

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DO TRIÂNGULO MINEIRO – *CAMPUS* PARACATU**

LAURO SILVA ROCHA

**Implementação de controlador *Fuzzy* em CLP de pequeno porte
utilizando lógica Ladder**

**PARACATU - MG
2023**

LAURO SILVA ROCHA

**Implementação de controlador *Fuzzy* em CLP de pequeno porte utilizando
lógica Ladder**

Trabalho de conclusão de curso
apresentado ao Instituto Federal de
Educação, Ciência e Tecnologia do
Triângulo Mineiro, Campus Paracatu,
como requisito parcial para conclusão
do Curso de Bacharelado em
Engenharia Elétrica.

Orientador: Prof. Me. Silas Martins
Sousa.

PARACATU - MG

Ficha Catalográfica elaborada pelo Setor de Referência do IFTM –
Campus Paracatu

R672i Lauro Silva Rocha -
 Implementação de controlador Fuzzy em CLP de pequeno porte
 utilizando lógica Ladder./ Lauro Silva Rocha - 2023.
 42f. : il.

 Orientador: Silas Martins Sousa.
Trabalho de conclusão de curso (graduação) - Instituto Federal de
Educação, Ciência e Tecnologia do Triângulo Mineiro, Curso de
bacharelado em Engenharia Elétrica, Paracatu, 2023.

 1. Controlador Fuzzy. 2. CLP. 3. PLC300. 4. Ladder. I. Silas Martins
Sousa. II. Instituto Federal de Educação, Ciência e Tecnologia do Triângulo
Mineiro - Campus Paracatu. III. Título.

CDD 629.895

Bibliotecária: Nathália de Moraes Torres CRB6-3097

2023

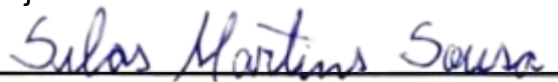
TERMO DE APROVAÇÃO

LAURO SILVA ROCHA

**Implementação de controlador *Fuzzy* em CLP de pequeno porte utilizando
lógica Ladder**

Trabalho de conclusão de curso
apresentado ao Instituto Federal
de Educação, Ciência e
Tecnologia do Triângulo Mineiro,
Campus Paracatu, como
exigência parcial para obtenção
do diploma de Bacharel em
Engenharia Elétrica, sob a
orientação do Prof. Msc. Silas
Martins Sousa.

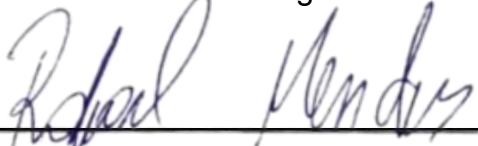
Aprovado em 21 de junho de 2023.



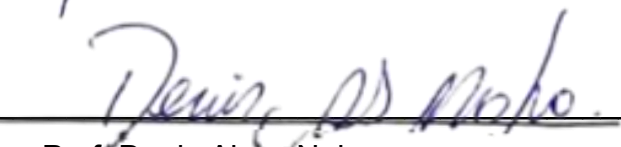
Prof. Msc. Silas Martins Sousa



Prof. Dr. Carlos Renato Borges dos Santos



Prof. Msc. Rafael Mendes Faria



Prof. Denis Alves Nobre

**PARACATU - MG
2023**

DEDICATÓRIA

Dedico este trabalho a toda minha família, especialmente à minha mãe Fernanda Silva Couto que sempre me apoiou e orientou em todas as questões da minha vida, dando muito carinho e amor. Dedico, também, ao meu irmão Lucas Silva Rocha que sempre esteve ao meu lado desde o nascimento e sempre nutriu uma forte amizade comigo. Gostaria de agradecer também a minha avó Maria Nilma Rodrigues Couto e tio Carlos Junior Silva Couto que faleceram durante a realização da minha graduação a quase dois anos e não conseguiram me ver graduado. Dedico também a todos os professores e colegas de classe que me acompanharam na minha jornada acadêmica. Muito Obrigado.

AGRADECIMENTOS

Gostaria de agradecer a toda minha família, que vem me apoiando ao longo dos anos, financeiramente, emocionalmente e psicologicamente. Suas assistências foram primordiais não apenas para a realização deste TCC como de todo o material produzido durante o processo de graduação e durante todo o período de faculdade. Também gostaria de agradecer a todos os professores que fizeram parte da minha trajetória na graduação em Engenharia Elétrica do IFTM Campus Paracatu; os conhecimentos por todos estes transmitidos se mostram muito úteis ao decorrer do curso e para a minha vida pessoal também. Acredito que ao longo deste período na instituição conseguimos formar uma verdadeira amizade. Não esquecendo que ainda tenho muito a aprender e que o processo de aprendizagem é constante e permanente. Mas acredito que as boas orientações e dicas que todos da instituição me proporcionaram acrescentaram muito para meu desenvolvimento pessoal e profissional. Agradeço em participar ao meu professor orientador Silas Martins de Sousa por toda a ajuda no desenvolvimento deste trabalho, sem sua orientação nada do que foi desenvolvido teria sido possível para a conclusão deste. Sua instrução foi muito além das expectativas, tanto técnicas quanto gramaticalmente e empaticamente. Concluo desejando a todos muito sucesso, e que este TCC seja de grande utilidade para aqueles que desejam lê-lo. Atenciosamente.

EPÍGRAFE

"Às vezes na vida você
toma a decisão certa,
outras vezes você torna ela
a certa".

Lauro Silva Rocha.

RESUMO

Atualmente, os sistemas de controle estão presentes em praticamente todas as indústrias. Sua função é realizar o ajuste do sistema para operar em uma condição desejada. A maioria dos sistemas de controle industrial utilizam o controlador PID (Proporcional, Integral, Derivativo). Porém esse sistema é eficaz apenas para sistemas lineares, em sistemas não lineares deve-se buscar por um controlador que seja apropriado para operar com essas não linearidades. Buscando atenuar este problema, o presente trabalho propõe uma alternativa para o controle de sistemas não lineares utilizando controlador Fuzzy, implementado em um CLP (Controlador Lógico Programável) de pequeno porte, utilizando lógica Ladder. O controlador fuzzy se baseia em uma lógica matemática que utiliza um conjunto fuzzy ao invés de conjuntos clássicos. Esse método é dividido em quatro partes: i) a Fuzzificação (onde as entradas são transformadas em conjuntos fuzzy); ii) o conjunto de regras (que é definido por um especialista no processo em questão); iii) o método de inferência (que é o cálculo e processamento fuzzy), que no caso foi utilizado o método de inferência de Mamdani; iv) e a defuzzificação (onde o conjunto fuzzy é transformado em um valor de saída). O resultado final é um valor numérico de ajuste de saída, para ser empregado novamente no sistema ao qual o controlador está inserido. Existem no mercado alguns Softwares que implementam lógica Fuzzy, bem como alguns trabalhos na literatura, porém nenhum deles utiliza a linguagem Ladder, que é a linguagem mais utilizada em CLP's e, além do mais, alguns CLP's só possuem essa linguagem para programação de suas lógicas, principalmente os CLP's de pequeno porte. Este trabalho implementou a lógica do controlador *fuzzy* no CLP da WEG modelo PLC300. Para validar a lógica, foi utilizada a ferramenta do Matlab *Fuzzy Logic Toolbox*. Os resultados mostraram que o controlador Fuzzy implementado no CLP conseguiu gerar valores de saída muito próximos ao resultado do toolbox do matlab, aplicando os mesmos valores de entrada.

Palavras Chave: Controlador Fuzzy; CLP; PLC300; Ladder; Mamdani; Fuzzy Logic Toolbox.

ABSTRACT

Currently, control systems are present in virtually all industries. Its function is to adjust the system to operate in a desired condition. Most industrial control systems use the PID (Proportional, Integral, Derivative) controller. But this system is effective only for linear systems, in non-linear systems one must look for a controller that is appropriate to operate with these non-linearities. Seeking to mitigate this problem, this work proposes an alternative for the control of non-linear systems using a Fuzzy controller, implemented in a small PLC (Programmable Logic Controller), using Ladder logic. The fuzzy controller is based on a mathematical logic that uses a fuzzy set instead of classical sets. This method is divided into four parts: i) Fuzzification (where the inputs are transformed into fuzzy sets); ii) the set of rules (which is defined by an expert in the process in question); iii) the inference method (which is the calculation and fuzzy processing), which in this case was used the Mamdani inference method; iv) and defuzzification (where the fuzzy set is transformed into an output value). The end result is a numerical output adjustment value, to be used again in the system in which the controller is inserted. There are some software on the market that implement fuzzy logic, as well as some works in the literature, but none of them use the Ladder language, which is the most used language in CLP's and, moreover, some CLP's only have this language for programming their logic, mainly the small CLP's. This work implemented the fuzzy controller logic in the WEG CLP model PLC300. To validate the logic, the Matlab tool Fuzzy Logic Toolbox was used. The results showed that the Fuzzy controller implemented in the PLC was able to generate output values very close to the matlab toolbox result, applying the same input values.

Keywords: Fuzzy controller; PLC; PLC300; Ladder; Mamdani; Fuzzy Logic Toolbox.

LISTA DE FIGURAS

Figura 1 - Variável linguística, Termos linguísticos e funções de pertinência. Adaptada de (SILVA, 2011).....	19
Figura 2 - Blocos do sistema do Controlador Fuzzy. Adaptado de (SILVA, 2011).....	19
Figura 3 - Representação gráfica de: a) regras fuzzy, expressões de inferência (Mamdani), b) Inferência Fuzzy pelo método de Mamdani.Adaptado de (SILVA, 2011).....	21
Figura 4 - Processo de Defuzzificação pelo centro de gravidade. Adaptado de (Silva, 2011).....	22
Figura 5- PLC300 WEG (WEG, 2023).....	23
Figura 6 - Fuzzy Toolbox MATLAB (PEDAMKAR, 2015).....	25
Figura 7 - Bloco de controle geral vitalidade das violetas (Fonte: Próprio Autor).....	26
Figura 8 - Gráfico triangular e suas equações subida e descida (Fonte: Próprio Autor).....	27
Figura 9 - Zonas da Função de Pertinência Triangular (Fonte: Próprio Autor)....	27
Figura 10- Programação em Ladder, função triangular (Fonte: Próprio Autor)....	29
Figura 11- Bloco da Fuzzificação de função triangular (Fonte: Próprio Autor).29	
Figura 12 - Função de pertinência para pequena, média e grande das duas variáveis de entrada (SILVA,2011).....	30
Figura 13 - Base de Regras Vitalidade dos Violetas (SILVA, 2011).....	31
Figura 14- Base de Regras em Ladder (Fonte: Próprio Autor).....	31
Figura 15- Programação Ladder mínimo entre 9 regras anteriores e entradas (Fonte: Próprio Autor).....	32
Figura 16- Programação em Ladder máximo entre as 9 regras posteriores (Fonte: Próprio Autor).....	33
Figura 17- Método Centro de Gravidade, divisão em pontos- (Fonte: Próprio Autor).....	33
Figura 19- Programação em Ladder Defuzzificação (Fonte: Próprio Autor)....	35
Figura 20- Equações de Subida e Descida regem as Funções de Pertinência (Fonte: Próprio Autor).....	36
Figura 21 - Fuzzificação em Ladder para entradas de Água e Sol (20,60) (Fonte: Próprio Autor).....	37
Figura 22 - Fuzzificação em Ladder para entradas de Água e Sol (20,60) (Fonte: Próprio Autor).....	37
Figura 23 - Fuzzificação em Ladder para entradas de Água e Sol (35,50) (Fonte: Próprio Autor).....	38
Figura 24 - Fuzzificação em Ladder para entradas de Água e Sol (35,50) (Fonte: Próprio Autor).....	38
Figura 25- Fuzzificação em Ladder para entradas de Água e Sol (50,60) (Fonte: Próprio Autor).....	39

Figura 26- Fuzzificação em Ladder para entradas de Água e Sol (50,60) (Fonte: Próprio Autor).....	39
Figura 27- Teste com CLP300 WEG no laboratório do campus (Fonte: Próprio Autor).....	40

LISTA DE TABELAS

Tabela 1 - Regras de um sistema de tomada de decisão de lavagem de roupa (SILVA, 2011).....	18
Tabela 2 - Base de regras para controladores Fuzzy.....	25

LISTA DE ABREVIATURAS E SIGLAS

CLP	Controlador Lógico Programável
PID	Proportional Integral e Derivativo
PLC	<i>Programmable Logic Controller</i>

SUMÁRIO

1 INTRODUÇÃO.....	15
1.1 Tema.....	16
1.2 Problema.....	16
1.3 Objetivos.....	16
1.3.1 Objetivo geral.....	16
1.3.2 Objetivos específicos.....	17
1.3.3 Hipótese.....	17
1.4 Justificativa.....	17
1.5 Delimitação.....	18
2 DESENVOLVIMENTO.....	18
2.1 Referencial Teórico.....	18
2.1.1 Lógica Fuzzy.....	18
2.1.2 CLP.....	22
2.1.2 PLC300 WEG.....	23
2.1.3 Fuzzy Logic Toolbox.....	25
3 MATERIAIS E MÉTODOS.....	26
4 RESULTADOS E DISCUSSÃO.....	36
5 CONCLUSÃO.....	41
REFERÊNCIAS.....	42

1 INTRODUÇÃO

No momento atual, o ser humano está vivenciando uma era industrial nunca vista anteriormente; com diversas tecnologias sendo formadas a cada ano, tanto as pessoas quanto as indústrias buscam novos meios para inovar e evoluir.

Porém esta nova etapa da história da humanidade transportou consigo novos problemas. Um deles são as dificuldades de automação para pequenas empresas; pois não possuem os mesmos recursos que as empresas maiores para reger seus processos industriais, principalmente se estes possuem características de não linearidade. Reconhecendo este problema, o objetivo deste trabalho é propor uma lógica matemática Fuzzy para ser implementada em CLPs de pequeno porte, auxiliando estas em seus processos automatizados.

Os CLPs são controladores, o que significa que são responsáveis pelas tomadas de decisões do sistema. Tendo sido criado por Morley (1968), esses dispositivos realizam o gerenciamento dos dados em uma planta industrial, e se necessário, realizam o ajuste de uma variável de saída.

A lógica matemática fuzzy, criada por Lotfi Asker Zadeh (1965), é um método matemático que trabalha com o princípio da incerteza, ou seja, quantificar matematicamente quando uma variável se qualifica em determinado conjunto ou não. Geralmente os trabalhos relacionados à lógica fuzzy em pequena escala remetem-se à área da matemática. Porém, esta lógica possui um grande potencial de crescimento.

“A Lógica Fuzzy vem aumentando sua aplicação na área de sistemas de controle, sendo utilizada, inclusive em Rastreadores do Ponto de Máxima frequência (MPPT – Maximum Power Point Tracker) de sistemas fotovoltaicos” (MOÇAMBIQUE, 2012,p. 1).

O método de construção lógica fuzzy deste trabalho utiliza a programação em Ladder. Nesta linguagem de programação primeiramente é usado para a construção dos gráficos triangulares, que representam as variáveis de entrada, que passam por um processador de entrada, enviado para uma máquina de inferência que funciona baseado em uma base de regras e finalmente é realizada a defuzzificação.

1.1 Tema

A Lógica matemática Fuzzy ou Lógica Difusa é uma ferramenta matemática utilizada para quantificar e modelar incertezas. Através de seu uso é possível quantificar a probabilidade de quando uma situação não se qualifica inteiramente como sendo A ou B é propenso a ser alguma dessas opções.

1.2 Problema

A maioria dos sistemas de controle industriais utiliza o sistema PID para realizar as suas ações automatizadas. Porém, em alguns casos, esse sistema não pode ser aplicado. Como o PID é construído apenas para os sistemas lineares é necessário utilizar outros meios quando precisamos controlar sistemas não lineares. Assim o problema designado é realizar a controle de sistemas não lineares de pequeno porte.

1.3 Objetivos

1.3.1 Objetivo geral

Desenvolver um bloco implementando a Lógica Fuzzy no PLC300 WEG, verificando a eficiência da implementação da Lógica Fuzzy em PLCs de pequeno porte.

1.3.2 Objetivos específicos

- Desenvolver as funções de pertinência das entradas e saídas em Ladder.
- Desenvolver Lógica Fuzzy em um Bloco em Ladder.
- Implementar o Bloco Fuzzy no CLP300 WEG.
- Realizar o processo de Fuzzificação para a Vitalidade das Violetas.
- Adaptar esta lógica para sistemas de variáveis físicas: temperatura, pressão, vazão, etc.

1.3.3 Hipótese

Pressupõe-se que a lógica Fuzzy pode ser aplicada em um CLP de pequeno porte, ou seja, CLP's com limitações de instruções lógicas, utilizando a lógica Ladder.

1.4 Justificativa

Como já citado anteriormente o principal benefício deste trabalho de pesquisa é criação de uma Lógica Fuzzy para PLCs de pequeno porte, sendo podendo estes serem empregados nas em plantas industriais de pequeno e médio porte. Assim beneficiando a indústria, e também ajudando na agregação de conhecimento de alunos e professores que podem utilizar desta lógica em estudos de PLCs industriais uma vez que este Trabalho de Pesquisa torna o emprego de lógica Fuzzy mais acessível.

1.5 Delimitação

O controlador Fuzzy foi desenvolvido e testado no CLP de pequeno porte modelo PLC300 da WEG. Controladores Fuzzy podem ser implementados com qualquer número de entradas e saídas, porém optou-se por utilizar duas entradas e uma saída.

2 DESENVOLVIMENTO

2.1 Referencial Teórico

2.1.1 Lógica Fuzzy

Na teoria dos conjuntos clássicos um número pertence ou não àquele conjunto. Nos conjuntos fuzzy (chamados também de difuso ou nebuloso) um número pertence ao conjunto com certo grau de pertinência. As principais operações entre conjuntos *fuzzy* são a união (o mesmo que obter o máximo de cada sinal ponto a ponto) e a interseção (o mesmo que obter o valor mínimo entre as entradas).

“As variáveis linguísticas fuzzy (ex.: Temperatura) podem ser expressas na forma de termos linguísticos como: “Baixa”, “Média” e “Alta”. As funções de pertinência também podem ser de vários formatos, como triangular, trapezoidal e gaussiana” (SILVA, 2011).

A Figura 1 mostra as funções de pertinência e também as variáveis e termos linguísticos.

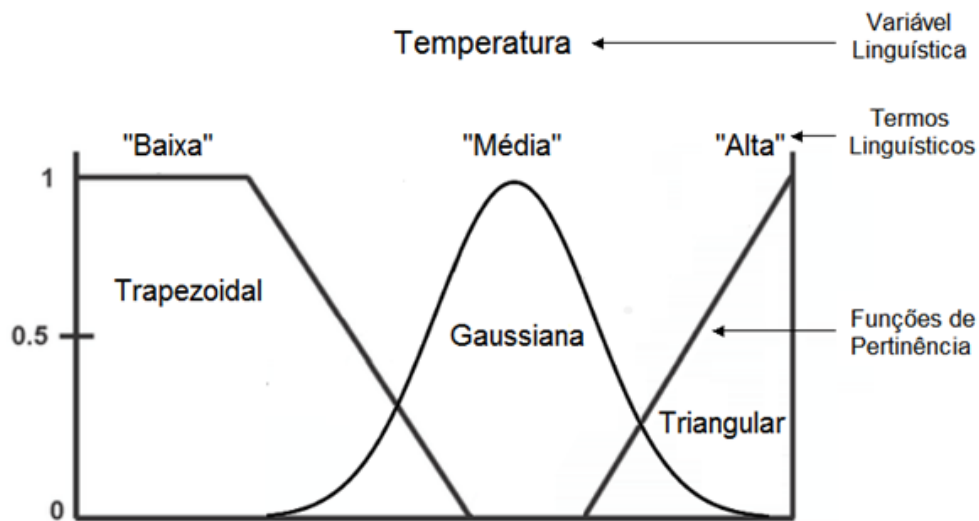


Figura 1 - Variável linguística, Termos linguísticos e funções de pertinência. Adaptada de (SILVA, 2011).

O sistema completo do Controlador Fuzzy é composto por quatro partes: Fuzzificação, Máquina de Inferência, Base de Regras e a Defuzzificação, como mostrado na figura 2.

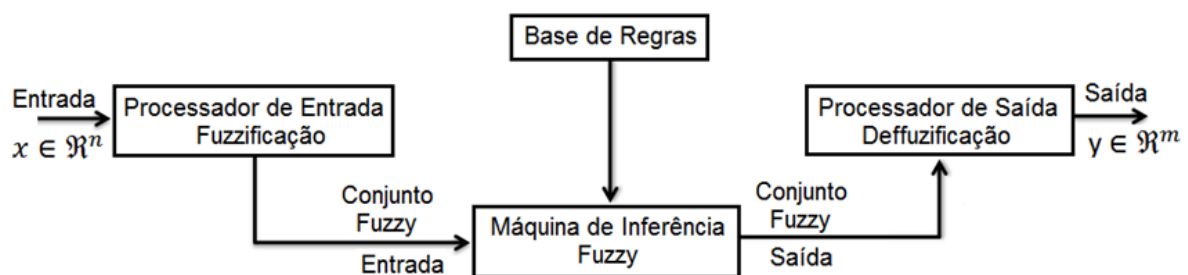


Figura 2 - Blocos do sistema do Controlador Fuzzy. Adaptado de (SILVA, 2011).

A Fuzzificação consiste em modelar o problema real nos conjuntos fuzzy, assim, surge a necessidade de um especialista no processo em questão para projetar os seguintes parâmetros: regras fuzzy, variáveis linguísticas, termos linguísticos e funções de pertinência.

Existem três métodos de inferência: Mamdani, Takagi-Sugeno (T-S) e Tsukamoto, sendo os dois primeiros mais conhecidos. O método de Mamdani é baseado na composição max-min, onde obtém-se o mínimo das entradas

(antecedentes) para cada regra fuzzy e depois é obtido a saída (consequente) através do resultado máximo. A diferença básica entre os métodos de Mamdani e Takagi-Sugeno estão na forma de escrever o consequente (saída) de cada regra e no processo de Defuzzificação. O método de Takagi-Sugeno utiliza qualquer tipo de função no consequente, já o de Mamdani apenas funções pré determinadas, como triangular, gaussiana e trapezoidal (SILVA, 2011).

Um exemplo de regras fuzzy do método de Mamdani é mostrado na Tabela 1, em que é utilizado a lógica Se – Então, ou seja, caso aconteça uma determinada condição, uma ação (consequência) será realizada.

A Tabela 1 ilustra um conjunto de regras utilizando termos linguísticos, mas essas regras são aplicadas a funções de pertinência do tipo triangular, trapezoidal e gaussiana, as quais podem ser configuradas (maiores ou menores) em função da não linearidade do sistema a ser controlado.

Tabela 1 - Regras de um sistema de tomada de decisão de lavagem de roupa (SILVA, 2011).

<p>R_1: Se a roupa é “grossa” e a sujeira é “difícil”</p> <p>então lava-se “muito tempo”.</p> <p>R_2: Se a roupa é “grossa” e a sujeira é “fácil”</p> <p>então lava-se “em tempo médio”.</p> <p>R_3: Se a roupa é “fina” e a sujeira é “fácil”</p> <p>então lava-se “pouco tempo”.</p> <p>R_4: Se a roupa é “fina” e a sujeira é “difícil”</p> <p>então lava-se “pouco tempo”.</p>
--

A Figura 3 mostra um sistema de controle fuzzy com duas entradas (A_1 e B_1), com duas regras (R_1 e R_2). Na Figura 2(a) é feito o processo de Fuzzificação, onde em cada regra é analisado o valor mínimo de cada entrada, obtendo as funções triangulares de respostas parciais C_1 e C_2 , respectivamente. A Figura 2(b) mostra como é realizada a inferência por meio do método de Mamdani, em que se utiliza os

valores máximos entre os resultados parciais de C_1 e C_2 , ponto a ponto, para gerar a função triangular C .

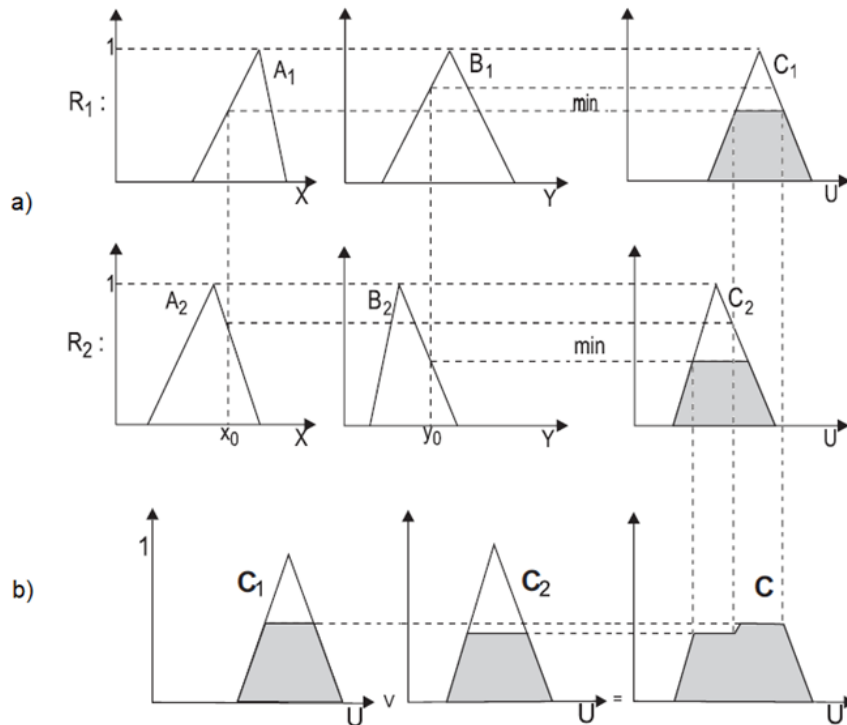


Figura 3 - Representação gráfica de: a) regras fuzzy, expressões de inferência (Mamdani), b) Inferência Fuzzy pelo método de Mamdani. Adaptado de (SILVA, 2011).

Para obter um valor final, que representa a saída da lógica fuzzy, é feita Defuzzificação, que consiste em achar o centro de gravidade (ou ainda o centróide).

“ele é semelhante à média ponderada para a distribuição de dados, com a diferença que os pesos são os valores $\varphi C(u_i)$, que indicam o grau de compatibilidade do valor u_i com o conceito modelado pelo conjunto fuzzy C . O centro de gravidade da média das áreas de todas as figuras que representam os graus de pertinência de um subconjunto fuzzy”. (SILVA, 2011, p. 26)

Como é possível deduzir através da citação acima, o método do centro de gravidade de busca encontra o valor médio do gráfico ao analisar a área e formato da sua figura. Como demonstrado na figura 4.

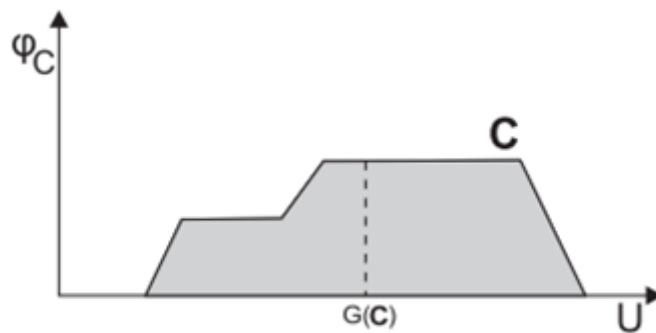


Figura 4 - Processo de Defuzzificação pelo centro de gravidade. Adaptado de (Silva, 2011).

2.1.2 CLP

O CLP é um controlador responsável pelo acionamento de vários dispositivos de um processo industrial, sendo este muito utilizado nas indústrias atualmente e podendo realizar diversas ações.

“Para quem não conhece o CLP ele é sem dúvidas a parte de inteligência de um processo de automação, ou seja, ele é responsável pelo controle de todo o processo. CLP é a sigla para Controlador Lógico Programável, que é um computador capaz de executar funções específicas através de programas criados. Embora seja considerado um computador, não confunda CLP com os desktops e notebooks que são computadores de uso geral, utilizados em casa ou no trabalho por exemplo!” (Mattele, 2006.p 1).

O CLP pode ser considerado um computador, porém se comparado a um computador convencional ele possui menor capacidade de processamento de dados. A sigla CLP vem da sigla em inglês PLC (*Programmable Logic Controller*) em tradução livre significa Controlador Lógico Programável. Mas ambas se referem ao mesmo equipamento, não havendo diferença entre PLC e CLP.

Todos os CLPs possuem algumas características em comuns, como: CPU, Memória, Fonte de Alimentação, Módulos de entrada e saída.

A CPU (*Central Processing Unit*) é responsável por buscar, interpretar e

executar as instruções da programação.

A memória é responsável por armazenar as informações, a CPU possui estas voláteis e não voláteis. As memórias voláteis perdem as informações armazenadas quando o PLC é desligado, já as não voláteis mantêm essas informações mesmo após o desligamento.

A função dos módulos de entrada e saída é a conexão do CLP com os sensores e atuadores. Essas entradas e saídas podem ser tanto analógica quanto digitais, dependendo do CLP, acionando dispositivos como :

- Sensores fim de curso;
- Termopares;
- Termostatos;
- Pressostatos;
- Fluxostatos;
- Contatores;
- Atuadores eletropneumáticos;
- Atuadores eletro-hidráulicos.

2.1.2 PLC300 WEG



Figura 5- PLC300 WEG (WEG, 2023).

O Modelo utilizado neste trabalho é o PLC300 da fabricante de motores e equipamentos elétricos WEG, caracterizam-se como um CLP de pequeno porte, com diversas funções embutidas. Entre elas:

- 10 entradas digitais e 1 analógica incorporada, 9 saídas digitais (sendo 1 rápida) e 1 analógica incorporada;
- Possibilidade de expansão de I/O digital e analógico via rede CANOpen; Módulos de expansão I/O do CFW11;
- Entrada de encoder (100Khz) incorporada;
- Interface de comunicação: Modbus-RTU (Mestre), CANOpen (Mestre);
- Até 240 telas programadas pelo usuário;
- Programado em linguagem ladder através do software gratuito WPS (WEG Programming Suite);
- IHM frontal e possibilidade de customização da membrana (disponível na versão com IHM incorporada – modelos HPC e HSC);
- Display traseiro para versão fundo de painel (modelos BP e BS);
- Porta USB incorporada para comunicação com o computador;
- Memória flash interna; Monitoramento da tensão da bateria; Porta Ethernet Modbus-TCP; Entrada para cartão de memória (tipo SD Card);
- Comunicação com a linha PWS de IHMs gráficas (3,3 a 15 pol).

Além de todas essas características, o PLC 300 WEG foi escolhido por possuir ótima flexibilidade e desempenho. Sua flexibilidade se deve ao fato dele ser um dos melhores custo benefícios do mercado e ser expansível, o que permite que possa ser customizado e aprimorado caso seja necessário. Seu desempenho se deve ao fato dele ser compacto e possuir alta velocidade de processamento e grande quantidade de memória, além de possuir 5 portas de comunicação incorporadas, possibilitando que o PLC300 seja mestre de redes como CANopen, Modbus-RTU e Ethernet Modbus-TCP/IP.

2.1.3 Fuzzy Logic Toolbox

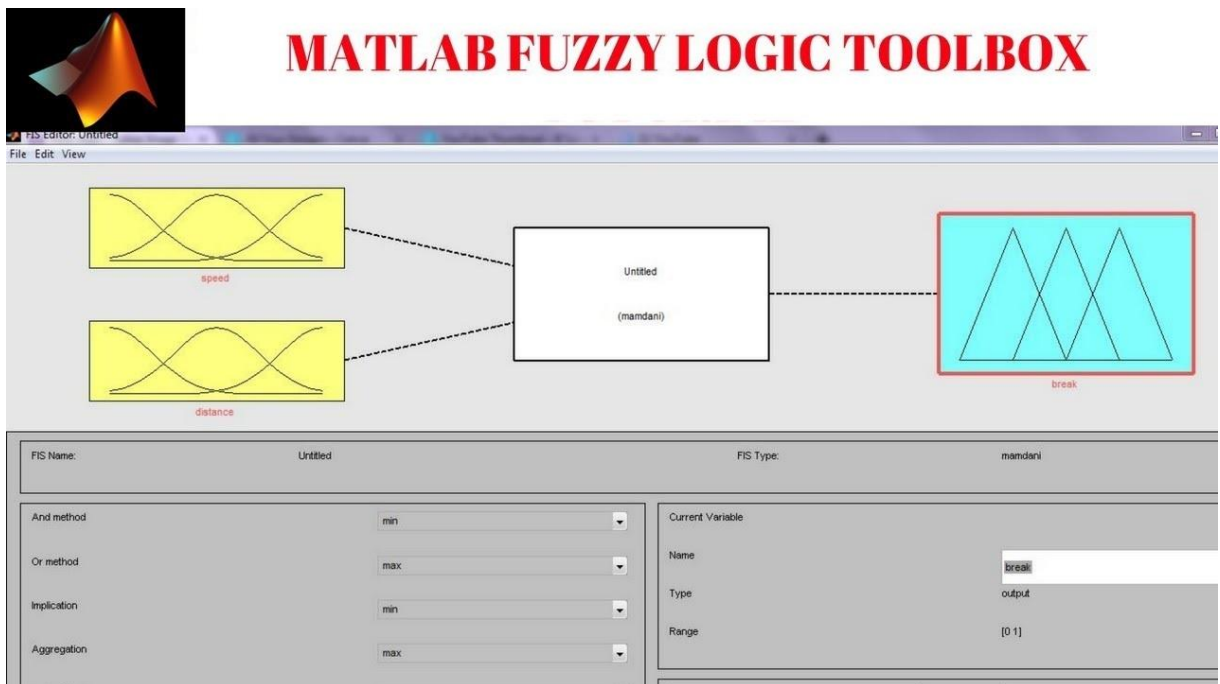


Figura 6 - Fuzzy Toolbox MATLAB (PEDAMKAR, 2015).

O *Matrix Laboratory*, mais conhecido como Matlab, é um software de cálculo e expressão gráfica muito utilizado pelos engenheiros em diversos países. Seu uso permite a construção de tabelas, gráficos e matrizes. Para fins de comparação e verificação dos resultados deste trabalho utiliza-se a ferramenta *Fuzzy Logic Toolbox* para conseguirmos observar o desenvolvimento da lógica fuzzy.

Para usar essa ferramenta digita-se a palavra “fuzzy” no painel central do Matlab. Em seguida, irá abrir uma janela para esta ferramenta, solicitando as características das variáveis de entrada, como as funções de pertinência, os termos linguísticos e as variáveis linguísticas.

O *Fuzzy Logic Toolbox* determina a saída processando as variáveis de entrada utilizando o sistema de Mamdani, mesmo sistema utilizado no lógica Ladder deste trabalho.

3 MATERIAIS E MÉTODOS

O PLC300 WEG foi submetido a uma programação que realiza a lógica *Fuzzy* através da programação em linguagem Ladder. Para realização desta lógica o programa foi dividido em diversas partes até sua conclusão.

No trabalho em questão foi desenvolvido para realizar a lógica *Fuzzy* aplicada ao controle da vitalidade das violetas. Foi desenvolvido sobre esse tema devido aos estudos de um trabalho anterior (SILVA, 2011) sobre a lógica *Fuzzy* aplicado a esse tema. Utilizando este trabalho como referencial teórico foi desenvolvido a lógica existente em Ladder.

Definido a programação como o processo de controle da a vitalidade de uma planta tem-se como entrada do processo: Quantidade de ÁGUA e Quantidade de SOL. Como demonstrado na figura 7.

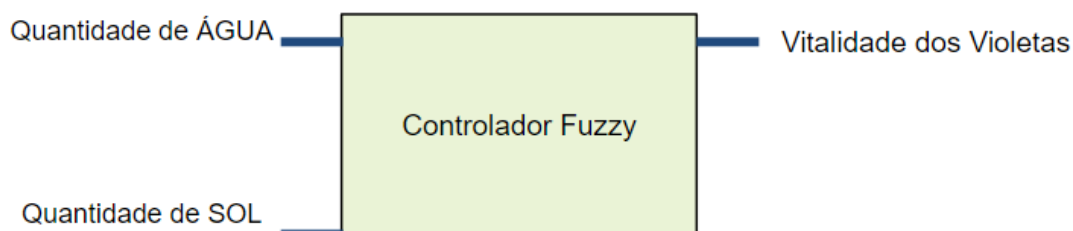


Figura 7 - Bloco de controle geral vitalidade das violetas (Fonte: Próprio Autor).

Na primeira parte do desenvolvimento da Lógica Ladder foi a programação da Função de Pertinência, que é uma forma geométrica triangular. Como mostrado na figura 7. Para ser representado esta etapa foi desenvolvido a sua forma de subida e descida.

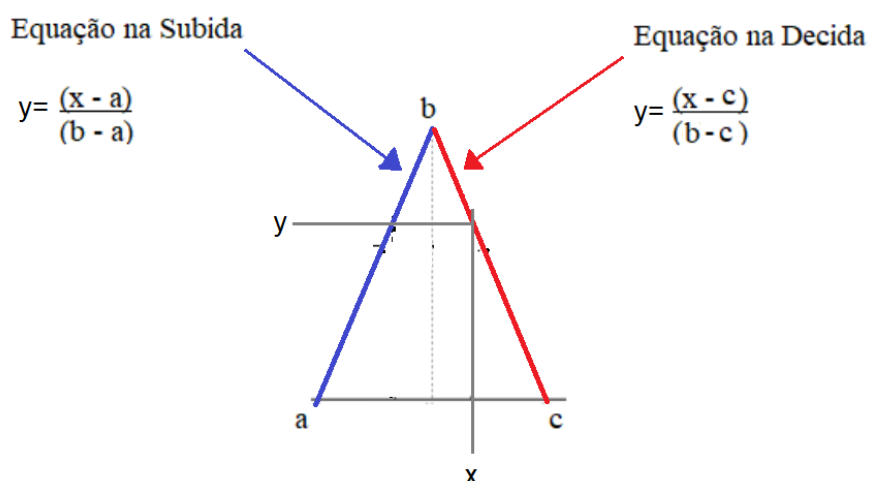


Figura 8 - Gráfico triangular e suas equações subida e descida (Fonte: Próprio Autor).

Para simplificar o desenvolvimento a função de pertinência foi dividida em quatro zonas: Zona I, representando as entradas para valores antes do ponto A; Zona II, representando os valores de entrada entre os pontos A e B; Zona III, representando os valores de entrada entre os pontos B e C; e a Zona IV representando os valores de entrada maiores que C. Como ilustrado na figura 8.

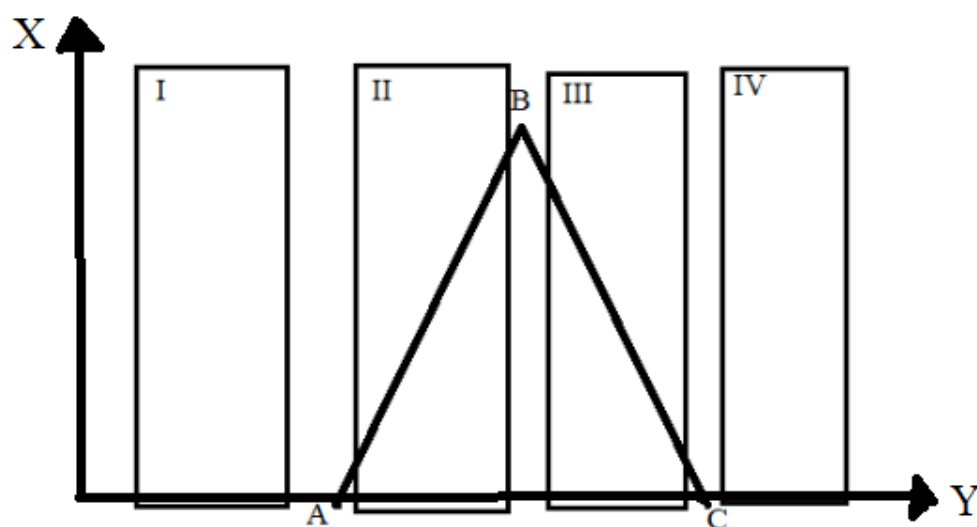


Figura 9 - Zonas da Função de Pertinência Triangular (Fonte: Próprio Autor).

A primeira linha da Figura 10 demonstra o desenvolvimento em Ladder da Zona I. Primeiramente é feita a comparação se a entrada (X) é menor que A , utilizando o bloco `COMP_LE`. Caso esta situação demonstre-se verdadeira, utiliza-se o bloco `MULT` para atribuir valor 0 a variável Y , multiplicando 0 e 1 e atribuindo na

saída, no caso Y.

A segunda linha da Figura 10 demonstra o desenvolvimento em Ladder da Zona II. Primeiramente é feita a comparação se a entrada (X) é maior que A, e menor que B, utilizando o bloco COMP_GT para X e A, e o bloco COMP_LE para X e B. Caso esta situação demonstre-se verdadeira, utiliza-se o bloco SUB para realizar a operação $(X-A)$ que é numerador da fórmula de subida. Também utiliza-se o bloco SUB para realizar a operação $(B-A)$ denominador da fórmula de subida. Por fim, é utilizado DIV para fazer a divisão entre numerador e denominador, a resposta é atribuída a variável Y.

A terceira linha da Figura 10 demonstra o desenvolvimento em Ladder da Zona III. Primeiramente é feita a comparação se a entrada (X) é maior que B, e menor que C, utilizando o bloco COMP_GT para X e B, e o bloco COMP_LE para X e C. Caso esta situação demonstre-se verdadeira, utiliza-se o bloco SUB para realizar a operação $(X-C)$ que é numerador da fórmula de descida. Também utiliza-se o bloco SUB para realizar a operação $(B-C)$ denominador da fórmula de descida. Por fim, é utilizado DIV para fazer a divisão entre numerador e denominador, a resposta é atribuída a variável Y.

A quarta linha da Figura 10 demonstra o desenvolvimento em Ladder da Zona IV. Primeiramente é feita a comparação se a entrada (X) é maior que C, utilizando o bloco COMP_GE. Caso esta situação demonstre-se verdadeira, utiliza-se o bloco MULT para atribuir valor 0 a variável Y, multiplicando 0 e 1 e atribuindo na saída, no caso Y.

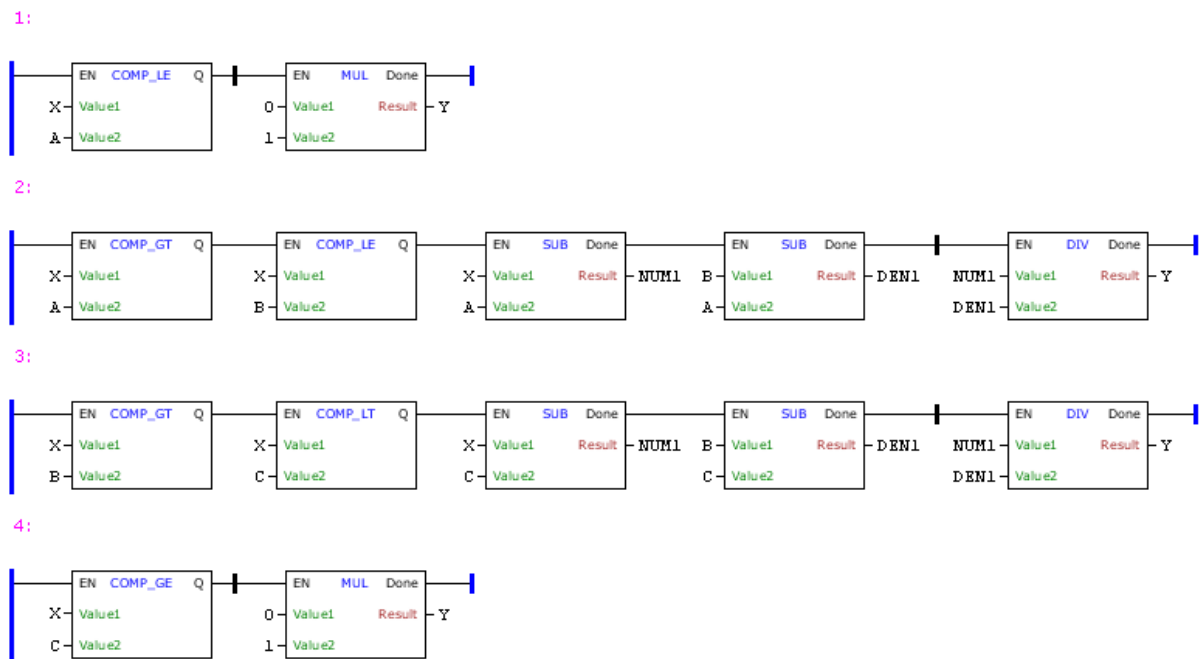


Figura 10- Programação em Ladder, função triangular (Fonte: Próprio Autor).

Em seguida toda essa lógica foi encapsulada em um bloco para ser usado quando necessário, como mostra a figura 11.

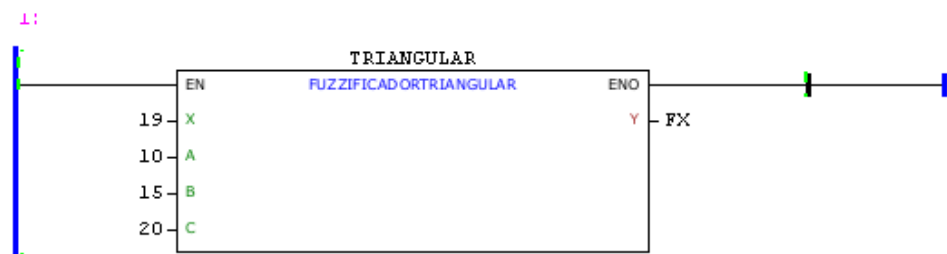


Figura 11- Bloco da Fuzzificação de função triangular (Fonte: Próprio Autor).

É utilizado esta função triangular para criar as 6 variáveis de entrada:

- Quantidade de água pequena;
- Quantidade de água média;
- Quantidade de água grande;
- Quantidade de sol pequena;
- Quantidade de sol média;
- Quantidade de sol grande.

Assim, pode-se expressar as funções de pertinência de entrada, como mostrado na figura 12.

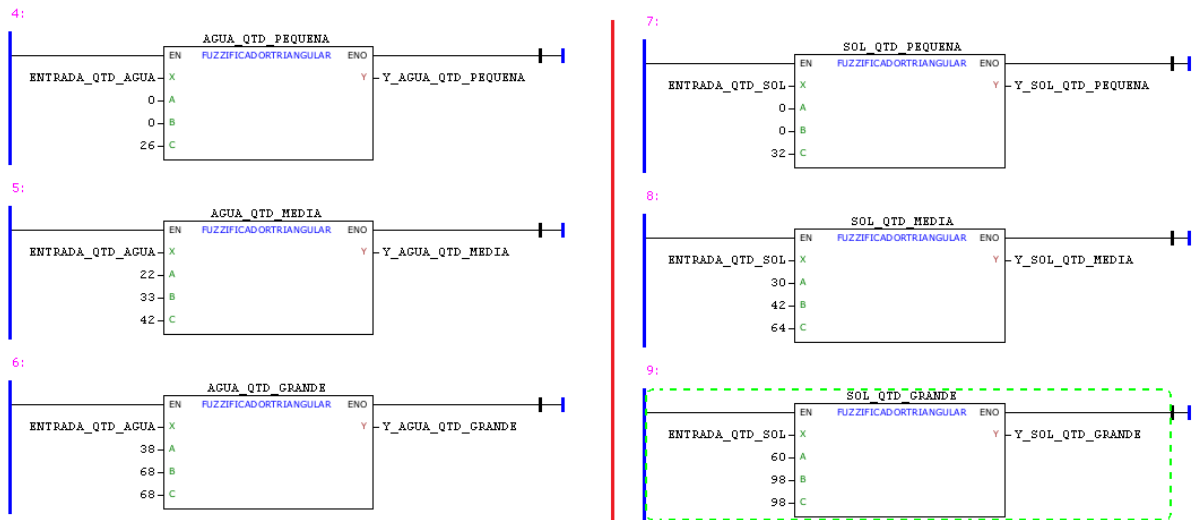


Figura 12 - Função de pertinência para pequena, média e grande das duas variáveis de entrada (SILVA,2011).

Na segunda parte definimos a base de regras ou conjunto de regras. Foi utilizado o mesmo conjunto de regras de SILVA (2011). Sendo este expresso na tabela 2 e Figura 13.

Tabela 2 - Base de regras para controladores Fuzzy.

Coluna 1: Quantidade de Água.	Coluna 2: Quantidade de Sol.	Coluna 3: Vitalidade das Violetas
Pouca: (1)	Pouca: (1)	Pequena: (1)
Média: (2)	Média: (2)	Média: (2)
Muita: (3)	Muita: (3)	Alta: (3)

A tabela 2 mostra a identificação de cada função de pertinência das entradas e da saída. Esses números de identificação são usados para compor a base de regras, conforme mostrado na figura 13.

$$\text{REGRAS} = \begin{pmatrix} 1 & 1 & 2 \\ 1 & 2 & 2 \\ 1 & 3 & 1 \\ 2 & 1 & 3 \\ 2 & 2 & 3 \\ 2 & 3 & 2 \\ 3 & 1 & 1 \\ 3 & 2 & 1 \\ 3 & 3 & 1 \end{pmatrix}.$$

Figura 13 - Base de Regras Vitalidade dos Violetas (SILVA, 2011).

A representação da base de regras em Ladder está presente na figura 14. Observa-se no campo superior esquerdo da figura 14, na linha 10 do programa Ladder é feito através do bloco de comando MIN, o mínimo entre: Quantidade de água pequena e Quantidade de sol pequena. O resultado dessa operação é guardado na variável MIN_REGRA_1. Assim é desenvolvido a primeira etapa de construção da linha 1 da base de regras. O mesmo procedimento é repetido para as outras linhas da base de regras.

10: Fazendo o MIN entre as regras (só para as duas entradas)



11: Fazendo o MIN entre as regras (só para as duas entradas)



12: Fazendo o MIN entre as regras (só para as duas entradas)



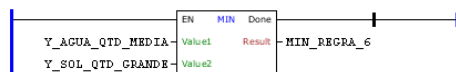
13: Fazendo o MIN entre as regras (só para as duas entradas)



14: Fazendo o MIN entre as regras (só para as duas entradas)



15: Fazendo o MIN entre as regras (só para as duas entradas)



16: Fazendo o MIN entre as regras (só para as duas entradas)



17: Fazendo o MIN entre as regras (só para as duas entradas)



18: Fazendo o MIN entre as regras (só para as duas entradas)



Figura 14- Base de Regras em Ladder (Fonte: Próprio Autor).

Então realiza-se o mínimo, através do bloco de comando MIN, entre o MIN_REGRA_1 e Vitalidade Média das violetas, observado no canto superior esquerdo da figura 15. O resultado é armazenado na variável R1_REGRA_1, este processo é a segunda e última etapa da linha 1 da base de regras. Esse processo é repetido para as outras 8 linhas .

6: 5) Calcular o mínimo entre o valor da saída (conforme a regra) e (mínimo



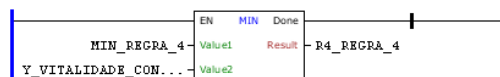
7: 5) R2



8: 5) R3



9: 5) R4



10: 5) R5



11: 5) R6



12: 5) R7



13: 5) R8



14: 5) R9



Figura 15- Programação Ladder mínimo entre 9 regras anteriores e entradas (Fonte: Próprio Autor).

A variável R1_REGRA_1 representa a primeira linha da base de regras, figura 13. No qual 1-1-2 é mínimo entre Quantidade de Água Pouca, Quantidade de Sol Pouca e Vitalidade Média.

A terceira parte do programa é o processamento de dados utilizando o método de Mamdani, demonstrado anteriormente na figura 3. É realizado o máximo das 9 regras, observado na figura 16. Primeiramente é realizado o máximo entre a regra 1 e regra 2, com o bloco MAX tendo como entrada R1_REGRA_1 e R2_REGRA_2, a maior resultante é guardado na variável AUX1 para ser usada na continuação do processo, é realizada o máximo de AUX1 com a próxima regra R3_REGRA_3. E assim por diante até que sejam concluído o máximo entre todas as

9 regras; detendo o valor final na variável MAX_REGRA.

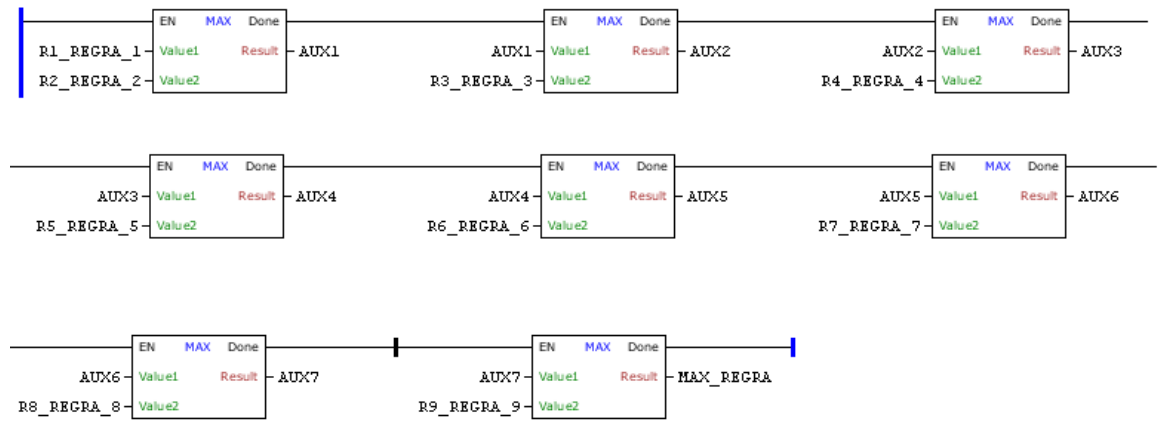


Figura 16- Programação em Ladder máximo entre as 9 regras posteriores (Fonte: Próprio Autor).

A saída é expresso em vetor que representa um gráfico, sendo necessário realizar uma defuzzificação para obter um valor absoluto. Neste trabalho é utilizado o método pelo centro de gravidade, ou centróide, demonstrando anteriormente na figura 4.

Para aplicar o método de centro de gravidade é necessário primeiramente dividir a figura em vários pontos, como mostrado na figura 17. Neste trabalho o sistema dividiu a figura foi dividida em 100 pontos.

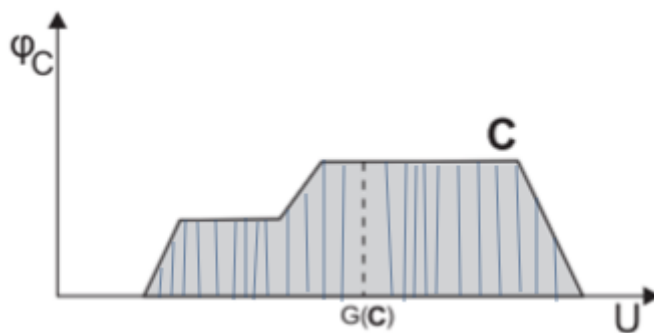


Figura 17- Método Centro de Gravidade, divisão em pontos- (Fonte: Próprio Autor).

Depois é necessário realizar a soma dos pontos e então realizar a multiplicação por uma constante. O resultado deve ser dividido pela somatória dos pontos. Como mostrado na figura 18.

$$G(C) = \frac{\int_R u f_{pC}(u) du}{\int_R f_{pC}(u) du}.$$

Figura 18- Fórmula Centro de Gravidade. Adaptada de (SILVA, 2011).

Para realizar a construção em Ladder da defuzzificação o primeiro passo foi a construção de uma condição para captar os 100 valores do vetor que representa a figura. Para isso foi utilizado o bloco COMP_LT, com variáveis de entrada CONT_PONTOS e 400. O CONT_PONTOS é uma variável auxiliar que verifica a quantidade de vezes que o CLP compilou, ela inicia no valor 0 e a cada ciclo de compilação do CLP é acrescentado 4 em seu valor. o bloco COMP_LT verifica se a variável CONT_PONTOS é menor que 400 ou seja se, se ainda está sendo realizado os 100 ciclos.

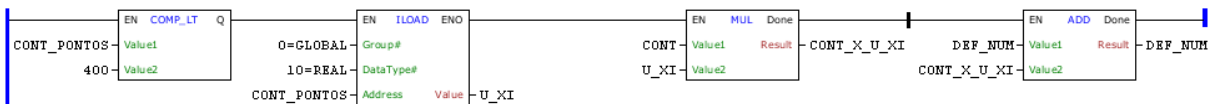
O bloco COMP_LT é utilizado nas primeiras 2 linhas do defuzzificador, como mostrado no início das linha 17,18 da figura 19. Como condição para captura dos pontos do gráfico e utilização da fórmula do centro de gravidade.

A linha 17 da figura 19 refere-se à construção do numerador da fórmula do centro de gravidade. Utilizando a bloco ILOAD para a criação do vetor de pontos, o bloco MULT executa a condição de multiplica a variável CONT por U_XT, criando assim o numerador do centro de gravidade. Ao final, soma-se o valor do numerador centro de gravidade anterior com o novo utilizando o bloco ADD.

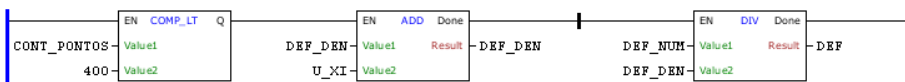
A linha 18 da figura 19 refere-se à construção do denominador da fórmula do centro de gravidade. Utilizando a bloco ADD para somar a denominador centro de gravidade anterior com o novo. O resultado é usado na divisão entre numerador e denominador com o bloco DIV.

A linha 19 da figura 19 refere-se a parte final da defuzzificação, bloco COMP_GE realiza uma comparação, verificando se a variável CONT_PONTOS é maior que 400, ou seja, se já foram contabilizados os cálculos de defuzzificação para os 100 pontos. Caso esta afirmativa seja verdadeira, é realizado no bloco MUL a atribuição do valor final da defuzzificação a variável DEF_FINAL.

17: Deffuzificação pelo Método de Centro de Gravidade (NUMERADOR)



18: Deffuzificação pelo Método de Centro de Gravidade (DENOMINADOR)..... e NUMERADOR/DENOMINADOR



19: Deffuzificação: Pegando apenas o valor final de DEF (que é quando CONT=400)

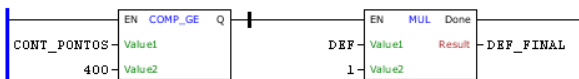


Figura 19- Programação em Ladder Defuzzificação (Fonte: Próprio Autor).

4 RESULTADOS E DISCUSSÃO

O primeiro resultado obtido neste trabalho foi a obtenção em Ladder de uma lógica para a criação das variáveis de entrada como gráficos triangulares. A figura 20 representa a função de pertinência na subida e na descida com exemplo.

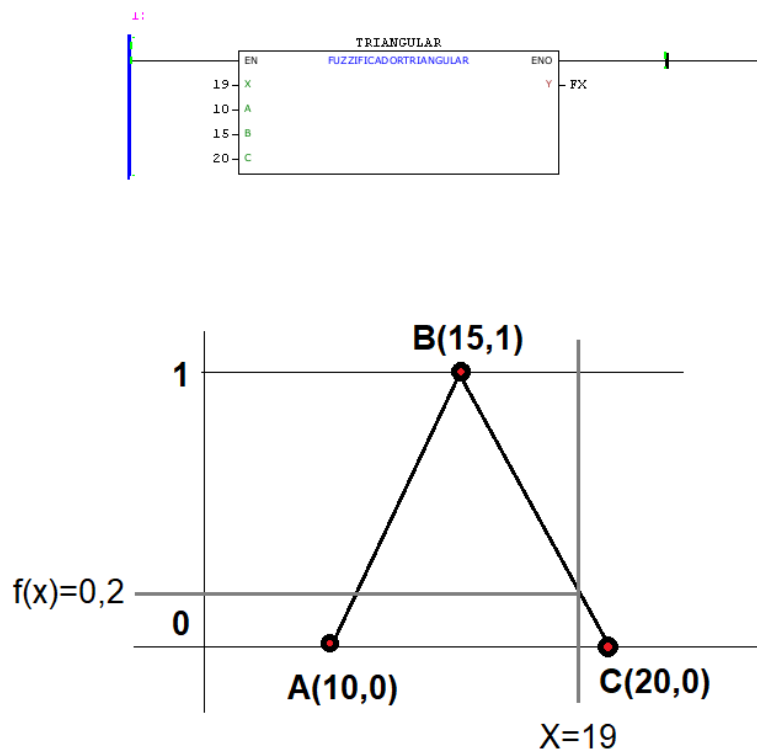


Figura 20- Equações de Subida e Descida regem as Funções de Pertinência (Fonte: Próprio Autor).

O segundo resultado é a obtenção da base de regras editável e ajustável ao programador; esse sistema identifica como as variáveis incertas de entrada deveriam se comportar na influência da saída.

O terceiro resultado é o processamento de dados onde a maior parte da programação consiste. E aqui que teve que ser dedicado maior tempo e prioridade, pois, além de maior complexidade dos passos anteriores, este tinha a função de uni-los de maneira coerente.

Por fim foi adicionado o sistema de defuzzificação, baseado no método de centro de gravidade, assim fechando o programa com todas as funcionalidades de

um sistema Fuzzy de Mamdani.

O sistema foi testado para uma entrada de 20 ml de água e 60 minutos de sol. Como pode ser visto na linha 2 da figura 20. O resultado final obtido pelo lógica Ladder foi de 0.5, podendo ser observado na linha 3 do programa Ladder da figura 21.

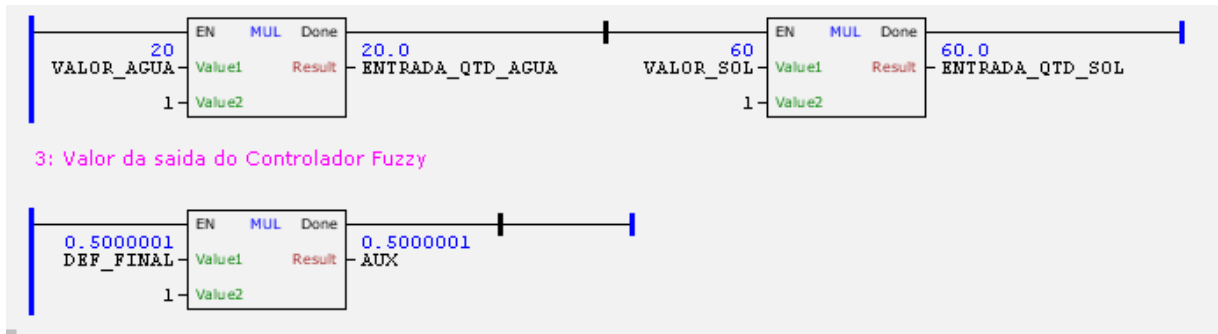


Figura 21 - Fuzzificação em Ladder para entradas de Água e Sol (20,60) (Fonte: Próprio Autor).

Após o registro desta fuzzificação em Ladder aplicou-se as mesmas entradas de 20 ml de água e 60 minutos de sol para a ferramenta Toolbox do Matlab. Como pode ser observado na figura 22.

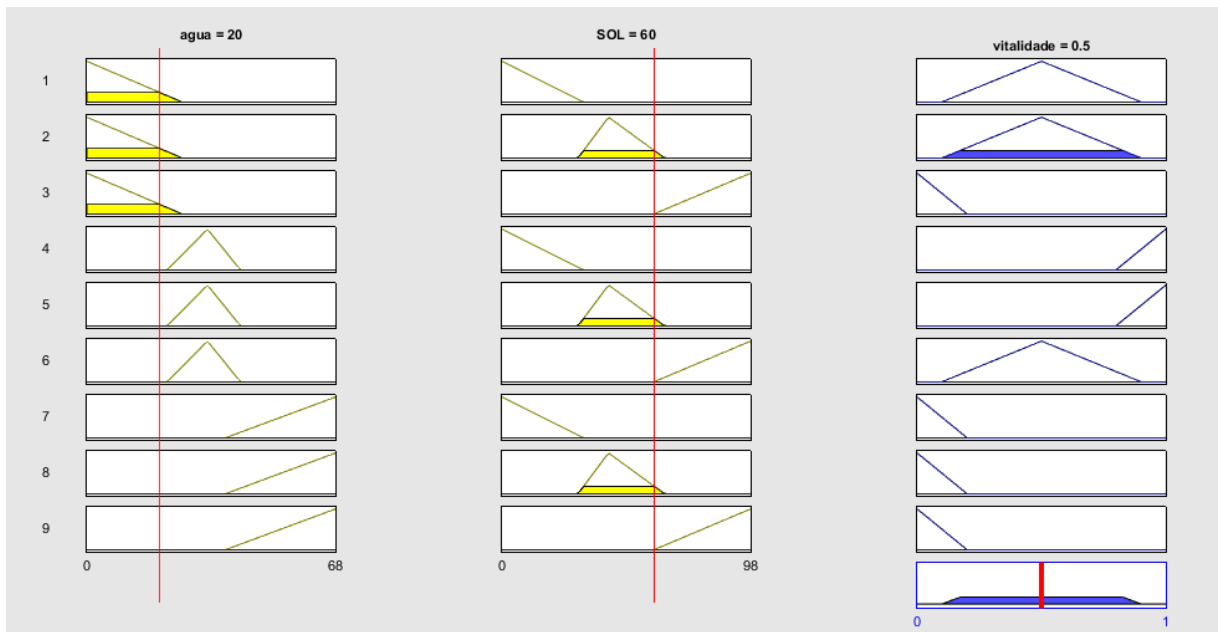


Figura 22 - Fuzzificação em Ladder para entradas de Água e Sol (20,60) (Fonte: Próprio Autor).

Outro teste foi realizado na lógica Ladder introduzindo 35 ml de água e 50 minutos de sol. Como pode ser observado na figura 23, o resultado destas fuzzificação é 0,924.

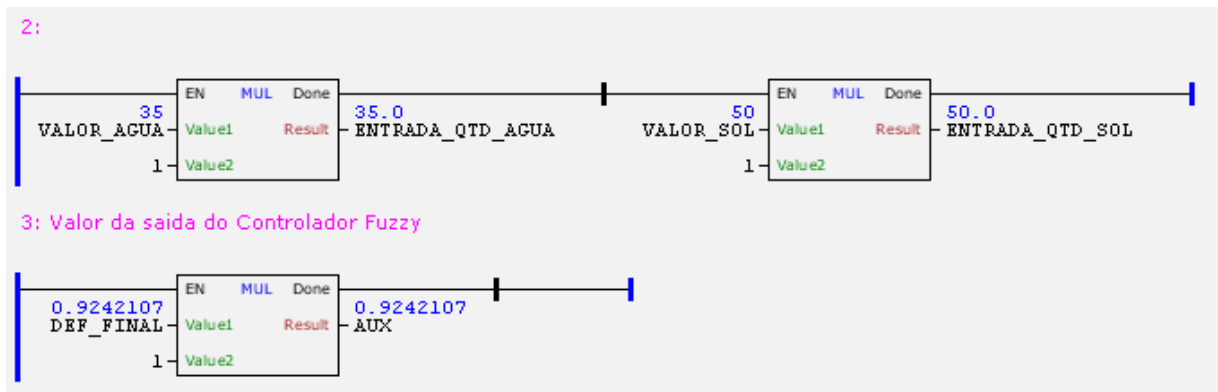


Figura 23 - Fuzzificação em Ladder para entradas de Água e Sol (35,50) (Fonte: Próprio Autor).

Realizando o mesmo teste(35ml de água, 50 minutos de sol) na ferramenta Toolbox do Matlab é obtido um resultado de 0,93. Como mostrado na figura 24.

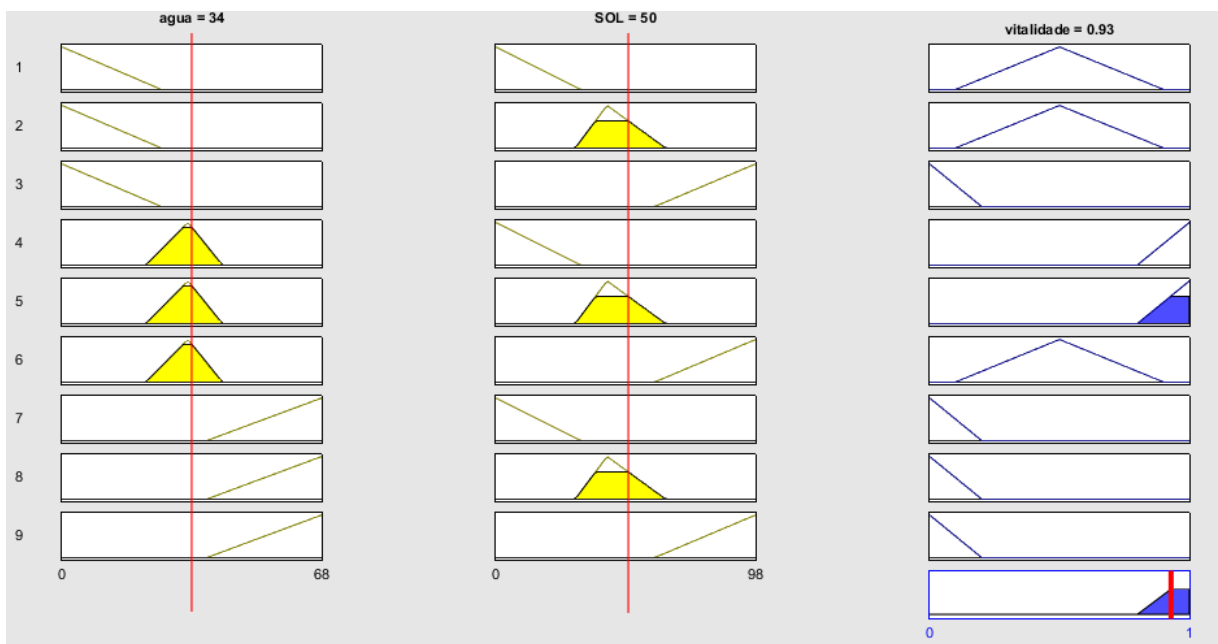


Figura 24 - Fuzzificação em Ladder para entradas de Água e Sol (35,50) (Fonte: Próprio Autor).

Um terceiro teste foi desenvolvido com as entradas de 50 ml de quantidade de água e 60 minutos de quantidade de sol. O resultado obtido pode ser visto na figura 25.

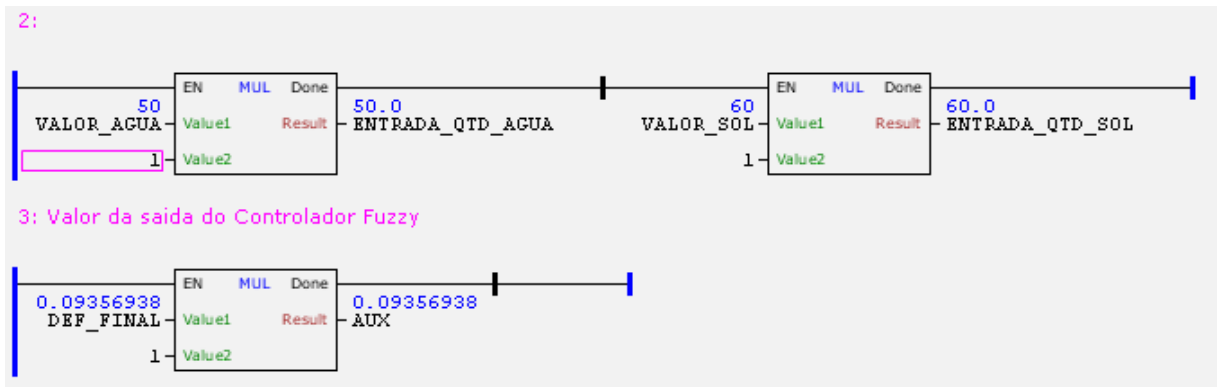


Figura 25- Fuzzificação em Ladder para entradas de Água e Sol (50,60) (Fonte: Próprio Autor).

Realizando o mesmo teste(50ml de água, 60 minutos de sol) na ferramenta Toolbox do Matlab é obtido um resultado de 0,086. Como mostrado na figura 26.

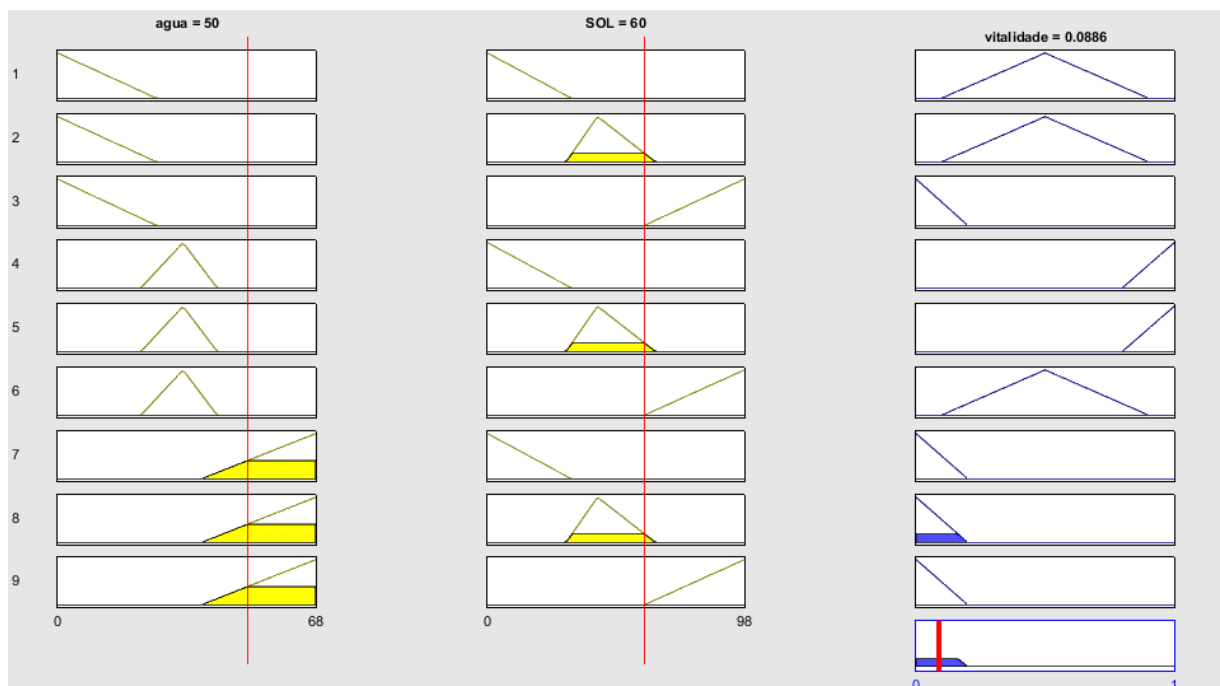


Figura 26- Fuzzificação em Ladder para entradas de Água e Sol (50,60) (Fonte: Próprio Autor).

Além de realizar os testes no próprio programa do CLP300 WEG, foi verificado o funcionamento com os mesmos testes no próprio display do CLP300 WEG, como demonstrado na figura 28. Ao qual foi submetido a programação desenvolvida.



Figura 27- Teste com CLP300 WEG no laboratório do campus (Fonte: Próprio Autor).

Os resultados apresentados no display mostraram-se condizentes com os apresentados na programação em Ladder. Para as mesmas entradas o display apresentou a mesma saída do programa. Indicando o bom funcionamento do CLP 300 WEG no que rege ao display.

5 CONCLUSÃO

Entende-se como sucesso o primeiro resultado apresentado. Visto que este consegue em linguagem Ladder simular as equações de descida e subida do gráfico.

Ademais, pode-se constatar como correta a base de regra utilizada; quando não é possível alterá-la, dependendo da aplicação que se deseja obter.

Averigua-se também o processador como funcional. Tomando como comparação outros programas de análise matemática como o Matlab, o desenvolvido neste trabalho consegue informar o resultado corretamente com a mesma precisão de casas decimais.

Conclui-se que foi atingido o objetivo proposto. Construiu-se através da linguagem de programação Ladder um sistema de lógica fuzzy aplicada a CLPs de pequeno porte. Esse sistema consegue aplicar o método inferência de Mamdani independente da entrada proposta. Conclui-se também que esse sistema pode ser aplicado em diversas aplicações industriais, desde de controle de temperatura, controle de vazão e nível, até métodos matemáticos que não possuem entradas físicas.

Sugestão para trabalhos futuros:

Implementação deste sistema em uma planta industrial real, ou seja, realizar o monitoramento em um sistema real de controle com variáveis físicas e sistema supervisorio. Também realizar a comparação deste sistema com o PID para sistemas não lineares.

REFERÊNCIAS

BALDOVINO, R. G. e DADIOS, E. P., *Design and development of a fuzzy-PLC for an earthquake simulator/shake table*, 2014 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Palawan, Philippines, pp. 1-6, 2014. Acesso em 14 de Fevereiro de 2023.

GE, Software de programação de CLP da família GE Fanuc, Disponível em: <<https://www.ge.com/digital/applications/proficy-software-family>>. Acesso em 14 de Fevereiro de 2023.

GUSAROVA, A. A. e SHILKINA, S. V., *Modeling the Operation of the System in the CODESYS Software Environment*, 2019 International Science and Technology Conference "EastConf", Vladivostok, Russia, pp. 1-6, 2019. Acesso em: 20 de abril.2023.

KOCIAN, J. , KOZIOREK, J. e POKORNY, M. *Implementation of fuzzy logic control based on PLC*, ETFA 2011, Toulouse, France, pp. 1-8, 2011. Acesso em: 22 de abril.2023.

MARCIAL, Anna Lygia. ACLIMATAÇÃO DE PLANTAS DE VIOLETA (Saintpaulia ionantha Wendl) OBTIDAS IN VITRO: EFEITOS DO SUBSTRATO. Acesso em 03 de julho de 2023.

MATTEDE, Henrique. O que é CLP e qual a sua aplicação. Mundo da Elétrica. Belo Horizonte. Disponível em: <https://www.mundodaeletrica.com.br/o-que-e-clp-para-que-serve/>. Acesso em: 26 de abril.2023.

Mercado Livre, Site de compra e venda de produtos. Disponível em : <https://produto.mercadolivre.com.br/MLB-2997996031-controlador-logico-programavel-weg-plc300-JM#position=7&search_layout=stack&type=item&tracking_id=a5c16e11-409b-47e7-b869-b787551d1d6a>. Acesso em 11 de abril de 2023.

MOÇAMBIQUE, N. E. M.; Aplicação de Algoritmos de Busca do Ponto de Máxima Potência e controladores lineares e/ou Fuzzy para regulação da tensão terminal de Painéis Fotovoltaicos. USP São Carlos, 2012. Acesso em: 26 de abril.2023.

MORAES, Jhonathans. UNB. Implementação de controle fuzzy em CLP industrial. Acesso em: 15 de maio.2023.

Morley(R.E). hall of fame richard morley. Disponível em: <https://themcaa.org/industry-insight/hall-of-fame/hall-of-fame-richard-morley/> .Acesso em:15 de maio.2023.

PEDAMKAR. P. Matlab Toolbox, 2015. Disponível em: <https://www.educba.com/matlab-toolbox/>. Acesso em 13 de junho de 2023.

PÉREZ, I. G., GODOY, A. J. C. e GODOY, M. C., Fuzzy controller based on PLC S7-1200: Application to a servomotor, *2014 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, Vienna, Austria, pp. 156-163, 2014. Acesso em 26 de maio.2023.

SILVA, F. F. B. Desvendando a Lógica Fuzzy. Dissertação de Mestrado, Universidade Federal de Uberlândia, Uberlândia/MG, 2011. Acesso em 26 de maio.2023.

SILVA, P., MATLAB: O que é e para que serve. Disponível em: [https://opencadd.com.br/o-que-e-matlab/#:~:text=Boa%20leitura!-,O%20que%20%C3%A9%20o%20MATLAB,conjunto%20de%20bibliotecas%20\(toolboxes\)](https://opencadd.com.br/o-que-e-matlab/#:~:text=Boa%20leitura!-,O%20que%20%C3%A9%20o%20MATLAB,conjunto%20de%20bibliotecas%20(toolboxes)). Acesso em 13 de junho de 2023.

UNESP. Agro fuzzy. UNESP. Acesso em 15 de maio.2023.

UNICAMP. Manual da teoria dos conjuntos Fuzzy no Matlab. UNICAMP. Acesso em 14 de junho. 2023.

WEG, Automação e Controle Industrial. Disponível em:

<https://www.weg.net/catalog/weg/BR/pt/Automa%C3%A7%C3%A3o-e-Controle-Industrial/Drives/Softwares/WEG-Programming-Suite-%28WPS%29/WEG-Programming-Suite-%28WPS%29/p/MKT_WDC_BRAZIL_SOFTWARE_WPS>. Acesso em: 12 de abril de 2023.

WEG. Controlador Lógico Programável PLC300. Disponível em:

https://www.weg.net/catalog/weg/BR/pt/Automa%C3%A7%C3%A3o-e-Controle-Industrial/Controle-de-Processos/Controladores-L%C3%B3gicos-Program%C3%A1veis/Controlador-L%C3%B3gico-Program%C3%A1vel-PLC300/Controlador-L%C3%B3gico-Program%C3%A1vel-PLC300/p/MKT_WDC_BRAZIL_PROGRAMMABLE_LOGIC_CONTROLLER_PLC_PLC300. Acesso em: 02 de maio.2023

WEG. PLC300 manual do usuário. Disponível em:

<https://static.weg.net/medias/downloadcenter/ha2/h4e/WEG-10004642430-PLC300-h3-manual-usuario-pt.pdf>. Acesso em: 12 de maio.2023

Zadeh(L.A), Lotfali Askar-Zadeh. Disponível

em: https://www.wikiwand.com/pt/Lotfali_Askar-Zadeh. Acesso em 26 de maio. 2023.