta contínua e suavizada.

Para a superfície do controlador do tipo Sugeno, apresenta um comportamento similar do controlador Mamdani, todavia nessa superfície há uma diminuição mais acentuada na intensidade em resposta ao aumento da luminosidade, resultando em uma transição mais rápida para intensidades mais baixas. Esse comportamento pode ser atribuído ao método no qual manipula as regras e as funções de pertinências, levando a diferentes respostas de saídas. Diferentemente da superfície de controle de Mamdani, a superfície do controlador Sugeno consegue adquirir valores onde o grau de pertinência, é máximo e mínimo, ou seja 1 e 0 respectivamente.

5.2 Diferença percentual entre os controladores

Para compreender melhor a diferença entre os dois controladores, será realizada uma comparação percentual, utilizando o controlador Mamdani como referência. Isso permitirá uma visualização mais clara das diferenças entre ambos. Para se obter o resultado da diferença será usada a equação (3) que se encontra abaixo.

$$D = \frac{V_{Mamdani} - V_{Sugeno}}{V_{Mamdani}} \times 100\% \quad (3)$$

onde:

D – Diferença Percentual.

$V_{Mamdani}$ – Valor da variável fornecido pelo controlador de Mamdani.

V_{Sugeno} – Valor da variável fornecido pelo controlador de Sugeno.

Na tabela 5 encontram-se os resultados da diferença percentual entre os dois controladores e nota-se que a diferença foi bastante significativa. Para os valores estipulados de luminosidade de 100 e 300, a diferença está na casa dos 20%, o que significa que, para o valor estipulado (nesse caso, 300), o método Sugeno está produzindo uma intensidade maior do que aquela calculada pelo método Mamdani. A diferença percentual calculada foi de 27,77%, o que indica que Sugeno gera um valor 27,77% maior do que Mamdani, quando

comparado ao valor de Mamdani como referência. É possível notar também que a menor diferença se deu entre os valores 400 a 600

Para valores entre 700 a 1023 a diferença pode ser de até 100%. Essa diferença significativa pode indicar que o método Sugeno prioriza uma saída mais alta, ou seja, para um mesmo valor estipulado, ele tende a produzir uma intensidade de luz maior. Essa análise fornece uma perspectiva quantitativa sobre a discrepância entre os dois modelos.

Tabela 5. Valores dos consequentes para a saída “intensidade”.
(Fonte: Autoria própria)

Valor Estipulado de luminosidade	Intensidade (Mamdani)	Intensidade (Sugeno)	Diferença Percentual (%)
0	0,953	1	4,93
100	0,756	0,917	21,3
200	0,756	0,75	0,79
300	0,587	0,75	27,77
400	0,486	0,5	2,88
500	0,486	0,5	2,88
600	0,349	0,35	0,29
700	0,2	0,35	75
800	0,2	0,28	40
900	0,0406	0	100
1023	0,0309	0	100

Com os dados obtidos é possível criar um gráfico que é apresentado na figura 12, onde o mesmo mostra bem a diferença entre esses dois controladores utilizando os métodos de Mamdani e Sugeno.

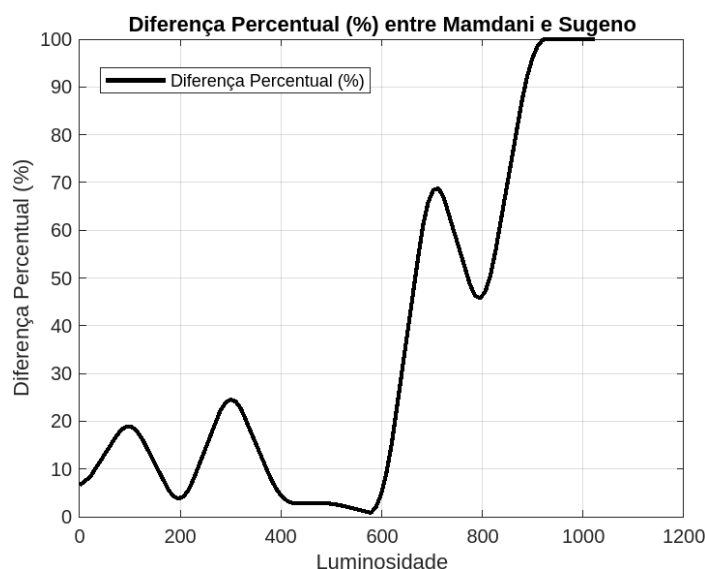


Figura 12. Diferença percentual entre os controladores. (Fonte: Autoria própria)

Observa-se que, para baixos valores luminosidade, a diferença percentual é relativamente baixa em relação a valores mais elevados, o que sugere que ambos os métodos estão convergindo em seus resultados. No entanto, à medida que a luminosidade aumenta, a diferença percentual se torna mais significativa, especialmente em níveis extremos de luminosidade, onde o método Sugeno atinge valores próximos a zero, enquanto o Mamdani mantém valores de intensidade em níveis mais elevados. Por exemplo com um valor de luminosidade igual a 1023, a intensidade no método Mamdani é diferente de zero, já para o método Sugeno o valor da intensidade é igual a zero, ou seja o método Sugeno não está produzindo qualquer intensidade em comparação com o que o Mamdani ainda fornece, mesmo que esse valor seja muito pequeno.

6. CONCLUSÃO

Ao se utilizar a lógica *fuzzy* em controladores, é possível obter um bom desempenho no sistema de controle. A comparação realizada entre os métodos de Mamdani e Sugeno, foi fundamental para mostrar que, apesar de terem estruturas semelhantes, os métodos possuem características diferentes em alguns aspectos, de forma que, a resposta obtida, pode apresentar resultados diferentes entre eles. Após a comparação e com os resultados obtidos, foi possível concluir que, o método de Mamdani apresentou um melhor controle em relação a o método de Sugeno, pois o mesmo apresentou uma melhor resposta para maiores níveis de luminosidade, porém o trabalho também mostrou que em baixo níveis, tanto o controlador do tipo Mamdani quanto o de Sugeno, operam de forma semelhante.

Por meio deste trabalho pode-se concluir que é importante escolher um modelo de inferência adequada para os sistemas de controle. Ao usar o controlador Mamdani foi possível observar um desempenho mais consistente e satisfatório, sendo uma ótima opção para os sistemas de controle. Esse trabalho pode ser aprimorado adicionando novas situações e novas variáveis em que se deseja controlar, também é possível alterar os tipos das funções de pertinência e comparar o desempenho dos sistemas de controles, bem como alterar também a forma de realizar a defuzzificação, observando os seus resultados e fazendo novas comparações que ajudaram a se ter um melhor entendimento a respeito da lógica *fuzzy*, trazendo também novas melhorias nas aplicações dos sistemas de controle mais complexos e dinâmicos.

7. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] SIMÕES, Marcelo Godoy; SHAW, Ian S. Controle e modelagem fuzzy. Editora Blucher, 2007.
- [2] ZADEH, Lotfi A. Fuzzy sets. Information and Control, 1965.
- [3] DE GODOY, Felipe Oliva et al. Utilização da lógica fuzzy aplicada à energia solar. Cadernos de Ciência & Tecnologia, v. 37, n. 2, 2020.
- [4] LOPES, Matheus Maia Holanda; ISONI, Gustavo Braga; MALVEZZI, William Roberto. Controle de luminosidade com a utilização da teoria fuzzy. Programa de Iniciação Científica-PIC/UniCEUB-Relatórios de Pesquisa, v. 4, n. 1, 2018.
- [5] ANDRADE, Michelle; JAQUES, Maria Alice Prudêncio. Estudo comparativo de controladores de Mamdani e Sugeno para controle de tráfego em interseções isoladas. Transportes, v. 16, n. 2, 2008.
- [6] GOMIDE, Fernando; GUDWIN, Ricardo R.; TANSCHKEIT, Ricardo. Conceitos fundamentais da teoria de conjuntos fuzzy, lógica fuzzy e aplicações. In: Proc. 6 th IFSA Congress-Tutorials. 1995.
- [7] GUERRA, Carlos E.; LIMA, Ivnon J. Farias. Um Modelo Matemático Para Filtragem De Perfis De Poço Utilizando Parâmetros Fuzzy Diferenciais. In: IV Simpósio Brasileiro de Geofísica. European Association of Geoscientists & Engineers, 2010. p. cp-197-00146.
- [8] ROCHA, Jhonantans Moraes. Implementação de controle fuzzy em CLP industrial. 2016.
- [9] TANSCHKEIT, Ricardo. Sistemas fuzzy. Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2004.
- [10] AGUIRRE, Luis Antonio. Enciclopédia de automática, v. 3: controle e automação. Editora Blucher, 2007.
- [11] MARRO, Alessandro Assi et al. Lógica fuzzy: conceitos e aplicações. Natal: Universidade Federal do Rio Grande do Norte (UFRN), 2010.
- [12] DE BARROS, Laécio Carvalho; BASSANEZI, Rodney Carlos. Tópicos de lógica fuzzy e biomatemática. Grupo de Biomatemática, Instituto de Matemática, Estatística e Computação Científica (IMECC), Universidade Estadual de Campinas (UNICAMP), 2010.
- [13] DICIEMBRE SANAHUJA, Samuel. Sistemas de control con lógica difusa: Métodos de mamdani y de takagi-sugeno-kang (tsk). 2017.
- [14] MathWorks Fuzzy Logic Toolbox. Disponível em: <https://www.mathworks.com/products/fuzzy-logic.html>. Acesso em: 26 set. 2024.