



UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
FACULDADE DE ENGENHARIAS ELÉTRICA E BIOMÉDICA
CURSO DE ENGENHARIA ELÉTRICA

**COMPARAÇÃO DE DESEMPENHO ENTRE
CONTROLADORES PID E FUZZY BASEADO
EM PID: UMA ANÁLISE APLICADA A
SISTEMAS DE CONTROLE AUTOMÁTICO**

JOÃO VICTOR MOURA SILVA

UFPA / ITEC
Campus Universitário do Guamá
Belém-Pará-Brasil

2025

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
FACULDADE DE ENGENHARIAS ELÉTRICA E BIOMÉDICA

JOÃO VICTOR MOURA SILVA

**COMPARAÇÃO DE DESEMPENHO ENTRE CONTROLADORES PID
E FUZZY BASEADO EM PID: UMA ANÁLISE APLICADA A
SISTEMAS DE CONTROLE AUTOMÁTICO**

Trabalho de Conclusão de Curso, apresentado à
Faculdade de Engenharias Elétrica e Biomédica
como requisito parcial para obtenção do Grau
de Bacharel em Engenharia Elétrica, pela
Universidade Federal do Pará.

Orientador: Prof. Dr. ANTONIO DA SILVA SILVEIRA

UFPA / ITEC
Campus Universitário do Guamá
Belém-Pará-Brasil


2025




SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
FACULDADE DE ENGENHARIA ELÉTRICA E BIOMÉDICA

ATA DE DEFESA DE TRABALHO DE CURSO (TC)


Ata de defesa de Trabalho de Curso (TC) intitulado: “COMPARAÇÃO DE DESEMPENHO ENTRE CONTROLADORES PID E FUZZY BASEADO EM PID: UMA ANÁLISE APLICADA A SISTEMAS DE CONTROLE AUTOMÁTICO”, apresentado durante 60 minutos pelo discente João Victor Moura Silva, matrícula nº 201707140051, realizada às 10:00h do dia 22 de Abril de 2025, no Sala de Ensino do Laboratório de Controle e Sistemas da UFPA e também por videoconferência com Google Meet pela sessão <https://meet.google.com/dbw-dxhh-kjz>, sob orientação do Prof. Dr. Antonio da Silva Silveira. Após a apresentação, o discente foi submetido à arguição oral pelos membros da banca examinadora. Em seguida, a banca se reuniu, em caráter sigiloso, para decidir o resultado do exame, ato que deliberou pela APROVAÇÃO do trabalho com o conceito EXCELENTE. Para serem efetuadas as modificações sugeridas pela banca para desenvolver a versão final do Trabalho de Curso, o discente recebeu o prazo máximo de 08 (oito) dias, a contar da data desta defesa. Nada mais havendo a tratar, a sessão foi encerrada, lavrando-se dela a presente Ata que, uma vez aprovada, foi assinada pelos membros da banca examinadora e pelo discente.

Documento assinado digitalmente
 ANTONIO DA SILVA SILVEIRA
Data: 23/04/2025 10:48:19-0300
Verifique em <https://validar.iti.gov.br>


Prof. Dr. Antonio da Silva Silveira
(Orientador – UFPA)

Documento assinado digitalmente
 ORLANDO FONSECA SILVA
Data: 23/04/2025 12:20:39-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Orlando Fonseca Silva
(Membro – UFPA)

Documento assinado digitalmente
 CARLOS TAVARES DA COSTA JUNIOR
Data: 28/04/2025 20:53:14-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Carlos Tavares da Costa Júnior
(Membro – UFPA)

Documento assinado digitalmente
 JOAO VICTOR MOURA SILVA
Data: 23/04/2025 11:29:52-0300
Verifique em <https://validar.iti.gov.br>

João Victor Moura Silva
(Discente)

Dedico à minha família e aos amigos.

Agradecimentos

Agradeço, primeiramente, a Deus, por me guiar, fortalecer e iluminar meu caminho durante toda essa jornada. Sem Sua graça e providência, nenhuma das minhas conquistas teria sido possível.

Aos meus pais, Edna e João Lindomar, por todo o amor, apoio e incentivo incondicional ao longo da minha jornada acadêmica. Sem vocês, nada disso seria possível.

À minha irmã, Joana Karoline, por estar sempre ao meu lado, oferecendo suporte e palavras de encorajamento nos momentos difíceis.

À minha namorada, Maria Clara, pelo carinho, paciência e por acreditar em mim, mesmo quando eu duvidei de mim mesmo. Sua presença foi essencial durante esse período desafiador.

Às minhas primas, Poliana e Juliana, pela amizade, pelos conselhos e por compartilharem comigo tantos momentos especiais ao longo da minha vida.

Aos meus amigos, Davyd, Caio, Glécia, Gabrielle, Eduardo, Lucas, Victor e Juliano, pelo companheirismo, pelas conversas descontraídas e pelo apoio nos momentos de cansaço e desafios acadêmicos.

Ao meu professor orientador, Antonio Silveira, pela dedicação, paciência e por compartilhar seu conhecimento, guiando-me nessa caminhada com sabedoria e incentivo. Sua orientação foi fundamental para a realização deste trabalho.

Ao Programa de Educação Tutorial, pelo apoio acadêmico e pelas oportunidades de aprendizado e crescimento que me proporcionou ao longo da minha formação.

Por fim, agradeço a todos que, de alguma forma, contribuíram para o meu crescimento pessoal e profissional, tornando essa conquista possível.

Resumo

Este trabalho compara os controladores proporcional-integral-derivativo (PID) e *Fuzzy*, com foco na análise de desempenho de cada técnica aplicada a sistemas de controle automático. A revisão teórica foi seguida de simulações no software Scilab, utilizando dois sistemas de segunda ordem: um com resposta ao degrau subamortecida e outro criticamente amortecido. Foram realizados testes de resposta ao degrau unitário e resposta a sequência de degraus com perturbações de carga e ruídos aleatórios. A performance dos controladores foi avaliada com base em índices como IAE, ISE, ITAE, ITSE, ISU e J (ISE+ISU), considerando a precisão, o tempo de resposta e a robustez frente a distúrbios. Os resultados mostraram que o controlador PID apresentou bom desempenho em sistemas simples, enquanto o controlador *Fuzzy* se destacou em cenários com perturbações e ruídos, mostrando maior robustez e capacidade de adaptação. A utilização da toolbox SCIFLT no software Scilab foi essencial para a implementação das técnicas de controle *Fuzzy*. Conclui-se que, embora o controlador PID seja mais simples, o controlador *Fuzzy* mostrou-se mais adequado para sistemas sujeitos a distúrbios, destacando-se em termos de adaptabilidade e robustez.

Palavras-chave: Controle Proporcional-Integral-Derivativo, Controle Fuzzy, Sistemas de Controle Automático, Scilab.

Abstract

This work compares proportional-integral-derivative (PID) controllers and *Fuzzy* controllers, focusing on performance analysis of each technique applied to automatic control systems. A theoretical review was followed by simulations in Scilab software, using two second-order systems: one with an underdamped step response and the other critically damped. Tests were conducted for unit step response and step sequences with load disturbances and random noise. The controllers' performance was evaluated using indices such as IAE, ISE, ITAE, ITSE, ISU, and J (ISE+ISU), considering accuracy, response time, and robustness against disturbances. Results showed that the PID controller performed well in simple systems, while the *Fuzzy* controller excelled in scenarios with disturbances and noise, demonstrating greater robustness and adaptability. The use of the SCIFLT toolbox in Scilab was essential for implementing the *Fuzzy* control techniques. It is concluded that, although the PID controller is simpler, the *Fuzzy* controller proved more suitable for systems subject to disturbances, standing out in terms of adaptability and robustness.

Key-words: Proportional-Integral-Derivative Control, Fuzzy Control, Automatic Control Systems, Scilab.

Lista de figuras

Figura 1 – Regulador de esferas de Watt	4
Figura 2 – Diagrama de blocos do sistema de controle PID	6
Figura 3 – A estrutura de um sistema de <i>Fuzzy</i>	7
Figura 4 – Tipos de funções de pertinência.	8
Figura 5 – Variável Linguística Temperatura.	9
Figura 6 – Métodos de Defuzzificação	11
Figura 7 – Resposta do 1º Sistema em Malha Aberta.	12
Figura 8 – Modelo equivalente do motor DC.	14
Figura 9 – Resposta do 2º Sistema (Motor DC) em Malha Aberta.	16
Figura 10 – SCAM (Sliced Cube Associative Memory).	20
Figura 11 – Índices de especificação do Sistema.	24
Figura 12 – Ambiente de Simulação Scilab.	25
Figura 13 – Ambiente gráfico da biblioteca SCIFLT.	25
Figura 14 – Ambiente de criação de <i>scripts</i>	26
Figura 15 – Conjuntos <i>Fuzzy</i> de entrada para o 1º Sistema.	29
Figura 16 – Conjuntos <i>Fuzzy</i> de saída para o 1º Sistema.	29
Figura 17 – Superfícies de Controle <i>Fuzzy</i> para o 1º Sistema.	30
Figura 18 – Resposta ao degrau unitário do 1º sistema com os controladores.	31
Figura 19 – Sinais de Erro do 1º Sistema.	32
Figura 20 – Resposta a série de degraus do 1º Sistema com ruídos e saturação.	34
Figura 21 – Variação dos índices de desempenho do 2º Caso.	35
Figura 22 – Conjuntos <i>Fuzzy</i> de entrada para o 2º Sistema.	37
Figura 23 – Conjuntos <i>Fuzzy</i> de saída para o 2º Sistema.	37
Figura 24 – Superfícies de Controle <i>Fuzzy</i> para o 2º Sistema.	38
Figura 25 – Resposta ao degrau unitário do 2º Sistema.	39
Figura 26 – Sinais de Erro do 2º Sistema.	40
Figura 27 – Resposta a série de degraus do 2º Sistema com ruídos.	42
Figura 28 – Variação dos índices de desempenho do 4º Caso.	43

Lista de tabelas

Tabela 1 – Parâmetros do Motor DC.	15
Tabela 2 – Tabela para $\dot{e}(t) = N$	27
Tabela 3 – Tabela para $\dot{e}(t) = NM$	28
Tabela 4 – Tabela para $\dot{e}(t) = Z$	28
Tabela 5 – Tabela para $\dot{e}(t) = PM$	28
Tabela 6 – Tabela para $\dot{e}(t) = P$	28
Tabela 7 – Comparação dos índices de desempenho entre os controladores PID com Filtro e <i>Fuzzy-PID</i> para o 1º Caso.	33
Tabela 8 – Comparação das especificações de resposta transitória dos controladores para o 1º Caso.	33
Tabela 9 – Comparação da média dos índices de desempenho entre os controladores PID com Filtro e <i>Fuzzy-PID</i> para o 2º Caso.	35
Tabela 10 – Comparação dos índices de desempenho entre os controladores PID com Filtro e <i>Fuzzy-PID</i> para o 3º Caso.	41
Tabela 11 – Comparação das especificações de resposta transitória dos controladores para o 3º Caso.	41
Tabela 12 – Comparação da média dos índices de desempenho entre os controladores PID com Filtro e <i>Fuzzy-PID</i> para o 4º Caso.	43

Lista de abreviaturas e siglas

CDA	Centro de Área
FEEB	Faculdade de Engenharias Elétrica e Biomédica
Fuzzy	Lógica Difusa (do inglês, Fuzzy Logic)
IAE	Integral of Absolute Error
ISE	Integral of Squared Error
ISU	Integral of Squared Control
J	Função de Custo
ITAE	Integral of Time-weighted Absolute Error
ITSE	Integral of Time-weighted Squared Error
MAX	Valor Máximo
MDM	Média dos Máximos
PID	Proporcional, Integral e Derivativo
SCAM	Sliced Cube Associative Memory
UFPA	Universidade Federal do Pará

Lista de símbolos

$\mu_{Tri}(x)$	Função de pertinência triangular
$\mu_{Tra}(x)$	Função de pertinência trapezoidal
$\mu_{Gauss}(x)$	Função de pertinência gaussiana
$\mu_Z(x)$	Função de pertinência tipo-Z
$\mu_S(x)$	Função de pertinência tipo-S
a, b, c, d	Parâmetros de forma das funções de pertinência
x	Variável de entrada da função de pertinência
σ	Desvio padrão da função gaussiana.
$e(t)$	Erro entre a saída do sistema e o valor de referência
$\dot{e}(t)$	Derivada do erro no tempo
$\int e(t)$	Integral do erro no tempo
$u(t)$	Sinal de controle aplicado ao sistema
$u(k)$	Sinal de controle no tempo discreto
$u(k - 1)$	Sinal de controle no tempo discreto anterior
$u(k - 2)$	Sinal de controle no tempo discreto de dois passos anteriores
$\omega(t)$	Velocidade angular do motor
V_a	Tensão da armadura do motor
R_a	Resistência da armadura do motor
L_a	Indutância da armadura do motor
i_a	Corrente da armadura do motor
E_b	Força contraeletromotriz do motor
T	Torque gerado pelo motor
θ	Posição angular do eixo do motor

J	Momento de inércia do rotor
B	Coefficiente de atrito viscoso
K_e	Constante de força contraeletromotriz
K_t	Constante de torque do motor
T_s	Tempo de amostragem
f_s	Frequência de amostragem
k_p	Ganho proporcional do controlador PID
k_i	Ganho integral do controlador PID
k_d	Ganho derivativo do controlador PID
ω_c	Frequência de corte do filtro no controlador PID
s	Operador de Laplace
z	Operador de Z-transform (transformada Z)
$\mu(E)$	Função de pertinência das variáveis de entrada no controlador fuzzy
α	Ativação da regra fuzzy

Sumário

1	INTRODUÇÃO	1
1.1	Objetivo Geral	2
1.2	Objetivos Específicos	2
1.3	Estrutura do documento	3
2	REVISÃO BIBLIOGRÁFICA	4
2.1	História dos Controladores.	4
2.2	Controladores na Atualidade	5
2.3	O Controlador Proporcional, Integral e Derivativo	5
2.4	A Lógica <i>Fuzzy</i>	7
3	METODOLOGIA	12
3.1	Descrição do sistema	12
3.2	Desenvolvimento do PID	16
3.3	Desenvolvimento do Controlador <i>Fuzzy-PID</i>	18
3.4	Critérios de Avaliação de Desempenho	22
3.4.1	IAE (<i>Integral of Absolute Error</i>)	22
3.4.2	ISE (<i>Integral of Squared Error</i>)	22
3.4.3	ITAE (<i>Integral of Time-weighted Absolute Error</i>)	22
3.4.4	ITSE (<i>Integral of Time-weighted Squared Error</i>)	23
3.4.5	ISU (<i>Integral of Squared Control</i>)	23
3.4.6	J (Função de Custo)	23
3.5	Ferramentas de Simulação	24
4	RESULTADOS E DISCUSSÃO	27
4.1	1º Caso: Resposta ao degrau unitário do 1º Sistema	29
4.2	2º Caso: Resposta a série de degraus do 1º Sistema com ruídos e saturação.	34
4.3	3º Caso: Resposta ao degrau unitário do 2º Sistema.	37
4.4	4º Caso: Resposta a série de degraus do 2º Sistema com ruídos.	42
5	CONCLUSÃO	45
	Referências	46
A	Códigos da Simulação	48
A.1	Código Principal	48
A.2	Funções Auxiliares	61
A.2.1	Função de Simulação do Sistema	61
A.2.2	Função de Controle PID	61

A.2.3	Função de Perturbações	61
A.2.4	Função de Criação do Sistema Fuzzy	62

1 INTRODUÇÃO

A escolha de métodos eficientes é de suma importância para a otimização de sistemas em várias áreas da engenharia. O controlador proporcional-integral-derivativo (PID) é massivamente aplicado. Isso se deve a sua simplicidade e eficácia frente a diversas aplicações industriais, visto que ele é eficaz na correção de erros de sistema, uma vez que ajusta de forma contínua a saída do sistema com base na diferença entre o valor desejado e o real (OGATA, 2010).

De outra perspectiva, a lógica *Fuzzy* proporciona uma forma alternativa para lidar com a incerteza e a imprecisão inerente a diversos sistemas reais. A lógica *Fuzzy* permite a modelagem de sistemas complexos através de regras linguísticas, o que fornece uma forma natural de tratar problemas de implementação de controladores que podem lidar com imprecisões (GOGUEN, 1973).

A comparação entre controladores PID e *Fuzzy* é um tema de relevância na literatura acadêmica. O controle convencional, PID, se dá, normalmente, a partir de um modelo matemático do processo e os controladores são projetados com base nesse modelo, já os controladores *Fuzzy* são feitos através do uso de métodos heurísticos combinados com a expertise humana. O controle *Fuzzy* pode ser útil para a maioria dos processos de controle práticos, visto que nesses tipos de problemas a obtenção de um modelo matemático acurado e simples é difícil de ser obtido (WANG, 1997).

Consequentemente, a constante evolução dos métodos de controle tem fomentado a criação de técnicas avançadas, como o controle adaptativo e o controle robusto. O controle adaptativo realiza o ajuste dos parâmetros do controlador em tempo real para lidar com alterações no sistema, de modo que ele promova uma resposta mais precisa e eficiente (ÅSTRÖM; WITTENMARK, 2013). Em contrapartida, o controle robusto é projetado para manter o desempenho do sistema mesmo na presença de incertezas e perturbações externas (ZHOU; DOYLE, 1998). Essas abordagens têm sido amplamente aplicadas em áreas como a engenharia aeroespacial, sistemas de energia e automação industrial, demonstrando sua eficácia em ambientes complexos e dinâmicos.

O estudo de técnicas de controle mais simples, como o PID e a lógica *Fuzzy*, é fundamental para a compreensão e desenvolvimento de métodos de controle mais avançados. Esses controladores fornecem uma base sólida de conceitos essenciais, como estabilidade, robustez e compensação de erros, que são diretamente aplicáveis a sistemas de maior complexidade. O entendimento profundo de técnicas clássicas como o PID é um passo crucial para a assimilação de controles mais sofisticados, como os sistemas adaptativos e robustos, uma vez que muitos princípios básicos permanecem relevantes (DORF; BISHOP, 2009). Da mesma forma, a lógica *Fuzzy*, ao permitir a manipulação de imprecisões e incertezas em sistemas, introduz conceitos

importantes que são expandidos em métodos de controle mais avançados, como o controle *Fuzzy* adaptativo (PASSINO; YURKOVICH, 1998). Portanto, a exploração dessas técnicas mais simples não só tem valor prático em muitas aplicações, mas também serve como um alicerce teórico para o entendimento de sistemas de controle de maior complexidade.

Este trabalho tem como objetivo comparar o controlador proporcional-integral-derivativo (PID) com o método de controle utilizando a lógica *Fuzzy* com parâmetros baseados no controlador PID, visando uma melhor compreensão de suas particularidades, bem como o entendimento da dinâmica de funcionamento de cada um e a identificação das áreas em que eles apresentam um desempenho mais otimizado. Para isso, o software *Scilab* foi utilizado para realizar as simulações dos modelos e o projeto dos controladores.

Ao fim do trabalho, através da análise baseada em critérios de tempo de subida, tempo de acomodação, *overshoot* percentual, variância e os índices de desempenho, notou-se a eficiência e as particularidades de cada tipo de controlador aplicado a essas plantas.

1.1 Objetivo Geral

O objetivo geral deste trabalho é comparar o método de controle PID com o método de controle que utiliza a lógica *Fuzzy-PID*, visando compreender a dinâmica de funcionamento de cada um e identificar as áreas em que estes apresentam um desempenho mais otimizado. Para isso, os controladores foram aplicados em duas plantas de segunda ordem, uma com resposta ao degrau subamortecida e outra criticamente amortecida, obtida a partir de um modelo de um motor dc. Essas duas dinâmicas de modelo existem amplamente no âmbito industrial, no posicionamento de braços robóticos em que oscilações ocorrem ao mover cargas e o *overshoot* pode danificar peças e ferramentas, assim como no controle de velocidade de motores em esteiras, em que é necessária uma aceleração suave para evitar o derramamento de materiais. Para isso, as simulações se deram com o intuito de avaliar o desempenho dos controladores.

1.2 Objetivos Específicos

- Revisar os conceitos teóricos fundamentais para os controladores PID e *Fuzzy*, fornecendo a base para a comparação entre esses dois métodos.
- Desenvolver e implementar controladores PID e *Fuzzy-PID* para dois sistemas de segunda ordem representativos: um com resposta ao degrau subamortecida e outro criticamente amortecido.
- Realizar simulações no software *Scilab*, colocando os controladores frente a uma entrada de degrau unitário para ambos os sistemas. Testando, também, a resposta dos controladores a uma sequência de degraus, analisando a capacidade de adaptação do sistema a variações

de entrada, introduzindo ruídos e saturação no sinal de controle para avaliar a robustez e a eficiência dos controladores em condições de distúrbios.

- Avaliar o desempenho dos controladores com base em índices como IAE, ISE, ITAE, ITSE, ISU e J, considerando o impacto da saturação, ruídos e a precisão nos tempos de resposta.
- Mapear as deficiências e vantagens de cada método, proporcionando uma visão clara das situações em que um controlador pode ser preferível ao outro.

1.3 Estrutura do documento

Com o intuito de imprimir uma lógica coesa para a exposição do conteúdo apresentado no estudo, o trabalho foi dividido em um total de cinco capítulos.

- **Capítulo 1: Introdução** - Apresenta o contexto geral do estudo, os objetivos principais e específicos, além da estrutura do documento. Este capítulo serve como base para entender a motivação e a importância da pesquisa realizada.
- **Capítulo 2: Revisão Bibliográfica** - Realiza uma revisão dos conceitos teóricos relacionados aos controladores PID e *Fuzzy*. Este capítulo aborda as origens, o desenvolvimento e as aplicações atuais desses controladores, além das vantagens e limitações de cada abordagem.
- **Capítulo 3: Metodologia** - Descreve detalhadamente os métodos e procedimentos utilizados para realizar as simulações, testes e análises comparativas dos controladores PID e *Fuzzy-PID*. Inclui a descrição dos sistemas utilizados, o desenvolvimento dos controladores e os critérios de avaliação de desempenho aplicados.
- **Capítulo 4: Resultados e Discussão** - Apresenta os resultados obtidos nas simulações e discute o desempenho dos controladores PID e *Fuzzy-PID*, comparando-os com base em parâmetros como precisão, tempo de resposta, e capacidade de adaptação a distúrbios. Este capítulo também analisa a eficácia de cada controlador em diferentes cenários.
- **Capítulo 5: Conclusão** - Resume os principais achados do estudo, destacando as vantagens e desvantagens de cada controlador em termos de desempenho. Este capítulo também sugere direções para trabalhos futuros, incluindo a possibilidade de integração de controladores híbridos e a validação experimental dos resultados obtidos.

2 REVISÃO BIBLIOGRÁFICA

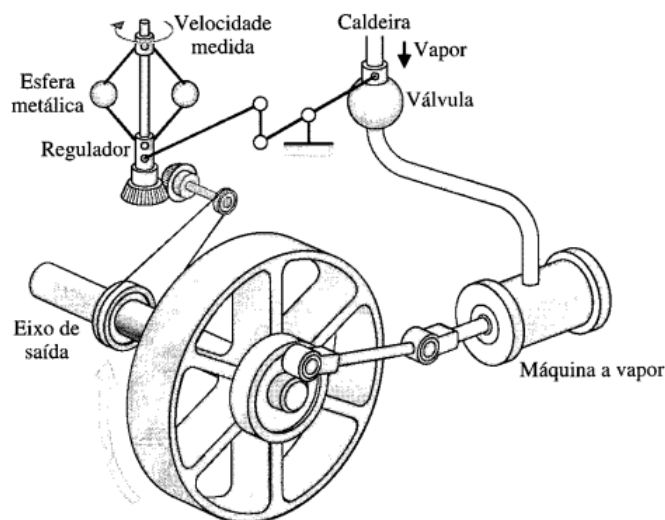
Para a compreensão dos controladores e suas particularidades, é necessária uma revisita a história e uma revisão acerca dos conceitos e dos avanços existentes ao longo do tempo.

2.1 História dos Controladores.

A origem dos controladores vem do século XVIII. A criação do motor a vapor marcou a época devido ao progresso tecnológico e a revolução industrial proporcionada pela dinamização das relações de tempo até então conhecidas (PASQUINI, 2014). Ao longo desse período, a procura por processos mais eficazes e automatizados se tornou uma meta cada vez mais premente, particularmente nas áreas têxtil e mecânica. Dentro deste cenário, emergiu o primeiro controlador automático utilizado em um processo industrial: o regulador de esferas, criado por James Watt em 1769 (DORF; BISHOP, 2009).

O regulador centrífugo foi projetado para regular a velocidade das máquinas a vapor, cruciais naquele período, ajustando a quantidade de vapor que entrava na máquina conforme sua velocidade de funcionamento (PERRONE, 2020). O dispositivo mostrado na Figura 1 operava através de um conjunto de esferas de metal ligadas a um eixo rotativo. A medida que a máquina acelerava, as esferas se afastavam do eixo devido a ação da força centrífuga, abrindo ou fechando válvulas responsáveis pelo controle do vapor; com isso, era possível manter uma velocidade constante, compensando as alterações provocadas por mudanças na carga da máquina (HYUN; PARK, 2016).

Figura 1 – Regulador de esferas de Watt



Fonte: (DORF; BISHOP, 2009)

Com o progresso das engenharias elétrica e eletrônica promovido durante o século XX, e principalmente durante a Segunda Guerra Mundial, surgiram novas necessidades por controle de sistemas mais complexos e exatos (DORF; BISHOP, 2009). Isso resultou na criação do controlador proporcional-integral-derivativo, um método de controle que se estabeleceu predominantemente na indústria (ADÃO, 2011). O controlador PID é um método matemático utilizado para regular sistemas contínuos e discretos, ajustando automaticamente as variáveis de controle com base na ação proporcional, integral e derivativa. Essa combinação de ações permite ao controlador PID estabilizar sistemas e ajustar variáveis de maneira eficiente (OGATA, 2010).

Durante a década de 1960, o avanço dos sistemas de computação possibilitou o surgimento do controle *fuzzy*. O regulador *fuzzy*, proposto por Lotfi Zadeh em 1965, emergiu como uma opção aos métodos convencionais, como o PID, especialmente em sistemas que exibem não linearidades ou comportamentos complexos de serem modelados matematicamente (WANG, 1997). Ao contrário dos controladores tradicionais, que exigem uma representação precisa do sistema, o regulador *fuzzy* emprega uma lógica que imita o comportamento humano, possibilitando o controle em sistemas com incertezas e variáveis imprecisas (ZADEH, 1965).

2.2 Controladores na Atualidade

Atualmente, o controlador PID e o controlador *fuzzy* são amplamente empregados em diversos setores científicos, industriais e de consumo, cada um com suas particularidades, mas ambos têm em comum a finalidade de garantir a estabilidade e eficácia dos sistemas, progredindo continuamente com o progresso tecnológico e as demandas do setor industrial (ÅSTRÖM; WITTENMARK, 2013). O controlador PID e suas variações são usados amplamente em diversos sistemas que exigem alta precisão e controle fino, a exemplo da indústria da manufatura, em sistemas robóticos e processos de controle térmico (OGATA, 2010).

A lógica *Fuzzy* é frequentemente empregada em circunstâncias que apresentem falta de exatidão e incerteza, como em sistemas em que a modelagem exata é difícil, sendo eficaz para tratar de maneira conveniente esse tipo de informação. Isso permite que sejam utilizados em processos complexos e, por consequência, proporciona resultados mais acurados, estáveis e robustos, principalmente por reduzir a complexidade de projetos (GOMIDE, 1994). Esse tipo de técnica é aplicada em diversas situações, como o controle de processos complexos, modelagem de processos industriais, diagnóstico de falhas, processamento de imagens e reconhecimento de padrões (PEÑAS, 2002).

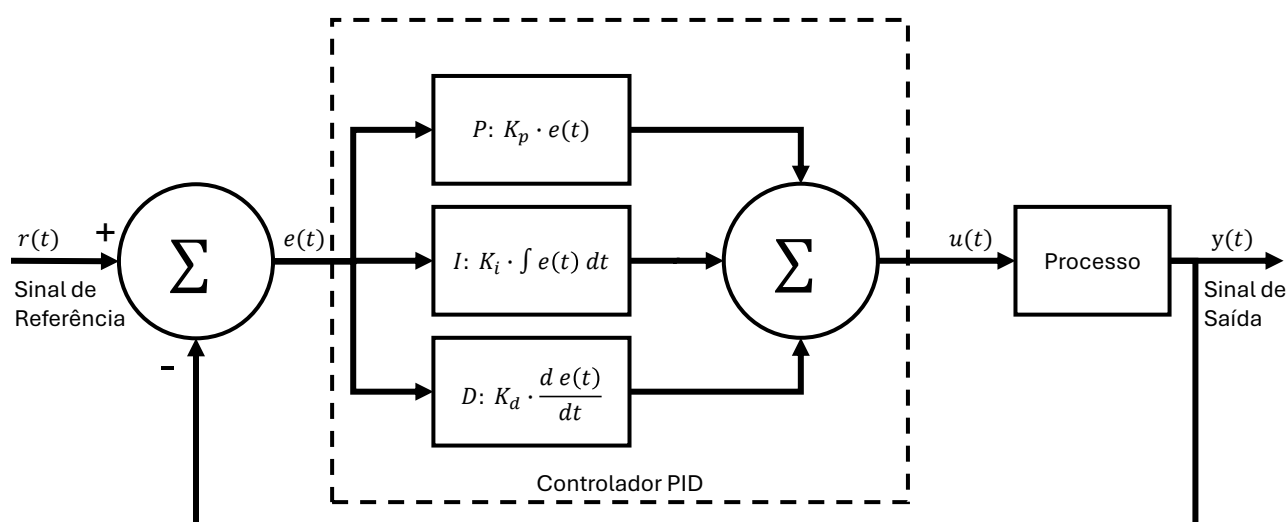
2.3 O Controlador Proporcional, Integral e Derivativo

O controlador PID é uma das técnicas de controle automático mais reconhecidas pela sua efetividade, simplicidade e vasta utilização em sistemas dinâmicos. Sua equação no tempo

contínuo é representada na Equação 2.1. O objetivo principal é ajustar uma variável de controle para reduzir o erro entre o valor desejado (ou *setpoint*) e o valor medido do sistema, conforme exemplifica a Figura 2, assegurando uma resposta ágil e consistente (OGATA, 2010). Esta estratégia possibilita que o controlador PID reaja a variadas circunstâncias operacionais, preservando a estabilidade do sistema e minimizando desvios constantes (ASTROM; HAGGLUND, 1995).

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (2.1)$$

Figura 2 – Diagrama de blocos do sistema de controle PID



Fonte: Adaptado de (HALIM; ISMAIL, 2019).

O controle PID opera a partir de três elementos essenciais: a ação proporcional, a ação integral e a ação derivativa. A ação proporcional (P) corrige o erro de acordo com sua magnitude, isto é, quanto maior o erro, maior é a amplitude da resposta de correção. A intensidade dessa resposta é determinada pelo ganho proporcional K_p , que age de maneira a minimizar o erro de imediato (BATESON, 1980). Já a ação integral (I) acumula o erro ao longo do tempo com o objetivo de eliminar desvios persistentes, sendo perfeita para corrigir o erro residual que a ação proporcional não consegue corrigir. A velocidade com que o controlador responde para eliminar erros acumulados determina o ganho integral K_i , possibilitando que o sistema alcance exatamente o ponto de ajuste (ASTROM; HAGGLUND, 1995). Por outro lado, a ação derivativa (D) prevê variações futuras do erro, ajustando o controle conforme a taxa de alteração do erro. Esta parte é encarregada de suavizar a resposta do sistema, prevenindo oscilações e aprimorando a estabilidade, particularmente em sistemas que passam por mudanças rápidas. O ganho derivativo

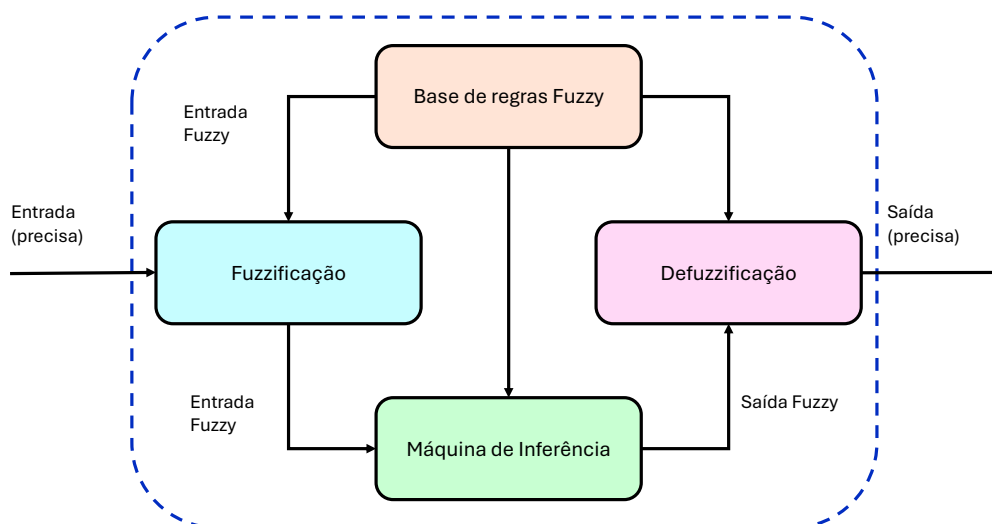
K_d estabelece o impacto da ação derivativa no sistema, proporcionando uma resposta preventiva e estabilizadora que reforça as ações proporcional e integral (DORF; BISHOP, 2009).

2.4 A Lógica Fuzzy

A lógica *Fuzzy*, também conhecida como lógica difusa, foi introduzida por Lotfi Zadeh em 1965 como uma extensão dos sistemas de lógica clássica. Ao contrário da lógica binária convencional, que restringe as variáveis a valores absolutos de "verdadeiro" (1) ou "falso" (0), a lógica *Fuzzy* possibilita graus de verdade variando entre 0 e 1, permitindo a representação de incertezas e ambiguidades (ZADEH, 1965).

A organização básica de um sistema *Fuzzy* consiste em três fases principais: *Fuzzificação*, *Inferência* e *Defuzzificação*, conforme ilustrado na Figura 3.

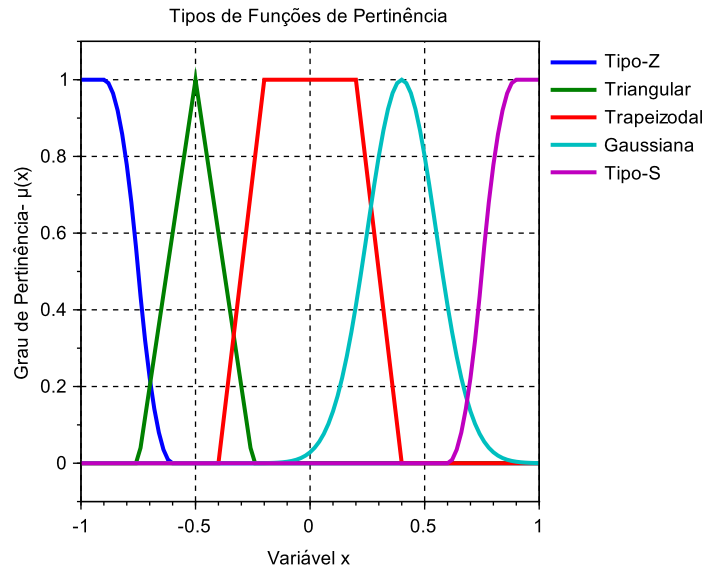
Figura 3 – A estrutura de um sistema de *Fuzzy*



Fonte: Adaptado de (SAATCHI, 2024).

A função de pertinência, também conhecida como função de associação, é usada para estabelecer a pertinência dos elementos em um conjunto *Fuzzy*. Ao contrário da lógica tradicional, onde um elemento pode ou não pertencer a um conjunto, na lógica *Fuzzy* esse elemento pode fazer parte de um conjunto com um grau de pertinência parcial, que pode variar de 0 a 1. Funções típicas de pertinência para a variável *Fuzzy* podem assumir a forma Triangular (Equação 2.2), Trapezoidal (Equação 2.3), Gaussiana (Equação 2.4), Tipo-Z (Equação 2.5) ou Tipo-S (Equação 2.6), conforme exibe a Figura 4.

Figura 4 – Tipos de funções de pertinência.



Fonte: Autor.

$$Triangular : \mu_{Tri}(x) = \begin{cases} 0 & \text{se } x \leq a \\ \frac{x-a}{b-a} & \text{se } a < x \leq b \\ \frac{c-x}{c-b} & \text{se } b < x \leq c \\ 0 & \text{se } x \geq c \end{cases} \quad (2.2)$$

$$Trapezoidal : \mu_{Tra}(x) = \begin{cases} 0 & \text{se } x \leq a \\ \frac{x-a}{b-a} & \text{se } a < x \leq b \\ 1 & \text{se } b < x \leq c \\ \frac{d-x}{d-c} & \text{se } c < x \leq d \\ 0 & \text{se } x \geq d \end{cases} \quad (2.3)$$

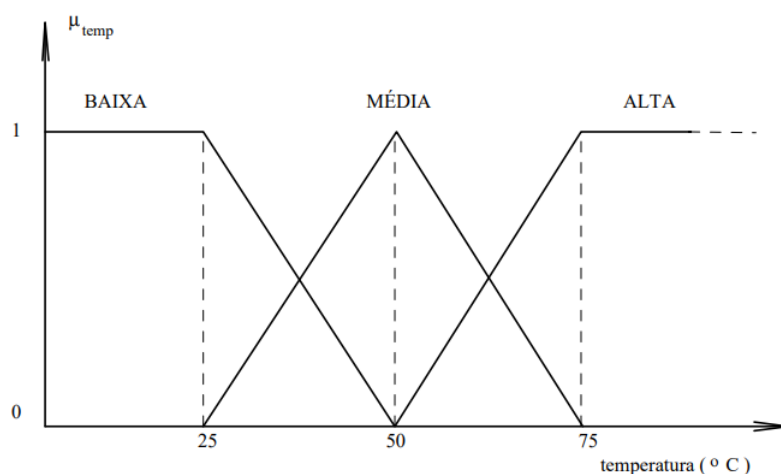
$$Gaussiana : \mu_{Gauss}(x) = e^{-\frac{(x-c)^2}{2\sigma^2}} \quad (2.4)$$

$$Tipo - Z : \mu_Z(x) = \begin{cases} 1, & \text{se } x \leq a \\ 1 - 2 \left(\frac{x-a}{b-a} \right)^2, & \text{se } a < x \leq \frac{a+b}{2} \\ 2 \left(\frac{b-x}{b-a} \right)^2, & \text{se } \frac{a+b}{2} < x < b \\ 0, & \text{se } x \geq b \end{cases} \quad (2.5)$$

$$Tipo - S : \mu_S(x) = \begin{cases} 0, & \text{se } x \leq a \\ 2 \left(\frac{x-a}{b-a} \right)^2, & \text{se } a < x \leq \frac{a+b}{2} \\ 1 - 2 \left(\frac{b-x}{b-a} \right)^2, & \text{se } \frac{a+b}{2} < x < b \\ 1, & \text{se } x \geq b \end{cases} \quad (2.6)$$

A *Fuzzificação* converte variáveis exatas em variáveis *Fuzzy* através de funções de pertinência, que associam valores reais a diferentes níveis de pertencimento a conjuntos *Fuzzy*. A Figura 5 mostra um sistema de regulação da temperatura, em que uma variável como "temperatura" pode estar ligada a conceitos linguísticos como "alta", "média" ou "baixa", com variados níveis de pertinência.

Figura 5 – Variável Linguística Temperatura.



Fonte: (GOMIDE, 1994)

Na fase de inferência, as regras *Fuzzy* são apresentadas em sentenças "se-então", onde cada uma delas estabelece uma condição e uma resposta específica. Essas normas, quando combinadas, possibilitam que o sistema reaja a uma vasta variedade de situações, mesmo quando a representação matemática precisa do sistema é complicada ou impraticável (PASSINO; YURKOVICH, 1998).

$$\text{Se } E_1 \text{ é } A_1 \text{ e } E_2 \text{ é } A_2, \dots, \text{ então } U_i = f_i(E_1, E_2, \dots) = c_i \quad (2.7)$$

Onde:

- E_1, E_2, \dots são as entradas,
- A_1, A_2, \dots são os conjuntos *Fuzzy* associados às entradas,
- c_i é o valor da saída da regra i , que pode ser uma constante ou uma função.

Para o sistema do tipo Takagi-Sugeno, as regras são expressas como funções matemáticas. A saída do processo de inferência é feita através de uma média ponderada, conforme a Equação 2.8, em que os pesos são os valores de ativação das regras.

$$U = \frac{\sum_{i=1}^N \alpha_i c_i}{\sum_{i=1}^N \alpha_i} \quad (2.8)$$

Onde:

- $\alpha_i = \min(\mu_{A_1}(E_1), \mu_{A_2}(E_2), \dots)$ é a ativação da i -ésima regra,
- c_i é o valor da saída da regra i ,
- N é o número total de regras.

Já para o sistema do tipo Mamdani, as saídas da máquina de inferência são funções *Fuzzy*, portanto, precisam da operação de *Defuzzificação*. Cada regra de inferência pode ser representada como:

$$\text{Se } E \text{ é } A_1 \text{ e } \dot{E} \text{ é } A_2, \text{ então } U \text{ é } B$$

Onde:

- E e \dot{E} são as variáveis de entrada,
- A_1 e A_2 são as funções de pertinência associadas às entradas E e \dot{E} ,
- B é a função de pertinência associada à saída U .

A ativação das regras é dada pelo valor mínimo entre a ativação das funções de pertinência das entradas, onde:

$$\alpha_i = \min(\mu_{A_1}(E), \mu_{A_2}(\dot{E}))$$

A saída de cada regra i é fuzzificada multiplicando a função de pertinência $\mu_B(U)$ pela ativação α_i :

$$\mu_{B_i}(U) = \alpha_i \mu_B(U)$$

Para o modelo do tipo Takagi-Sugeno, a defuzzificação não é necessária, devido as regras serem expressas por fórmulas matemáticas bem definidas, as suas saídas resultam em um valor *crisp*, ou seja, um valor real e exato. Já no modelo Mamdani, a defuzzificação é necessária para realizar a transformação da saída *Fuzzy* em um valor numérico específico, que pode ser utilizado para o controle do sistema. Na Figura 6, o centro de área (CDA) definido pela Equação 2.9 determina a saída através da média ponderada dos valores dos conjuntos *Fuzzy* ativados, proporcionando uma saída equilibrada e fiel às condições especificadas. O critério do

máximo (MAX) definido pela Equação 2.10, essa técnica consiste em escolher o ponto em que a função inferida tem seu máximo. A média dos máximos (MDM) como definido na Equação 2.11 representa o valor médio de todos os pontos máximos.

$$U_{\text{crisp}} = \frac{\int_{U_{\min}}^{U_{\max}} U \cdot \mu_{B_{\text{total}}}(U) dU}{\int_{U_{\min}}^{U_{\max}} \mu_{B_{\text{total}}}(U) dU} \quad (2.9)$$

Onde U_{\min} e U_{\max} são os limites do conjunto da variável de saída.

$$U_{\text{crisp}} = \arg \max \mu_{\text{total}}(U) \quad (2.10)$$

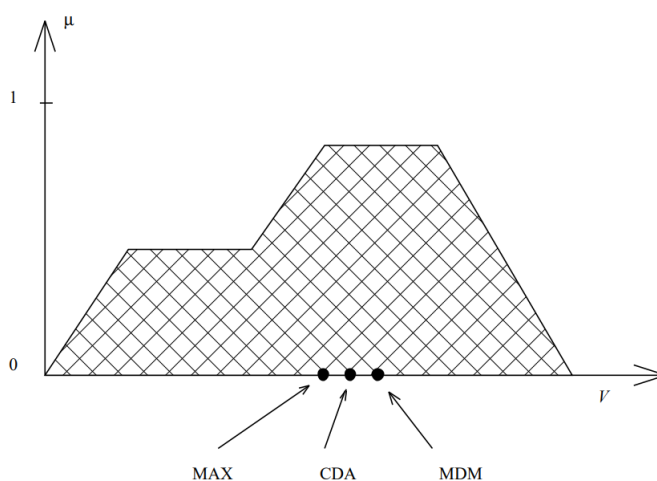
Onde $U_{\max1}$ e $U_{\max2}$ são os limites do intervalo de maior altura da função de pertinência.

$$U_{\text{crisp}} = \frac{\sum U_i \in \arg \max \mu_{\text{total}}(U)}{|\arg \max \mu_{\text{total}}(U)|} \quad (2.11)$$

Onde:

- $\arg \max \mu_{\text{total}}(U)$ é o conjunto de valores de U onde a função de pertinência $\mu_{\text{total}}(U)$ atinge seu valor máximo.
- $|\arg \max \mu_{\text{total}}(U)|$ é o número de pontos de U onde a função de pertinência atinge o máximo.
- U_{crisp} é o valor de saída crisp obtido.

Figura 6 – Métodos de Defuzzificação



Fonte: (GOMIDE, 1994)

3 METODOLOGIA

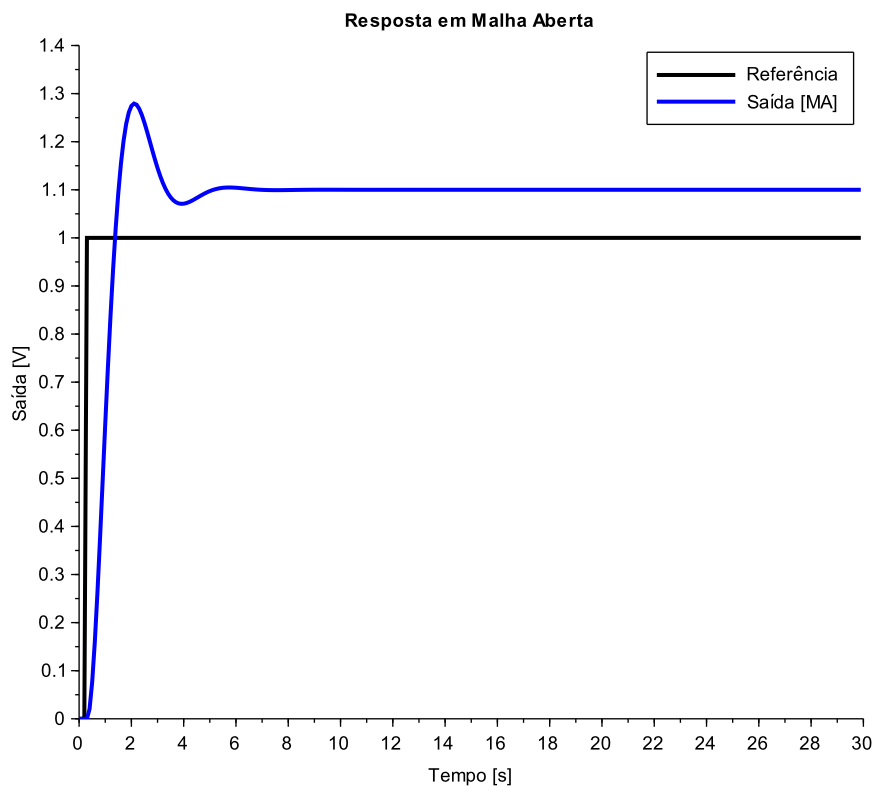
3.1 Descrição do sistema

Considerando que o objetivo da definição do sistema a ser utilizado é que ele possua características que possam ser consideradas indesejadas ao seguimento do sinal de referência, foram definidos 2 sistemas: um sistema de 2ª ordem com resposta ao degrau unitário subamortecida e um sistema de um motor DC com resposta sobreamortecida. O sistema de 2ª ordem possui a seguinte função de transferência:

$$G(s) = \frac{4.4}{s^2 + 2s + 4} \quad (3.1)$$

Esse sistema foi escolhido devido a sua resposta em malha aberta apresentar características consideradas indesejáveis para o seguimento do sinal de referência, como um sobressinal e um erro de regime permanente, conforme ilustra a Figura 7.

Figura 7 – Resposta do 1º Sistema em Malha Aberta.



Fonte: Autor

Para a discretização do sistema, foi utilizado o ZOH (*Zero Order Holder*), para isso, foi feita a seguinte operação.

$$G(z) = (1 - z^{-1})Z\left[\frac{G(s)}{s}\right] \quad (3.2)$$

O tempo de amostragem T_s foi definido como sendo o inverso da frequência de amostragem f_s , que por sua vez, é baseado na regra de Shannon-Nyquist que diz que essa frequência tem de ser pelo menos duas vezes maior que a frequência natural do sistema, como mostra a Equação 3.3.

$$f_s \geq 2 \cdot \frac{w_n}{2\pi} \quad (3.3)$$

Para garantir a amostragem suficiente para o sistema de controle, essa frequência é definida como um valor maior ou igual a 10 vezes a frequência natural do sistema. Analisando o sistema, temos que a sua frequência natural é dada por:

$$s^2 + 2\zeta w_n s + w_n^2 = s^2 + 2s + 4 \therefore w_n = 2; \zeta = 0.5; \quad (3.4)$$

Observando que $w_n = 2\pi f_n$, pode-se dizer que $10f_n \leq f_s$, de modo que fica definido um valor acima de $f_s \geq 3.183Hz$ ou $T_s \leq 0.314s$.

Utilizando o critério do tempo de estabilização para um sistema de segunda ordem, temos que:

$$t_s \approx \frac{4}{\zeta w_n} = \frac{4}{0.5 \cdot 2} = 4s. \quad (3.5)$$

Considerando que o tempo da amostragem deve ser suficientemente grande para garantir ao menos 20 amostras durante o tempo de acomodação do sistema, pode-se fazer:

$$T_s \approx \frac{t_s}{N} = \frac{4}{20} = 0.2s. \quad (3.6)$$

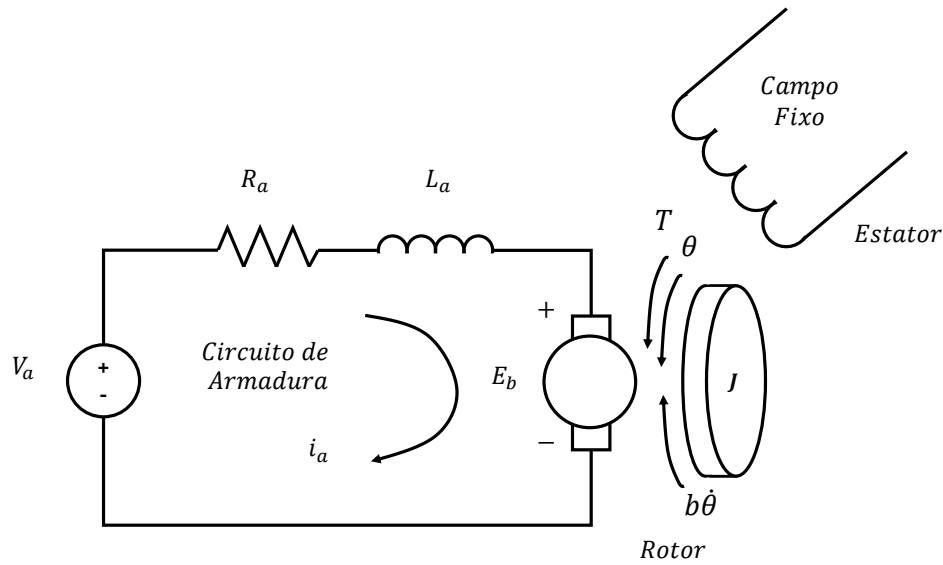
Onde N é a quantidade de amostras no intervalo, e t_s é o tempo de estabilização do sistema.

Para a simplificação dos cálculos e adequação a ambos os critérios, foi definido o valor da frequência de amostragem como sendo $f_s = 10Hz$ e, portanto, o tempo de amostragem é $T_s = 0.1s$. O sistema discretizado obtido é mostrado na Equação 3.7.

$$G(z) = \frac{0,0205362z + 0,192104}{z^2 - 1,7825975z + 0,8187308} \quad (3.7)$$

Já o 2º sistema a ser tratado foi um motor DC. Conforme apresentado por [Rahman \(2017\)](#), o seu modelo equivalente é mostrado na Figura 8.

Figura 8 – Modelo equivalente do motor DC.



Fonte: Adaptado de (RAHMAN, 2017).

Onde:

- V_a é a Tensão da armadura (V).
- R_a é a resistência da armadura (Ω).
- L_a é a indutância da armadura (H).
- i_a é a corrente da armadura (A).
- E_b é a Força contraeletromotriz (V).
- T é o Torque ($N \cdot m$).
- θ é a posição angular do eixo do rotor (rad).

Desse modo, aplicando a segunda lei de Kirchhoff, que se trata da lei das tensões, podemos chegar à Equação 3.8.

$$V_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + E_b(t) \quad (3.8)$$

Tendo em vista que a força contraeletromotriz é dada por:

$$E_b(t) = K_e \omega(t) \quad (3.9)$$

Em que $\omega(t)$ é a velocidade angular do motor e K_e é a constante de força contraeletromotriz.

Aplicando a Equação 3.9 na Equação 3.8 chegamos ao modelo elétrico do motor DC:

$$V_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + K_e \omega(t) \quad (3.10)$$

Já para o modelo mecânico do motor DC, partimos da relação do torque gerado (T) e a velocidade angular (ω). A equação que define o torque é a seguinte:

$$T(t) = K_t i_a(t) \quad (3.11)$$

Em que K_t é a constante de torque do motor.

A equação de movimento do motor é dada por:

$$T(t) = J \frac{d\omega(t)}{dt} + B\omega(t) \quad (3.12)$$

Em que J é o momento de inércia do rotor e B é o coeficiente de atrito viscoso. Aplicando a Equação 3.11 na Equação 3.12, é obtida a equação que descreve o sistema mecanicamente:

$$K_t i_a(t) = J \frac{d\omega(t)}{dt} + B\omega(t) \quad (3.13)$$

Aplicando a transformada de Laplace na Equação 3.10 e 3.13 temos:

$$V_a(s) = R_a i_a(s) + L_a s i_a(s) + K_e \omega(s) \quad (3.14)$$

$$K_t \cdot i_a(s) = J s \omega(s) + B \omega(s) \quad (3.15)$$

Isolando o termo $i_a(s)$ na Equação 3.15:

$$i_a(s) = \frac{J s \omega(s) + B \omega(s)}{K_t} \quad (3.16)$$

Aplicando a Equação 3.16 na Equação 3.14 e realizando as manipulações matemáticas, chegou-se à função de transferência do motor DC:

$$\frac{\omega(s)}{V(s)} = \frac{K_t}{(R_a + L_a s)(B + J s) + K_e K_t} \quad (3.17)$$

Os parâmetros utilizados para a simulação do 2º sistema foram extraídos de (SRIVASTAVA; PANDIT, 2012). A seleção foi feita devido a esse exemplo ser a representação de um sistema real de um motor DC. Os coeficientes do sistema em questão são mostrados na Tabela 1:

Parâmetro	Sigla	Valor
Momento de Inércia do rotor	J	0,01 $kg \cdot m^2$
Coeficiente de Atrito Viscoso	B	0,1 $N \cdot m \cdot s$
Constante de Força Contraeletromotriz	K_e	0,01 $V \cdot s/rad$
Resistência da Armadura	R_a	1 Ω
Indutância da Armadura	L_a	0,5 H
Constante de Torque	K_t	0,01 $N \cdot m/A$

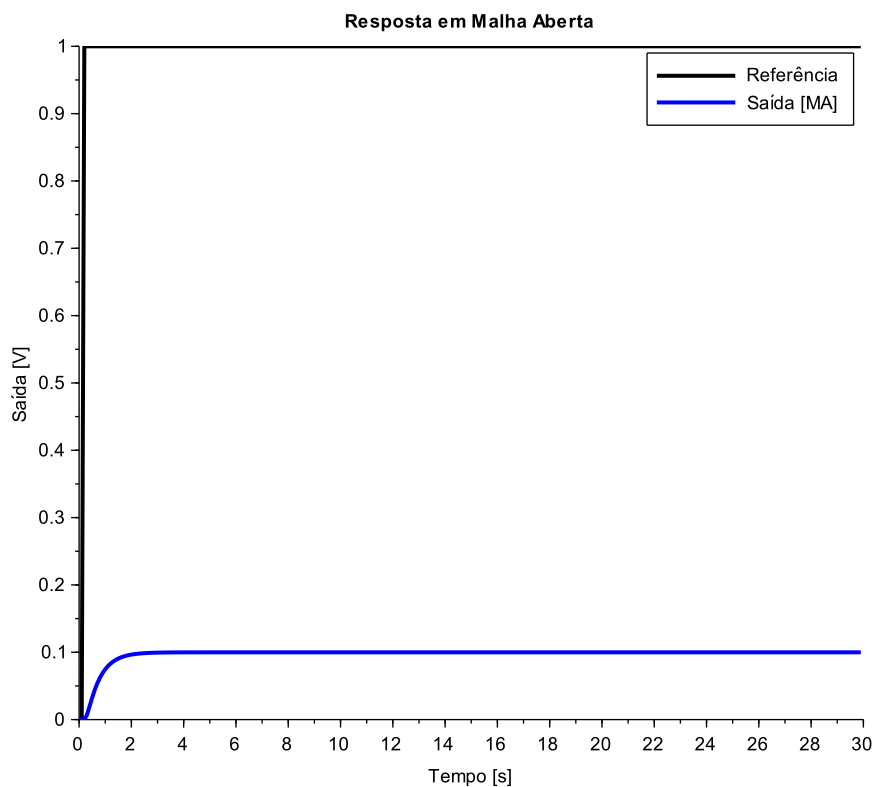
Tabela 1 – Parâmetros do Motor DC.

Substituindo os valores da Tabela 1 na Equação 3.17 temos o seguinte sistema:

$$G(s) = \frac{2}{s^2 + 12s + 20,2} \quad (3.18)$$

A resposta ao degrau unitário do sistema em malha aberta é mostrada na Figura 9, apresentando um elevado erro de regime permanente no sistema:

Figura 9 – Resposta do 2º Sistema (Motor DC) em Malha Aberta.



Fonte: Autor

A discretização do 2º sistema foi feita da mesma forma que no 1º sistema, definido o tempo de amostragem T_s igual a 0,1 s. O sistema discretizado obtido é mostrado na Equação 3.19.

$$G(z) = \frac{0,0046027z + 0,0068555}{z^2 - 1,1864975z + 0,3011942} \quad (3.19)$$

3.2 Desenvolvimento do PID

Para a discretização do controlador, foi utilizada a transformação bilinear (Tustin), que possibilita uma aproximação mais fiel da relação entre os domínios contínuo e discreto. Essa transformação preserva a estabilidade do sistema ao mapear o eixo $j\omega$ do plano s para o círculo

unitário no plano z e pode ser expressa pela Equação 3.20. Ela combina as aproximações *Forward* e *Backward*, resultando em uma representação mais precisa das dinâmicas do sistema em comparação com essas abordagens.

$$s = \frac{2}{T_s} \frac{z - 1}{z + 1} \quad (3.20)$$

Partindo do PID no domínio s temos:

$$\frac{U(s)}{E(s)} = k_p + k_i \frac{1}{s} + k_d s \quad (3.21)$$

Onde:

- $U(s)$ é o sinal do controlador no domínio s
- $E(s)$ é o sinal de erro no domínio s .
- k_p, k_i, k_d são os ganhos proporcional, integral e derivativo.

Aplicando a transformação bilinear na lei de controle que descreve o PID contínuo, conforme descrito na Equação 3.21, chega-se ao seguinte resultado.

$$\frac{U(z)}{E(z)} = k_p + k_i \frac{1}{\frac{2}{T_s} \frac{z-1}{z+1}} + k_d \frac{2}{T_s} \frac{z-1}{z+1} \quad (3.22)$$

Resolvendo a equação e organizando o sistema, pode-se chegar ao seguinte resultado:

$$\frac{U(z)}{E(z)} = \frac{(k_p + \frac{k_i T_s}{2} + \frac{2k_d}{T_s}) + (k_i T_s - \frac{4k_d}{T_s})z^{-1} + (-k_p + \frac{k_i T_s}{2} + \frac{2k_d}{T_s})z^{-2}}{(1 - z^{-2})} \quad (3.23)$$

Aplicando a transformada Z inversa, considerando que z^{-1} é o operador de atraso de tempo unitário, pode-se chegar ao seguinte resultado:

$$u(k) = u(k-2) + s_0 e(k) + s_1 e(k-1) + s_2 e(k-2) \quad (3.24)$$

Onde:

- $u(k)$ é o sinal do controlador no tempo discreto
- $s_0 = k_p + \frac{k_i T_s}{2} + \frac{2k_d}{T_s}$
- $s_1 = k_i T_s - \frac{4k_d}{T_s}$
- $s_2 = -k_p + \frac{k_i T_s}{2} + \frac{2k_d}{T_s}$

Um problema originado com a implementação desse controlador é o de que a parte derivativa gera um ruído que reverbera para o sinal de controle e, conseqüentemente, para o sinal

de saída do sistema (ÅSTRÖM; MURRAY, 2021). Para contornar esse problema, é colocado um filtro na parte derivativa do controlador.

$$C(s) = \frac{U(s)}{E(s)} = k_p + k_i \frac{1}{s} + k_d s \left(\frac{w_c}{s + w_c} \right) \quad (3.25)$$

Onde:

- w_c é a frequência de corte do filtro em rad/s

O valor dessa frequência de corte é comumente definido através da utilização do critério de Nyquist como sendo $w_c \geq 10/T_s$. Aplicando a esse novo modelo de controlador a transformação de Tustin:

$$\frac{U(z)}{E(z)} = k_p + k_i \frac{1}{\frac{2}{T_s} \frac{z-1}{z+1}} + k_d \frac{2}{T_s} \frac{z-1}{z+1} \left[\frac{w_c}{\left[\frac{2}{T_s} \frac{z-1}{z+1} \right] + w_c} \right] \quad (3.26)$$

Resolvendo a equação e organizando o sistema, pode-se chegar ao seguinte resultado:

$$\frac{U(z)}{E(z)} = k_p + \frac{k_i T_s}{2} \left(\frac{z+1}{z-1} \right) + k_d w_c \left[\frac{(z-1)}{(z-1) + \frac{w_c T_s}{2} (z+1)} \right] \quad (3.27)$$

Aplicando a transformada Z inversa, chega-se ao seguinte resultado:

$$u(k) = \frac{1}{q_0} [-q_1 u(k-1) - q_2 u(k-2) + s_0 e(k) + s_1 e(k-1) + s_2 e(k-2)] \quad (3.28)$$

Onde:

- $u(k)$ é o sinal do controlador no tempo discreto.
- $s_0 = k_p + \frac{k_p w_c T_s}{2} + \frac{k_i T_s}{2} + \frac{k_i w_c T_s^2}{4} + k_d w_c$
- $s_1 = -2k_p + k_i T_s - 2k_d w_c$
- $s_2 = k_p - \frac{k_p w_c T_s}{2} - \frac{k_i T_s}{2} + \frac{k_i w_c T_s^2}{4} + k_d w_c$
- $q_0 = 1 + \frac{w_c T_s}{2}$
- $q_1 = -2$
- $q_2 = 1 - \frac{w_c T_s}{2}$

3.3 Desenvolvimento do Controlador *Fuzzy-PID*

Partindo da Equação 2.1, que escreve o sinal de controle PID no domínio do tempo, temos os quatro termos que serão transformados em variáveis *fuzzy*, o sinal de controle, o sinal de erro, a integral do sinal de erro e a derivada do sinal de erro. Tendo isso em mente, definem-se, também, os valores operacionais dos sinais, ou seja, os limites em que os valores desses sinais variam:

- Para $u(t)$ o limite é definido como: $L_u = [-a_u, a_u]$
- Para $e(t)$ o limite é definido como: $L_e = [-a_e, a_e]$
- Para $f e(t)$ o limite é definido como: $L_i = [-a_i, a_i]$
- Para $\dot{e}(t)$ o limite é definido como: $L_d = [-a_d, a_d]$

Definindo o sistema *fuzzy* como sendo um do tipo Takagi-Sugeno e colocando m como o número de funções de pertinência dentro de uma variável *fuzzy* de entrada, considerando que todas as entradas têm a mesma quantidade m de variáveis, e que elas estão igualmente espaçadas e sendo definidas por funções triangulares, e também, afirmando que a variável de saída é *fuzzificada* através de uma quantidade $3m - 2$ de funções de pertinência do tipo constante, além de definir que, e_k, i_k, d_k e u_k são os centros dos conjuntos *fuzzy* E_k, I_k, D_k e U_k , chegamos às Equações 3.29 e 3.30.

$$e_k = \frac{(2k - m - 1)}{m - 1} a_e; i_k = \frac{(2k - m - 1)}{m - 1} a_i; d_k = \frac{(2k - m - 1)}{m - 1} a_d; 1 \leq k \leq m \quad (3.29)$$

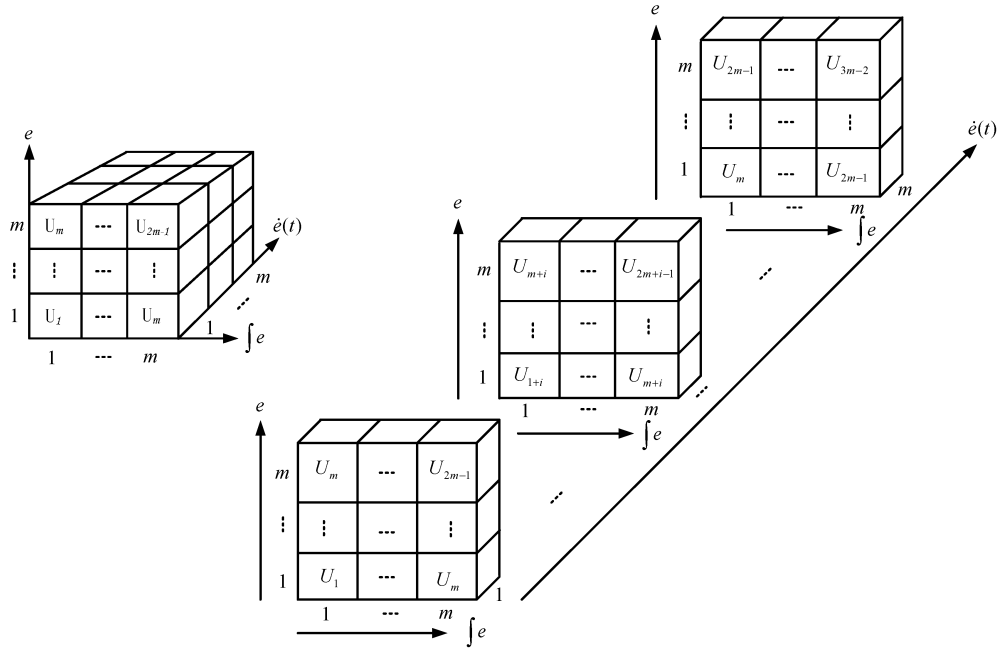
$$u_k = \frac{(2k - 3m + 1)}{3(m - 1)} a_u, 1 \leq k \leq (3m - 2) \quad (3.30)$$

Essas equações definem os valores dos intervalos nas funções de pertinência nos parâmetros *fuzzy* de entrada e os valores dos conjuntos *fuzzy* de saída. A distância entre os valores do sinal de saída e seu valor futuro é um valor Δu definido conforme a Equação 3.31.

$$\Delta u = u_{k+1} - u_k \quad (3.31)$$

Como o sistema *fuzzy* em questão se trata de um do tipo Takagi-Sugeno, a definição das regras é feita através da Equação 2.7. A associação dessas regras é definida pela estrutura SCAM (*Sliced Cube Associative Memory*), em que organiza-se o espaço das variáveis de entrada em subespaços discretos conhecidos como "fatias", conforme mostra a Figura 10, e cada subespaço corresponde a uma combinação específica dos valores das variáveis de entrada. Organizar dessa forma reduz a complexidade computacional ao facilitar a associação entre entradas e saídas (CHAO et al., 2017).

Figura 10 – SCAM (Sliced Cube Associative Memory).



Fonte: (CHAO et al., 2017)

Aplicando a inferência do tipo Takagi-Sugeno para definir o valor de saída do controlador baseado nas entradas $e(t)$, $f e(t)$ e $\dot{e}(t)$ temos a Equação 3.32.

$$U = \frac{\sum u_i (\mu_{E_i}(e(t)) \times \mu_{I_j}(f e(t))) \times \mu_{D_k}(\dot{e}(t))}{\sum (\mu_{E_i}(e(t)) \times \mu_{I_j}(f e(t))) \times \mu_{D_k}(\dot{e}(t))}, 1 \leq i, j, k \leq m, l = i + j + k - 2 \quad (3.32)$$

Considerando o menor conjunto possível de funções de pertinência, temos a entrada $e(t)$ correspondendo às funções de pertinência E_i e E_{i+1} e os graus p e $1 - p$, da mesma forma para as outras entradas temos $f e(t)$ correspondendo I_j e I_{j+1} e os graus q e $1 - q$ e $\dot{e}(t)$ correspondendo D_k e D_{k+1} e os graus r e $1 - r$, com isso, pode-se escrever as seguintes relações dos graus de pertinência p , q e r :

$$p = \frac{e_{i+1} - e(t)}{e_{i+1} - e_i}, q = \frac{i_{j+1} - f e(t)}{i_{j+1} - e_j}, r = \frac{d_{k+1} - \dot{e}(t)}{d_{k+1} - d_k} \quad (3.33)$$

Com isso, pode-se escrever as regras para o sistema:

1. SE $e(t)$ é E_i e $f e(t)$ é I_j e $\dot{e}(t)$ é D_k ENTÃO $u(t)$ é $U_{i+j+k-2}$
2. SE $e(t)$ é E_i e $f e(t)$ é I_j e $\dot{e}(t)$ é D_{k+1} ENTÃO $u(t)$ é $U_{i+j+k-1}$
3. SE $e(t)$ é E_i e $f e(t)$ é I_{j+1} e $\dot{e}(t)$ é D_k ENTÃO $u(t)$ é $U_{i+j+k-1}$
4. SE $e(t)$ é E_i e $f e(t)$ é I_{j+1} e $\dot{e}(t)$ é D_{k+1} ENTÃO $u(t)$ é U_{i+j+k}
5. SE $e(t)$ é E_{i+1} e $f e(t)$ é I_j e $\dot{e}(t)$ é D_k ENTÃO $u(t)$ é $U_{i+j+k-1}$
6. SE $e(t)$ é E_{i+1} e $f e(t)$ é I_j e $\dot{e}(t)$ é D_{k+1} ENTÃO $u(t)$ é U_{i+j+k}

7. SE $e(t)$ é E_{i+1} e $f e(t)$ é I_{j+1} e $\dot{e}(t)$ é D_k ENTÃO $u(t)$ é U_{i+j+k}

8. SE $e(t)$ é E_{i+1} e $f e(t)$ é I_{j+1} e $\dot{e}(t)$ é D_{k+1} ENTÃO $u(t)$ é $U_{i+j+k+1}$

Como mostrado na Equação 3.32 pode-se definir a saída do sistema como $u(t) = \frac{u_{num}(t)}{u_{den}(t)}$. Fazendo a média ponderada com as 8 regras, pode-se chegar às seguintes equações.

$$\begin{aligned} u_{den}(t) &= pqr + pq(1-r) + p(1-q)r \\ &+ p(1-q)(1-r) + (1-p)qr + (1-p)q(1-r) \\ &+ (1-p)(1-q)r + (1-p)(1-q)(1-r) = 1 \end{aligned} \quad (3.34)$$

$$\begin{aligned} u_{num}(t) &= pqr u_{i+j+k-2} + pq(1-r)u_{i+j+k-1} + p(1-q)r u_{i+j+k-1} \\ &+ p(1-q)(1-r)u_{i+j+k} + (1-p)qr u_{i+j+k-1} + (1-p)q(1-r)u_{i+j+k} \\ &+ (1-p)(1-q)r u_{i+j+k} + (1-p)(1-q)(1-r)u_{i+j+k+1} \end{aligned} \quad (3.35)$$

Para simplificar o resultado, aplicamos a Equação 3.31:

$$u_{num}(t) = u_x + \Delta u_{(1-p-q-r)} = u_{i+j+k+1-p-q-r} = \frac{(2(i+j+k+1-p-q-r) - 3m + 1)}{3(m-1)} a_u \quad (3.36)$$

Aplicando a Equação 3.29 na Equação 3.33, podemos escrever os graus de pertinência como sendo:

$$p = \frac{(2i - m + 1)a_e - (m - 1)e(t)}{2a_e}, \quad (3.37)$$

$$q = \frac{(2j - m + 1)a_i - (m - 1)f e(t)}{2a_i}, \quad (3.38)$$

$$r = \frac{(2k - m + 1)a_d - (m - 1)\dot{e}(t)}{2a_d} \quad (3.39)$$

Substituindo a Equação 3.39 na Equação 3.36, temos:

$$\begin{aligned} u_{num}(t) &= \frac{a_u}{3(m-1)} [2i + 2j + 2k - 3m + 3 + \\ &\frac{(m-1)e(t)}{a_e} - \frac{2ia_e}{a_e} + \frac{(m-1)a_e}{a_e} + \frac{(m-1)f e(t)}{a_i} \\ &\frac{-2ja_i}{a_i} + \frac{(m-1)a_i}{a_i} + \frac{(m-1)\dot{e}(t)}{a_d} - \frac{-2ka_d}{a_d} + \frac{(m-1)a_d}{a_d}] \\ &= \frac{a_u}{3a_e} e(t) + \frac{a_u}{3a_i} \int e(t) + \frac{a_u}{3a_d} \dot{e}(t). \end{aligned} \quad (3.40)$$

Portanto, comparando com o controlador PID, podemos dizer que:

$$k_p = \frac{a_u}{3a_e}, k_i = \frac{a_u}{3a_i}, k_d = \frac{a_u}{3a_d} \quad (3.41)$$

Onde:

- a_u é o valor máximo para o sinal de controle na saída.
- a_e é o valor máximo para o sinal de erro na entrada.
- a_i é o valor máximo para o sinal da integral do erro na entrada.
- a_d é o valor máximo para o sinal da derivada do erro na entrada.

3.4 Critérios de Avaliação de Desempenho

Para avaliar a qualidade do controle dos sistemas, é necessário o uso de índices de desempenho. Esses índices consideram critérios como amplitude, persistência e impacto do erro ao longo do tempo. Dentre diversas métricas, as escolhidas para verificação dos controladores foram: IAE, ISE, ITAE, ITSE e ISU.

3.4.1 IAE (*Integral of Absolute Error*)

O índice *Integral of Absolute Error* (IAE) é definido pela integral do valor absoluto do erro ao longo do tempo, dada pela Equação 3.42, onde $e(t)$ é o erro entre a saída do sistema e a referência (OGATA, 2010). Esse índice mede o módulo do erro acumulado ao longo do tempo, sendo amplamente utilizado por sua simplicidade e interpretação direta. O IAE aponta a magnitude total do erro, sendo particularmente útil em sistemas onde erros persistentes, independentemente de sua direção, são inaceitáveis. No entanto, não penaliza picos de erro ou erros em tempos mais tardios (ÅSTRÖM; MURRAY, 2021).

$$IAE = \int_0^{\infty} |e(t)| dt \quad (3.42)$$

3.4.2 ISE (*Integral of Squared Error*)

O *Integral of Squared Error* (ISE) é calculado pela integral do erro elevado ao quadrado, descrito pela Equação 3.43 (OGATA, 2010). Essa métrica prioriza erros de maior magnitude, dado que o erro é elevado ao quadrado antes da integração, atribuindo maior peso a desvios significativos (DORF; BISHOP, 2009). O ISE é útil para sistemas onde é crucial minimizar grandes erros, mas pode ser menos adequado quando picos momentâneos não são tão críticos, já que esses podem influenciar significativamente o índice (ÅSTRÖM; MURRAY, 2021).

$$ISE = \int_0^{\infty} e(t)^2 dt \quad (3.43)$$

3.4.3 ITAE (*Integral of Time-weighted Absolute Error*)

O *Integral of Time-weighted Absolute Error* (ITAE) é definido conforme a Equação 3.44 onde o erro absoluto é ponderado pelo tempo (ÅSTRÖM; MURRAY, 2021). Essa métrica

penaliza erros que ocorrem em tempos mais tardios, incentivando respostas rápidas do sistema de controle. O ITAE é frequentemente utilizado em aplicações onde se busca minimizar o tempo de acomodação e reduzir os erros residuais (OGATA, 2010). Ele fornece uma medida direta da eficácia temporal do controlador, sendo ideal em situações que exigem estabilidade rápida (DORF; BISHOP, 2009).

$$ITAE = \int_0^{\infty} t \cdot |e(t)| dt \quad (3.44)$$

3.4.4 ITSE (*Integral of Time-weighted Squared Error*)

O índice *Integral of Time-weighted Squared Error* (ITSE) é expresso pela Equação 3.45, combinando os conceitos de ponderação temporal do ITAE com o peso adicional de erros maiores fornecido pelo ISE (ÅSTRÖM; MURRAY, 2021). Essa métrica é particularmente útil em sistemas críticos, onde tanto a magnitude quanto a persistência dos erros precisam ser controladas rigorosamente (OGATA, 2010). O ITSE fornece uma avaliação detalhada do desempenho do sistema, considerando tanto a magnitude dos erros quanto o impacto do tempo em que ocorrem (DORF; BISHOP, 2009).

$$ITSE = \int_0^{\infty} t \cdot e(t)^2 dt \quad (3.45)$$

3.4.5 ISU (*Integral of Squared Control*)

O *Integral of Squared Control* (ISU) é definido pela Equação 3.46, onde $u(t)$ representa o sinal de controle aplicado ao sistema (ÅSTRÖM; MURRAY, 2021). Esse índice avalia o esforço total do controlador ao longo do tempo, medindo a energia associada ao sinal de controle (OGATA, 2010). O ISU é particularmente útil em sistemas onde o consumo de energia ou a limitação de esforço do atuador são aspectos críticos. Ao minimizar o ISU, busca-se projetar controladores que utilizem menos energia, garantindo maior eficiência e prolongando a vida útil dos atuadores (DORF; BISHOP, 2009).

$$ISU = \int_0^{\infty} u(t)^2 dt \quad (3.46)$$

3.4.6 J (Função de Custo)

A função de custo é definida pela Equação 3.47, onde $e(t)$ representa o sinal de erro e $u(t)$ representa o sinal de controle, α representa o peso de ISE e β representa o peso de ISU. Esse índice representa o consumo de energia dos sinais ponderados pelos índices α e β (OGATA, 2010). Para o uso desse índice no trabalho, foi considerado que os valores das ponderações são unitários. Minimizar a função custo ajuda a equilibrar dois objetivos críticos e frequentemente conflitantes: precisão no rastreamento de referências e eficiência energética do sistema.

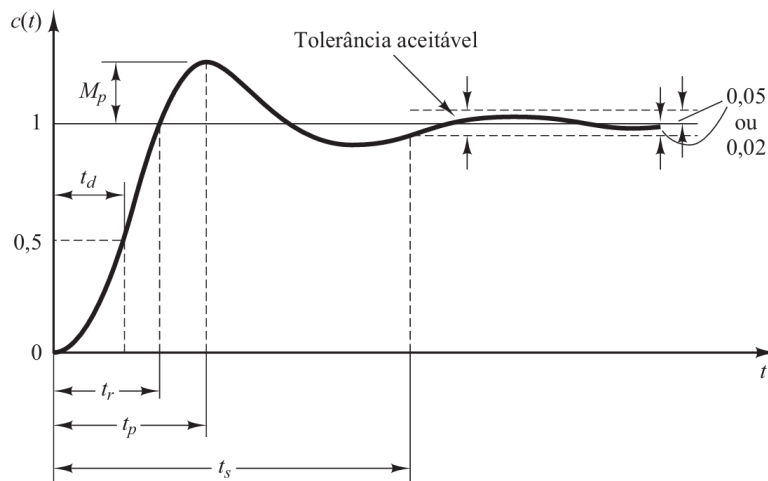
$$J = \alpha \text{ISE} + \beta \text{ISU} = \int_0^{\infty} \alpha e(t)^2 + \beta u(t)^2 dt \quad (3.47)$$

Também são utilizados outros critérios relacionados às especificações da resposta transitória conforme as definições postas por Ogata. O Tempo de Subida t_r é o tempo em que a resposta ao degrau do sistema passa de 10% a 90% do valor final para sistemas com características de 1ª ordem ou de 0% a 100% do valor final para sistemas com aspectos de 2ª ordem subamortecidos. Como o 1º sistema em malha aberta possui particularidades de 2ª ordem subamortecida, o tempo de subida utilizado é o de 0% a 100%. Já para os outros casos, foi usado o tempo de subida de 10% a 90%. O tempo de pico t_p é o tempo em que a resposta do sistema atinge o primeiro pico do sobressinal. O sobressinal (*overshoot*) M_p é definido como sendo o valor de pico máximo percentual do sinal de saída da resposta do sistema. É definido conforme a Equação 3.48.

$$M_p = \frac{y(t_p) - y(\infty)}{y(\infty)} \times 100\% \quad (3.48)$$

Tempo de acomodação t_s é o tempo para que a resposta da planta alcance valores em torno da faixa de 2% a 5% do valor final. Para a análise de desempenho neste trabalho, a faixa considerada foi de 2%. Erro de regime permanente é a diferença do valor de referência para a resposta da planta quando passada a fase transitória do sistema. Todos esses índices são mostrados na Figura 11.

Figura 11 – Índices de especificação do Sistema.



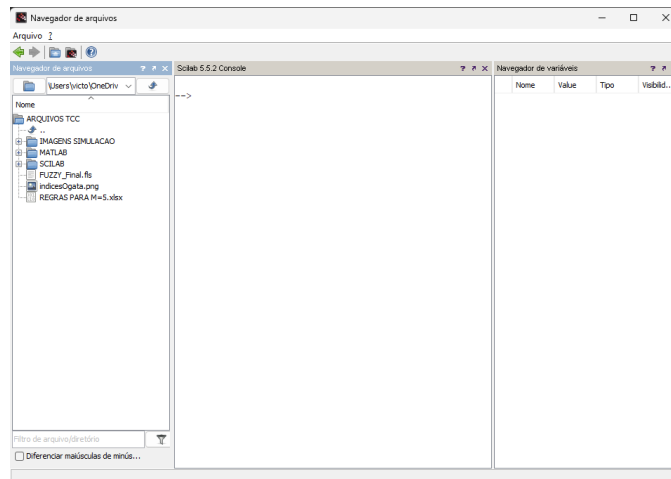
Fonte: (OGATA, 2010)

3.5 Ferramentas de Simulação

Para realizar a simulação do sistema e dos controladores, foi utilizado o *software* SCILAB (ENTERPRISES, 2024). O programa oferece diversas ferramentas úteis para a análise de sistemas, além de ser *Open Source* (código aberto), portanto, de fácil acesso, o que facilita a

reprodução dos resultados obtidos no trabalho, além da familiaridade já existente devido ao uso em diversas aulas durante a graduação. O ambiente de simulação do Scilab é mostrado na Figura 12.

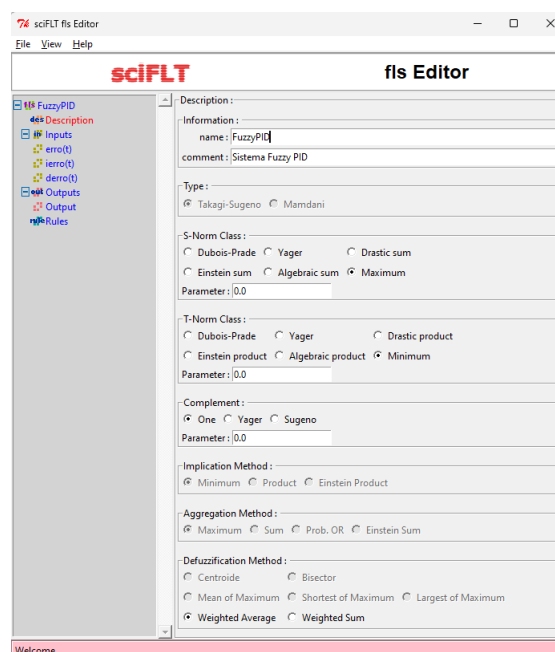
Figura 12 – Ambiente de Simulação Scilab.



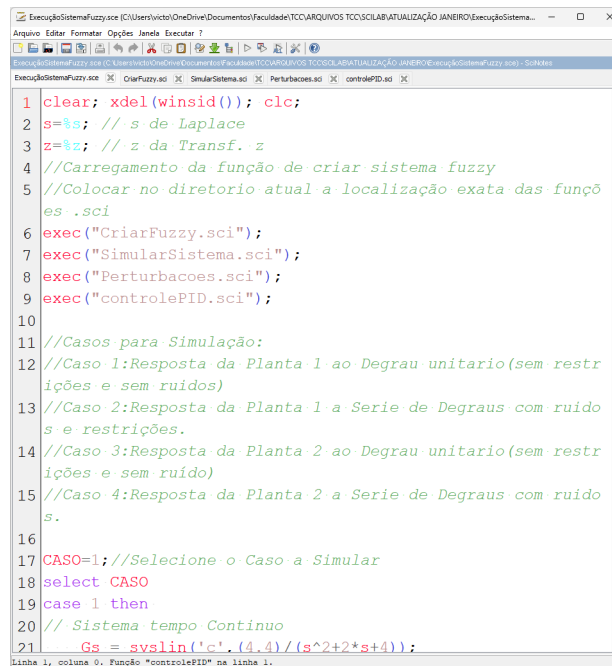
Fonte: Autor

Foi utilizada uma biblioteca chamada *Fuzzy Logic Toolbox*. Essa biblioteca adiciona ao Scilab funcionalidades que ajudam na criação e simulação de sistemas, além de uma interface gráfica para a criação desses sistemas. A Figura 13 mostra o ambiente gráfico da *toolbox*. Para simulação, foi feito um código no ambiente de criação de *scripts* do Scilab. Esse ambiente é mostrado na Figura 14.

Figura 13 – Ambiente gráfico da biblioteca SCIFLT.



Fonte: Autor

Figura 14 – Ambiente de criação de *scripts*.

```
1 clear; xdel(winsid()); clc;
2 s=%s; // s de Laplace
3 z=%z; // z da Transf. z
4 //Carregamento da função de criar sistema fuzzy
5 //Colocar no diretório atual a localização exata das funções .sci
6 exec("CriarFuzzy.sci");
7 exec("SimularSistema.sci");
8 exec("Perturbacoes.sci");
9 exec("controlePID.sci");
10
11 //Casos para Simulação:
12 //Caso 1:Resposta da Planta 1 ao Degrau unitario(sem restrições e sem ruidos)
13 //Caso 2:Resposta da Planta 1 a Serie de Degraus com ruído s e restrições.
14 //Caso 3:Resposta da Planta 2 ao Degrau unitario(sem restrições e sem ruído)
15 //Caso 4:Resposta da Planta 2 a Serie de Degraus com ruídos.
16
17 CASO=1; //Selezione o Caso a Simular
18 select CASO
19 case 1 then
20 // Sistema tempo Continuo
21 Gs = syslin('c', (4.4)/(s^2+2*s+4));
```

Linha 1, coluna 0. Função "controlePID" na linha 1.

Fonte: Autor

4 RESULTADOS E DISCUSSÃO

Neste capítulo serão discutidos os resultados obtidos através das simulações feitas. Para facilitar o entendimento das dinâmicas e melhorar a avaliação dos controladores, as simulações foram divididas em 4 casos. O 1º caso simulado é a resposta da planta de segunda ordem ao degrau unitário sem saturação no sinal do controlador e sem a inserção de ruídos. No 2º caso simulado, é feita a resposta dessa mesma planta a uma série de degraus, porém, com uma perturbação de carga da ordem de 10% do valor da referência e adição de um ruído Gaussiano branco com variância de 0,0002 na entrada do sistema, junto com a limitação do controlador a valores entre 0 e 5. O 3º caso é semelhante ao 1º, porém, com o sistema do Motor DC reagindo ao degrau unitário sem saturação no sinal do controlador e sem a inserção de ruídos. O 4º caso, similarmente ao 2º, é a resposta do Motor DC a uma série de degraus com a inserção de uma perturbação de carga com valor de 10% do sinal de referência inserida na metade do tempo da simulação e um ruído Gaussiano branco com variância 0,0002.

Para os casos relacionados ao 1º sistema, os valores dos ganhos k_p , k_i e k_d foram definidos por tentativa e erro como sendo 0,1, 0,5 e 0,05, respectivamente. Já para o 2º sistema, os valores dos ganhos do controlador foram extraídos de (SRIVASTAVA; PANDIT, 2012), sendo o ganho proporcional de 6, o ganho integral de 10 e o ganho derivativo de 0,5556. Para todos os casos, foi definido que o controlador *Fuzzy-PID* teria $m = 5$, ou seja, 5 funções de pertinência nas variáveis *fuzzy* de entrada, tendo $3m - 2$, ou seja, 13 conjuntos para a saída *fuzzy*.

Os conjuntos de Regras Fuzzy a serem usados para o 1º e 2º sistemas são descritos nas tabelas 2 a 6.

$e(t) \setminus \int e(t)$	N	NM	Z	PM	P
N	P	PL	PML	PM	PMS
NM	PL	PML	PM	PMS	PS
Z	PML	PM	PMS	PS	Z
PM	PM	PMS	PS	Z	NS
P	PMS	PS	Z	NS	NMS

Tabela 2 – Tabela para $\dot{e}(t) = N$

$e(t) \setminus f e(t)$	N	NM	Z	PM	P
N	PL	PML	PM	PMS	PS
NM	PML	PM	PMS	PS	Z
Z	PM	PMS	PS	Z	NS
PM	PMS	PS	Z	NS	NMS
P	PS	Z	NS	NMS	NM

Tabela 3 – Tabela para $\dot{e}(t) = NM$

$e(t) \setminus f e(t)$	N	NM	Z	PM	P
N	PML	PM	PMS	PS	Z
NM	PM	PMS	PS	Z	NS
Z	PMS	PS	Z	NS	NMS
PM	PS	Z	NS	NMS	NM
P	Z	NS	NMS	NM	NML

Tabela 4 – Tabela para $\dot{e}(t) = Z$

$e(t) \setminus f e(t)$	N	NM	Z	PM	P
N	PM	PMS	PS	Z	NS
NM	PMS	PS	Z	NS	NMS
Z	PS	Z	NS	NMS	NM
PM	Z	NS	NMS	NM	NML
P	NS	NMS	NM	NML	NL

Tabela 5 – Tabela para $\dot{e}(t) = PM$

$e(t) \setminus f e(t)$	N	NM	Z	PM	P
N	PMS	PS	Z	NS	NMS
NM	PS	Z	NS	NMS	NM
Z	Z	NS	NMS	NM	NML
PM	NS	NMS	NM	NML	NL
P	NMS	NM	NML	NL	N

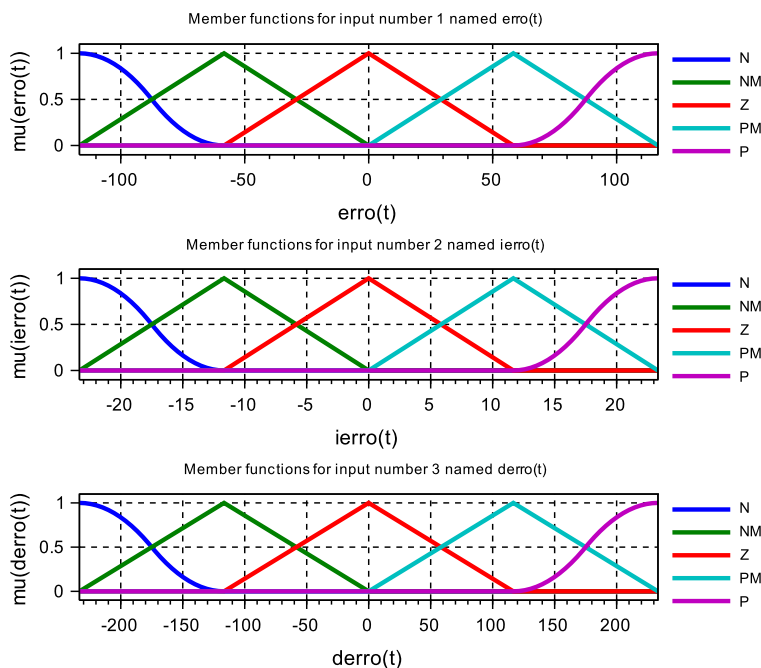
Tabela 6 – Tabela para $\dot{e}(t) = P$.

Definido isso, foram feitas as simulações conforme a ordem apresentada.

4.1 1º Caso: Resposta ao degrau unitário do 1º Sistema

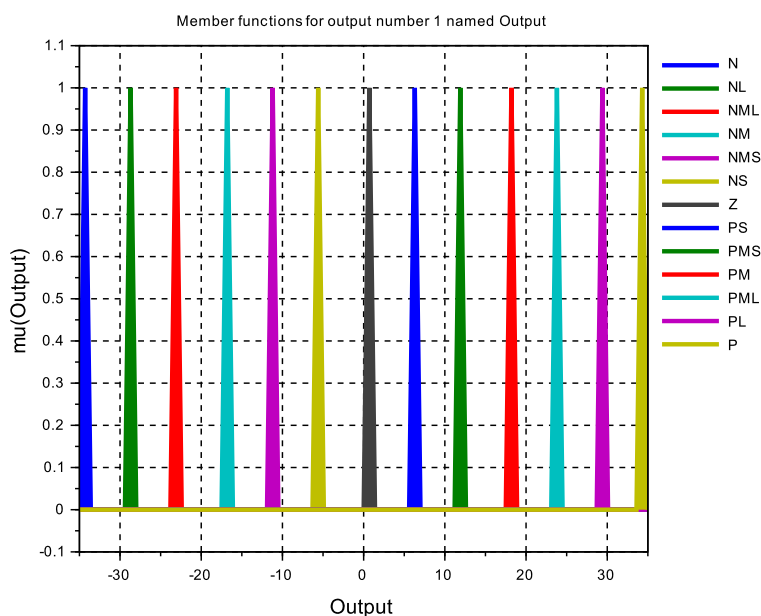
As variáveis de entrada e saída do controlador *Fuzzy-PID* para o 1º sistema são mostradas na Figura 15 e na Figura 16.

Figura 15 – Conjuntos *Fuzzy* de entrada para o 1º Sistema.



Fonte: Autor

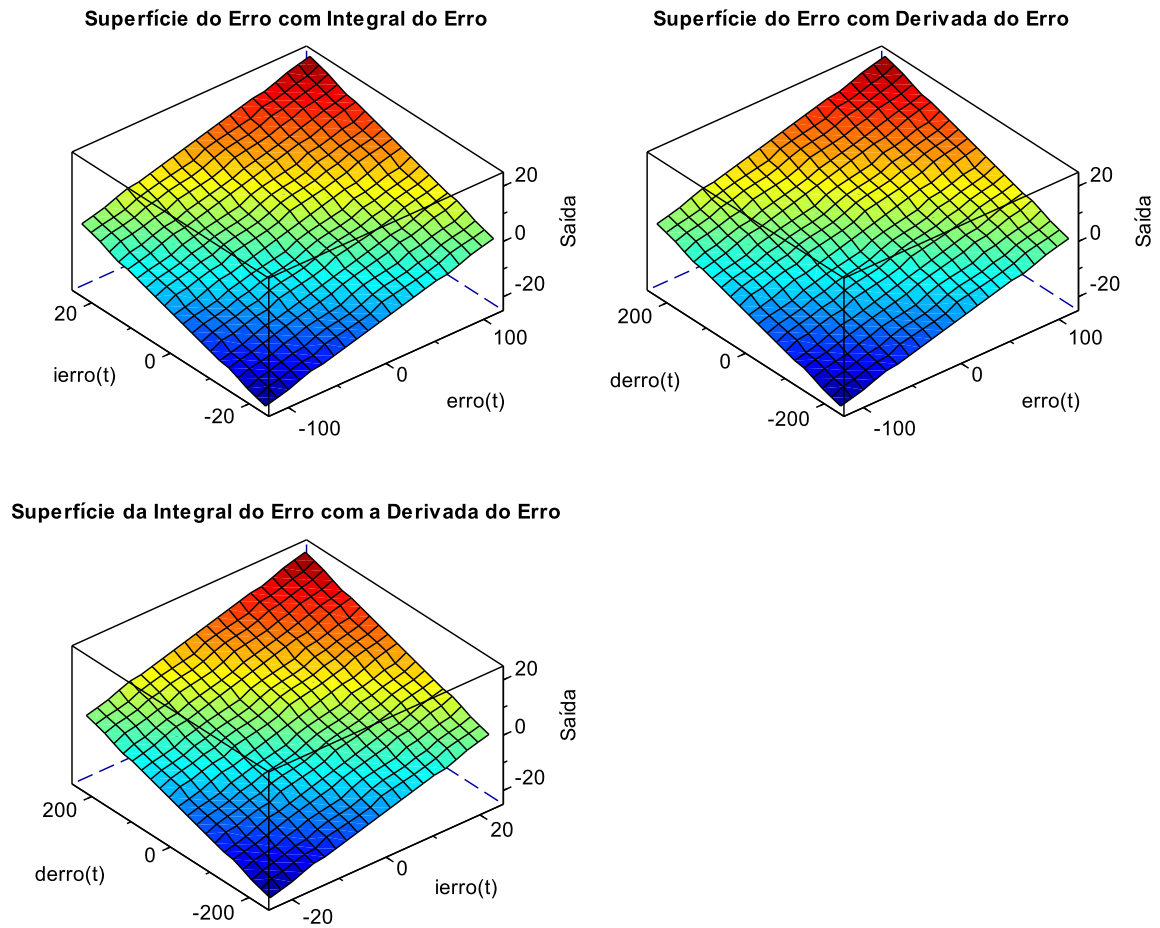
Figura 16 – Conjuntos *Fuzzy* de saída para o 1º Sistema.



Fonte: Autor

A superfície de controle para o 1º sistema é mostrada na Figura 17.

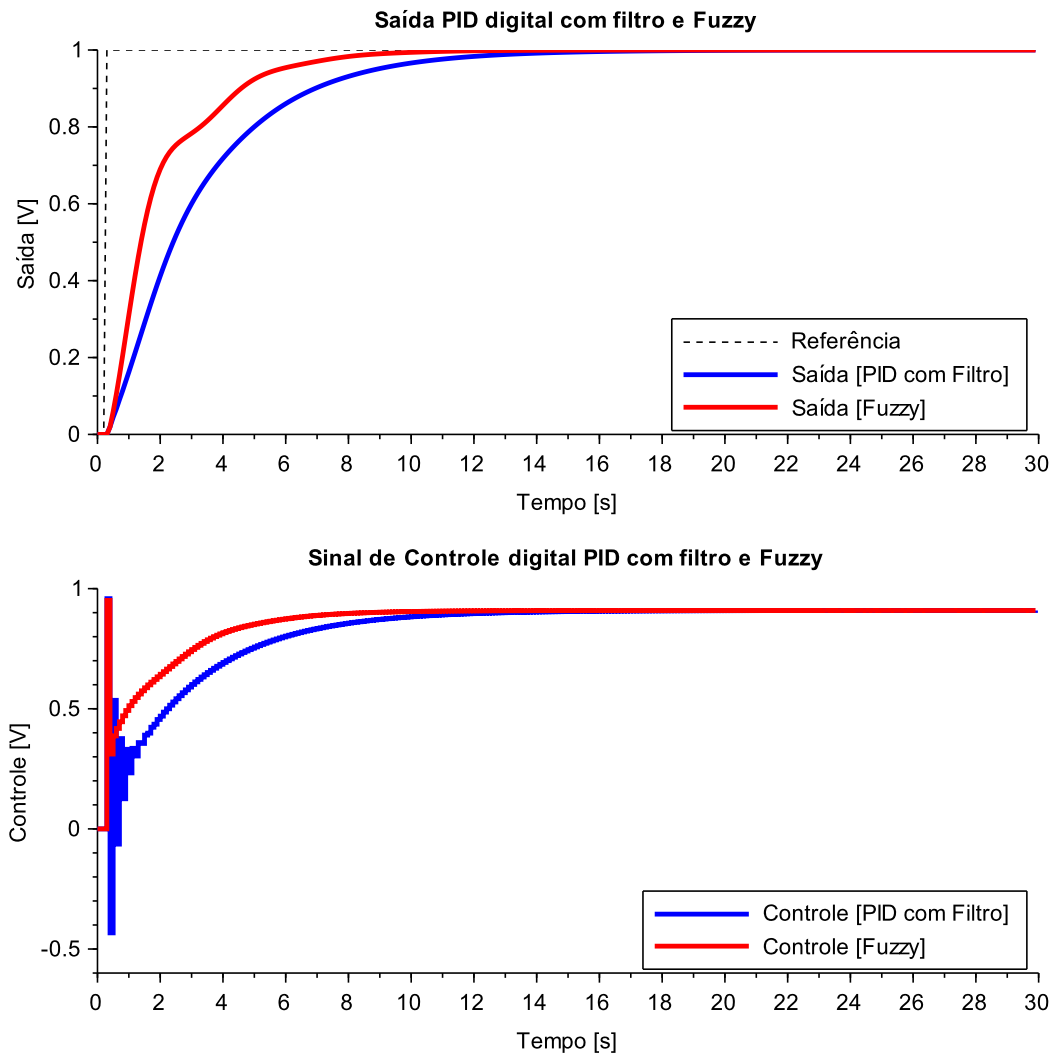
Figura 17 – Superfícies de Controle *Fuzzy* para o 1º Sistema.



Fonte: Autor.

A Figura 18 mostra a resposta ao degrau unitário do sistema controlado pelo PID digitalizado pela aproximação de Tustin com filtro derivativo e o sistema controlado pelo controlador *Fuzzy* baseado no PID. Também se tem como resultado da simulação o sinal de controle digital dos controladores.

Figura 18 – Resposta ao degrau unitário do 1º sistema com os controladores.

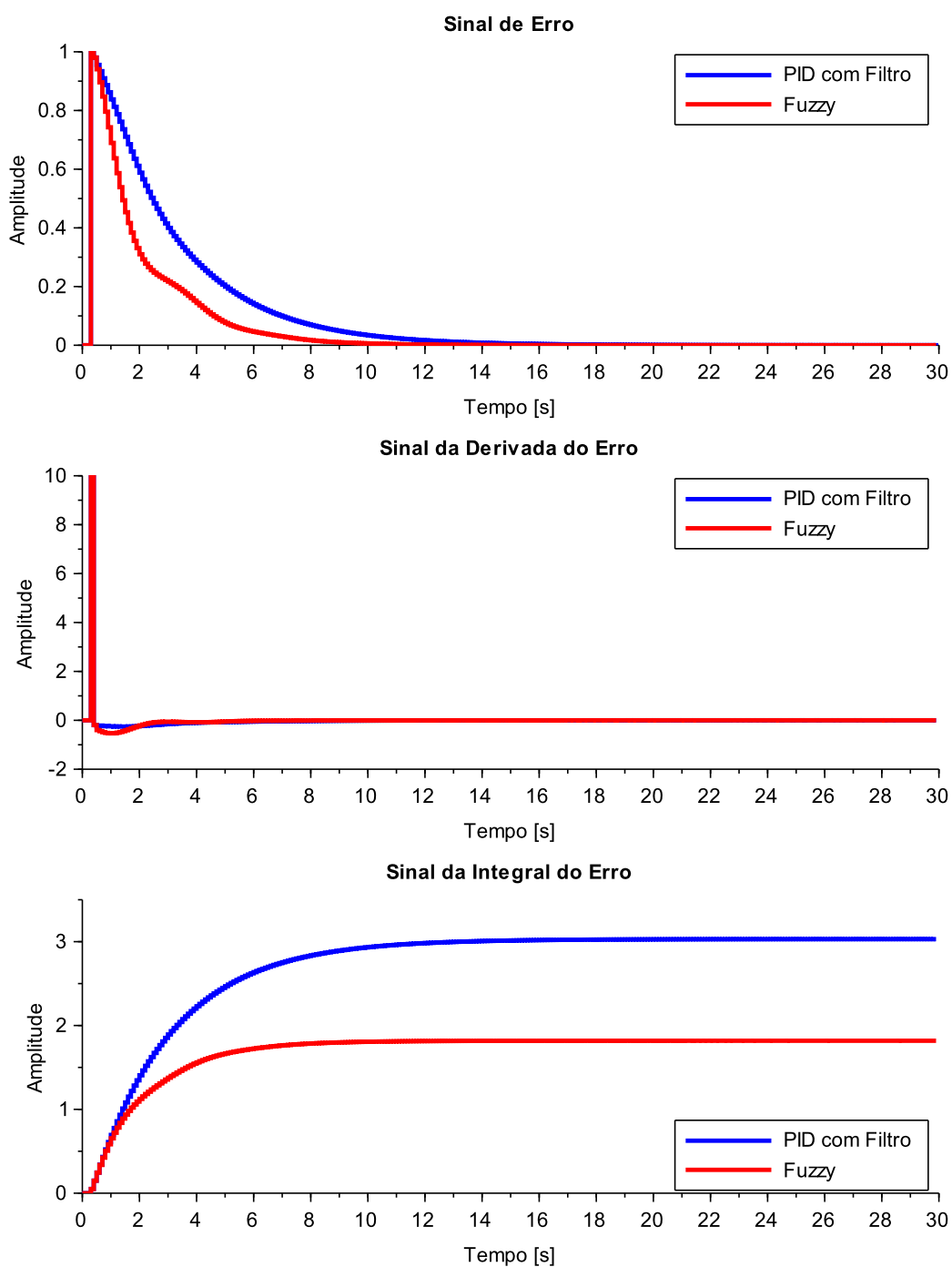


Fonte: Autor

A resposta apresentada pelo sistema controlado pelo *Fuzzy-PID* mostra um desempenho melhor em relação ao PID em termos de capacidade de resposta, uma vez que o sistema conseguiu seguir o sinal de referência em um tempo menor que o PID. Entretanto, o sinal de controle necessário para imprimir essa dinâmica no sistema acabou sendo maior que o do PID.

Os sinais do erro, da derivada do erro e da integral do erro são mostrados na Figura 19.

Figura 19 – Sinais de Erro do 1º Sistema.



Fonte: Autor

Outro resultado da simulação são os índices de desempenho, como mostra a Tabela 7, em que os valores destacados são os que apresentam melhor performance.

Método de Controle	IAE	ISE	ISU	J	ITAE	ITSE
$PID_{filtrado}$	30.302	16.250	214.354	230.604	93.013	27.192
$Fuzzy - PID$	18.182	9.282	227.796	237.078	38.543	10.140

Tabela 7 – Comparação dos índices de desempenho entre os controladores PID com Filtro e $Fuzzy-PID$ para o 1º Caso.

Os índices IAE, ISE, ITAE e ITSE do controlador $Fuzzy-PID$ apresentaram, respectivamente, o valor de 66,662%, 75,078%, 141,321%, 168,165% dos índices do PID. Ao mesmo tempo, para conseguir imprimir esses resultados, foi necessário um esforço 5,901% maior no sinal de controle, indicado pelo índice ISU do $Fuzzy-PID$, e um gasto total de energia 2,731% maior, exposto por J.

Já para as especificações de resposta transitória, temos os resultados mostrados na Tabela 8. Os valores destacados foram os que apresentaram melhor desempenho.

Índice	PID c/ Filtro	$Fuzzy-PID$
Tempo de Subida t_r	6,2 s	4,0 s
Tempo de Acomodação t_s	11,5 s	7,8 s
Variância de $y(t)$	0,052	0,036
Variância de $u(t)$	0,034	0,017
Sobressinal M_p	Não	Não
Erro de Regime Permanente	Não	Não

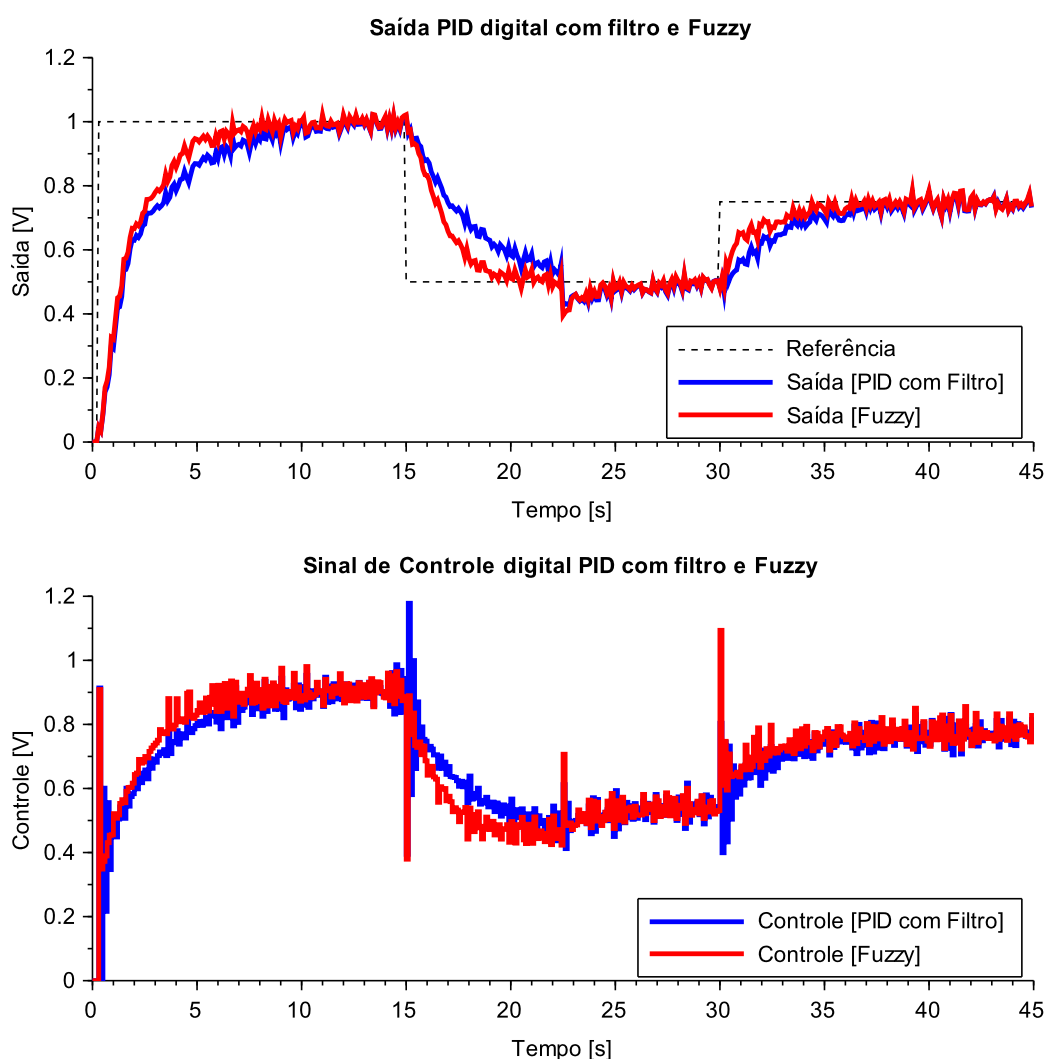
Tabela 8 – Comparação das especificações de resposta transitória dos controladores para o 1º Caso.

Os resultados do 1º caso simulado mostram que o controlador $Fuzzy-PID$ apresentou uma resposta mais rápida, conforme explicita o tempo de subida e de acomodação. Seu sinal de saída e o de controle apresentaram menor variância, cerca de 44,198% e 100,668% do valor do PID.

4.2 2º Caso: Resposta a série de degraus do 1º Sistema com ruídos e saturação.

As variáveis de entrada e saída do controlador *Fuzzy-PID* e a superfície de controle para essa simulação são as mesmas mostradas na Figura 15, Figura 16 e na Figura 17. A Figura 20 mostra a resposta ao degrau unitário do sistema controlado pelo PID e o sistema controlado pelo controlador *Fuzzy-PID* e o sinal de controle dos mesmos.

Figura 20 – Resposta a série de degraus do 1º Sistema com ruídos e saturação.



Fonte: Autor.

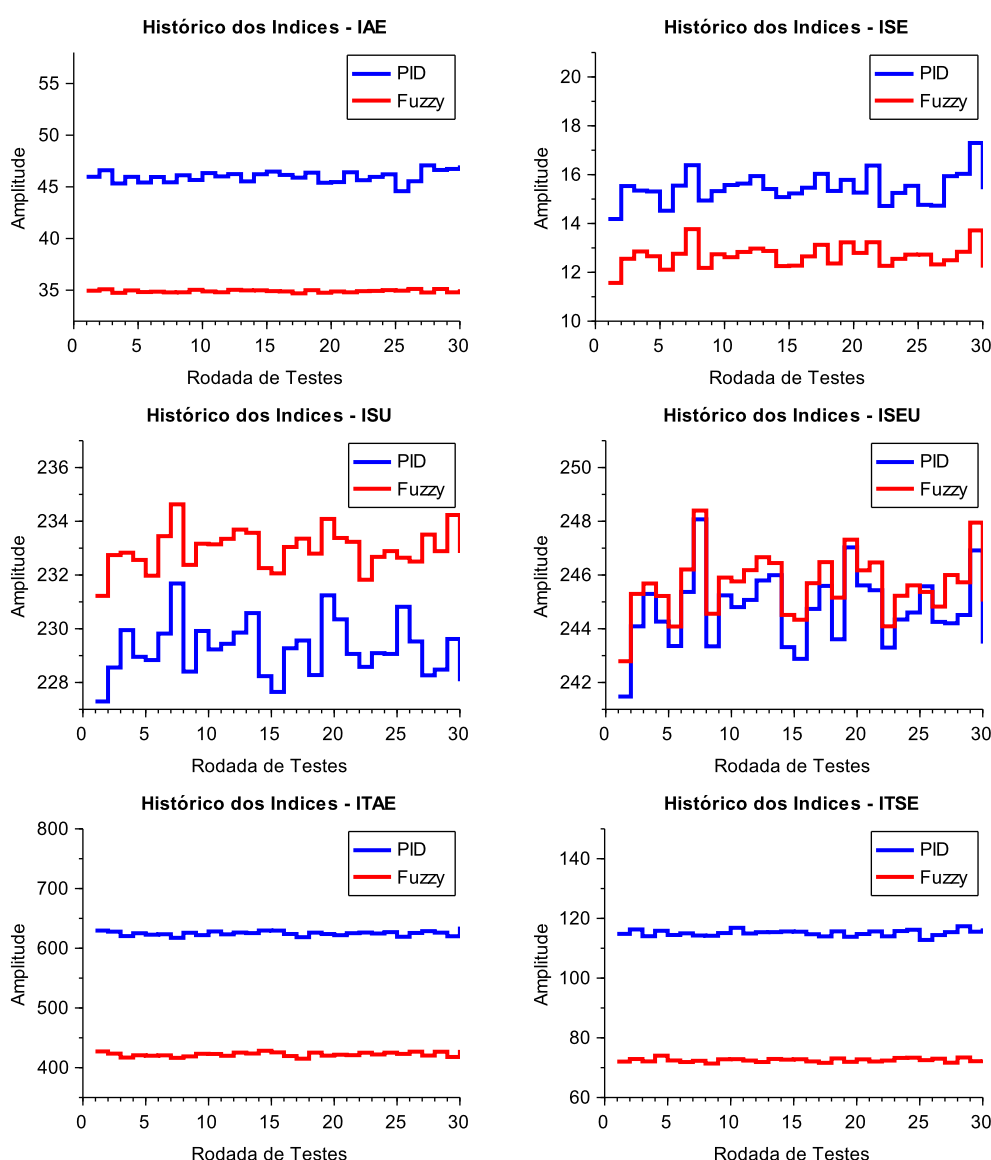
A resposta apresentada pelo sistema controlado pelo *Fuzzy-PID* mostra um desempenho melhor em relação ao PID. O sinal de controle do *Fuzzy-PID* foi maior que o do PID. As variâncias dos sinais de saída em média foram 0,0386 e 0,0423 para o PID e o *Fuzzy-PID*, respectivamente, já para o sinal de controle foram 0,025 e 0,028. Como no caso o sistema foi simulado com a adição de um ruído estocástico, foi feita uma média para definir os valores dos

índices na Tabela 9. O histórico desses índices é mostrado na Figura 21.

Método de Controle	IAE	ISE	ISU	J	ITAE	ITSE
$PID_{filtrado}$	46,014	15,463	229,256	224,719	625,022	115,149
$Fuzzy - PID$	34,911	12,675	232,963	245,638	422,411	72,512

Tabela 9 – Comparação da média dos índices de desempenho entre os controladores PID com Filtro e $Fuzzy-PID$ para o 2º Caso.

Figura 21 – Variação dos índices de desempenho do 2º Caso.



Fonte: Autor.

Os índices IAE, ISE, ITAE e ITSE do controlador $Fuzzy-PID$ apresentaram, respectivamente, o valor de 31,806%, 21,993%, 47,965%, 58,799% dos índices do PID. Ao mesmo tempo, para conseguir imprimir esses resultados, foi necessário um esforço 1,55% maior no sinal

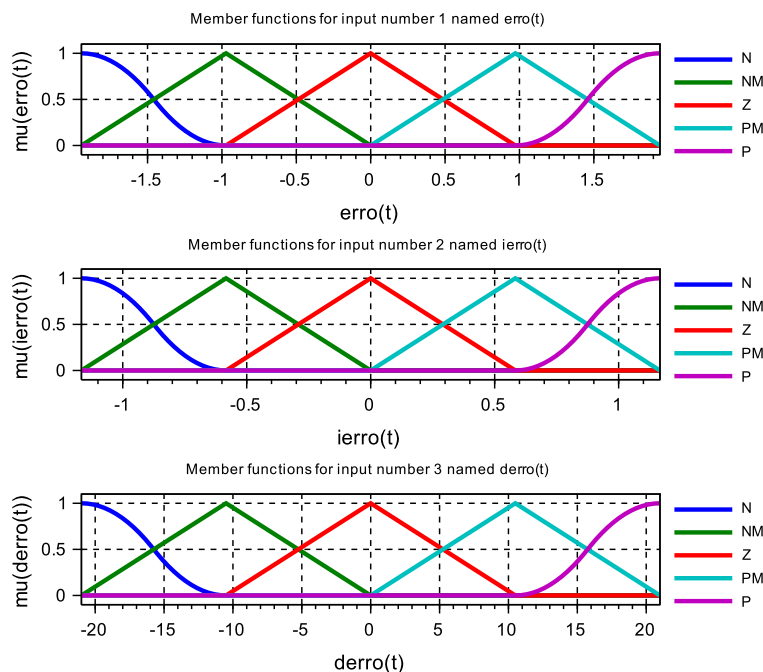
de controle, indicado pelo índice ISU do *Fuzzy-PID* e um gasto total de energia 0,374% maior exposto por J.

Os resultados do 2º caso simulado mostram que, com o controlador *Fuzzy-PID*, o sistema apresentou uma resposta mais rápida, seu sinal de saída e o de controle apresentaram, em média, uma maior variância, comparados ao PID, cerca de 9,7% maior para o sinal de saída e 10,66% maior para o sinal de controle.

4.3 3º Caso: Resposta ao degrau unitário do 2º Sistema.

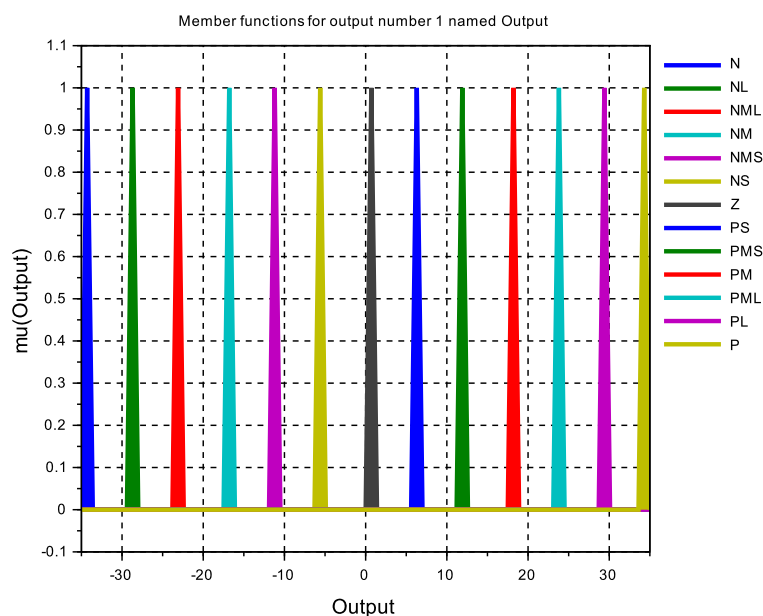
As variáveis de entrada e saída do controlador *Fuzzy-PID* para o 2º sistema são mostradas na Figura 22 e na Figura 23.

Figura 22 – Conjuntos *Fuzzy* de entrada para o 2º Sistema.



Fonte: Autor

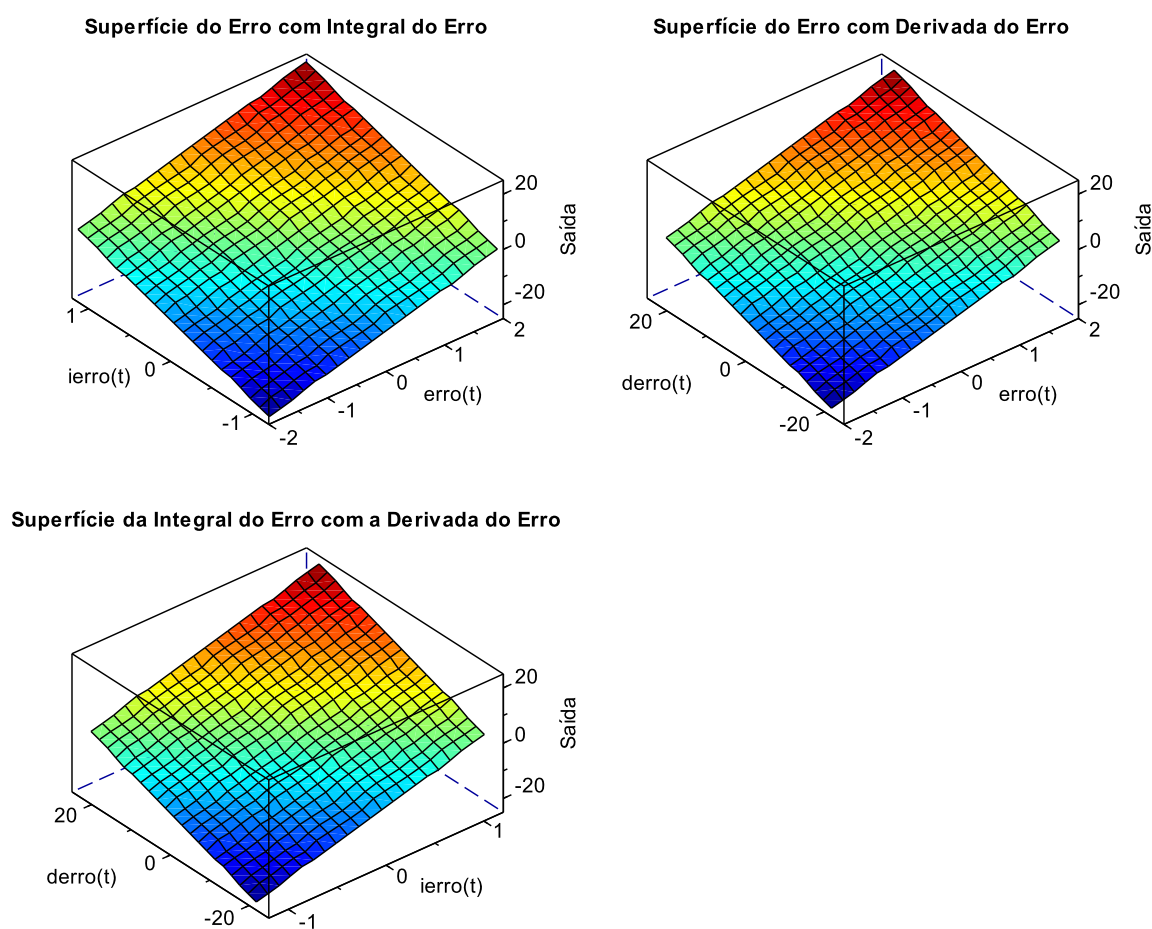
Figura 23 – Conjuntos *Fuzzy* de saída para o 2º Sistema.



Fonte: Autor.

As superfícies de controle para o 2º sistema são mostradas na Figura 24.

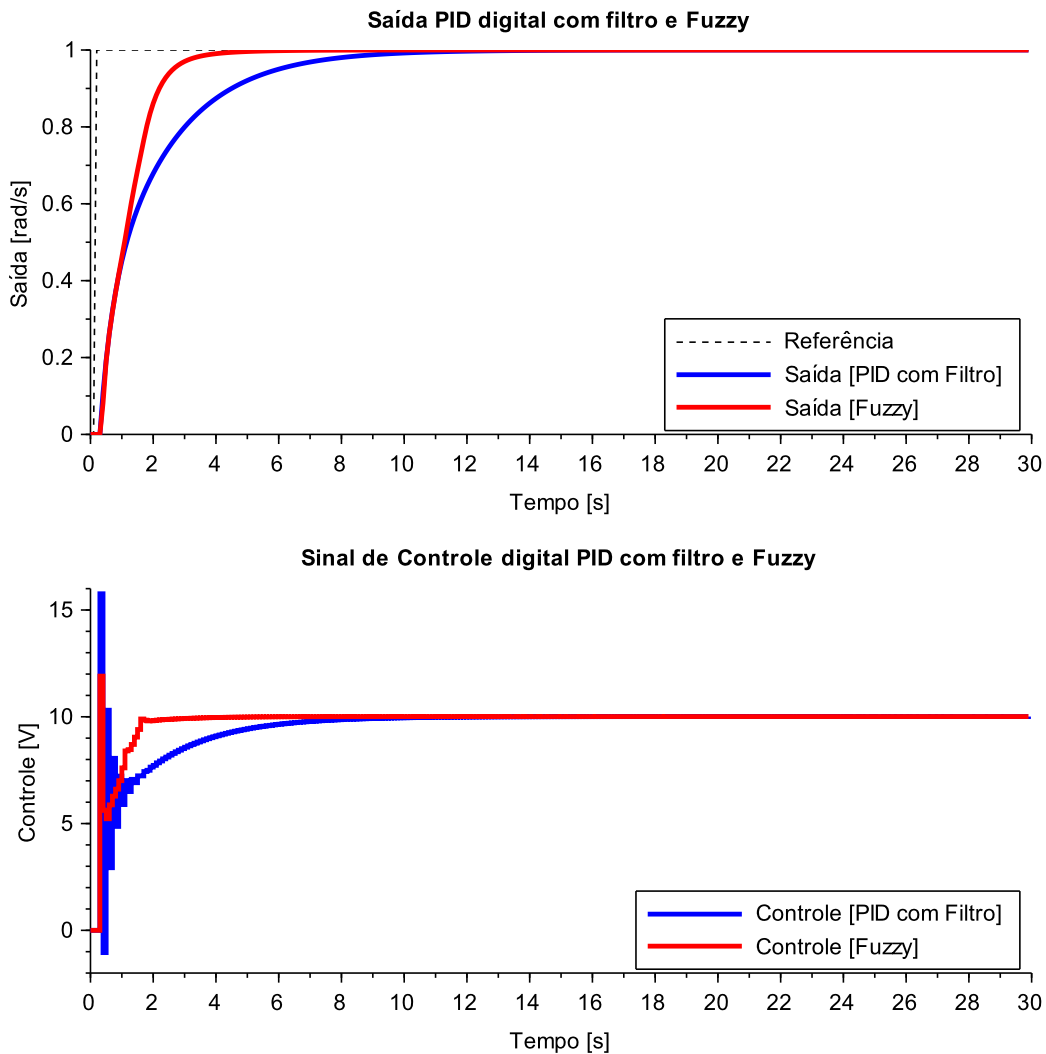
Figura 24 – Superfícies de Controle *Fuzzy* para o 2º Sistema.



Fonte: Autor.

A Figura 25 mostra a resposta ao degrau unitário do sistema controlado pelo PID digitalizado pela aproximação de Tustin com filtro derivativo e o sistema controlado pelo controlador *Fuzzy* baseado no PID. Também se tem como resultado da simulação o sinal de controle digital dos controladores.

Figura 25 – Resposta ao degrau unitário do 2º Sistema.

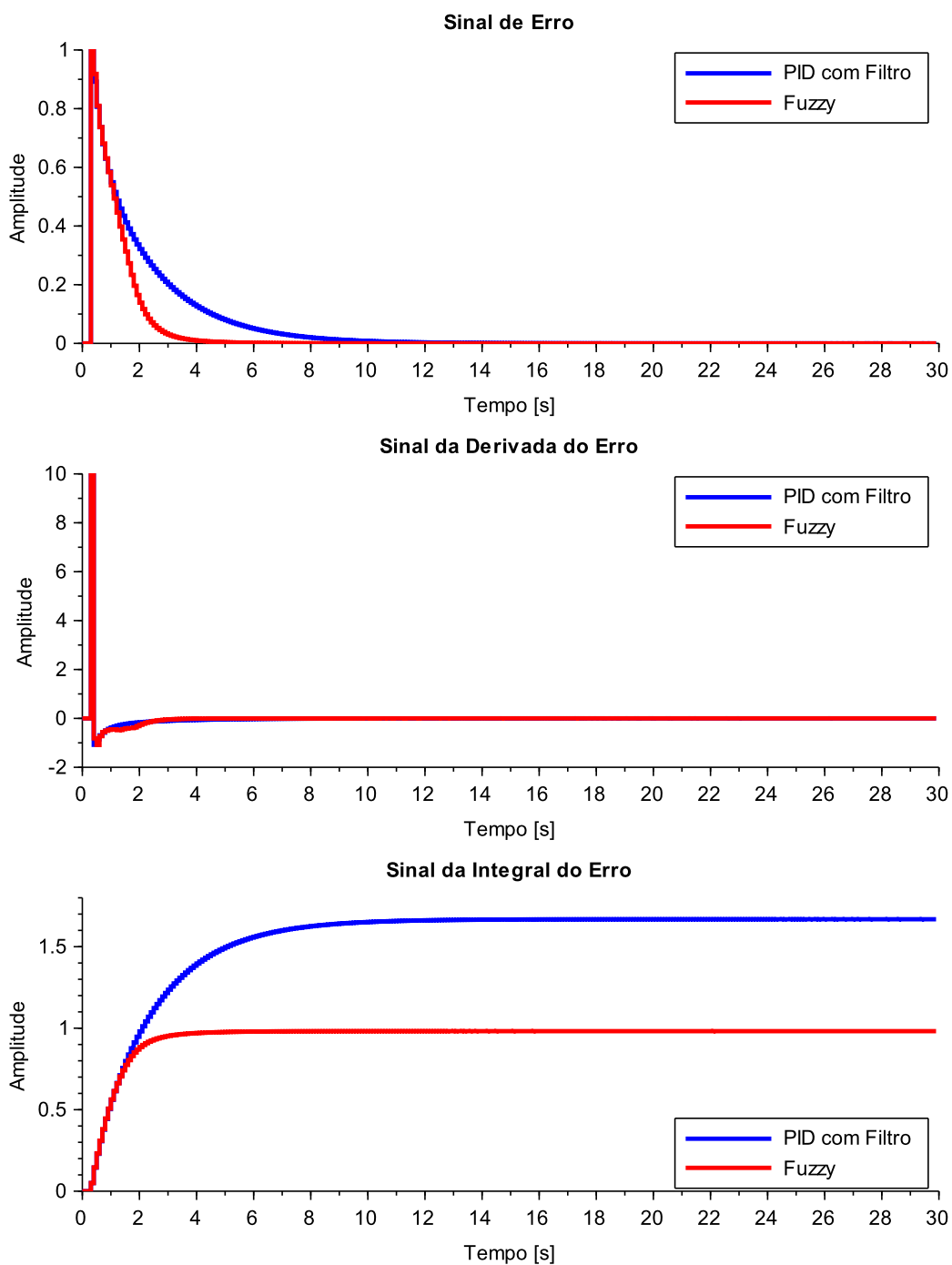


Fonte: Autor.

A resposta apresentada pelo sistema controlado pelo *Fuzzy-PID* mostra um desempenho melhor em relação ao PID. Entretanto, o acumulado do sinal de controle foi maior que o do PID.

Os sinais do erro, da derivada do erro e da integral do erro são mostrados na Figura 26.

Figura 26 – Sinais de Erro do 2º Sistema.



Fonte: Autor

Outro resultado da simulação são os índices de desempenho, como mostra a Tabela 10. Os valores destacados foram os que apresentaram melhor performance.

Método de Controle	IAE	ISE	ISU	J	ITAE	ITSE
$PID_{filtrado}$	16,683	7,247	28110,25	28117,5	37,964	8,297
$Fuzzy - PID$	9,818	5,625	29204,81	29210,43	10,586	4,062

Tabela 10 – Comparação dos índices de desempenho entre os controladores PID com Filtro e $Fuzzy-PID$ para o 3º Caso.

Os índices IAE, ISE, ITAE e ITSE do controlador $Fuzzy-PID$ apresentaram, respectivamente, o valor de 69,93%, 28,83%, 258,627%, 104,268% dos índices do PID. Ao mesmo tempo, para conseguir imprimir esses resultados, foi necessário um esforço 3,748% maior no sinal de controle, indicado pelo índice ISU do $Fuzzy-PID$ e um gasto total de energia 3,742% maior exposto por J.

Já para as especificações de resposta transitória, temos os resultados mostrados na Tabela 11. Os valores destacados foram os que apresentaram melhor desempenho.

Índice	PID c/ Filtro	$Fuzzy-PID$
Tempo de Subida t_r	4,1 s	1,7 s
Tempo de Acomodação t_s	8,0 s	3,4 s
Variância de $y(t)$	0,029	0,027
Variância de $u(t)$	2,124	1,333
Sobressinal M_p	Não	Não
Erro de Regime Permanente	Não	Não

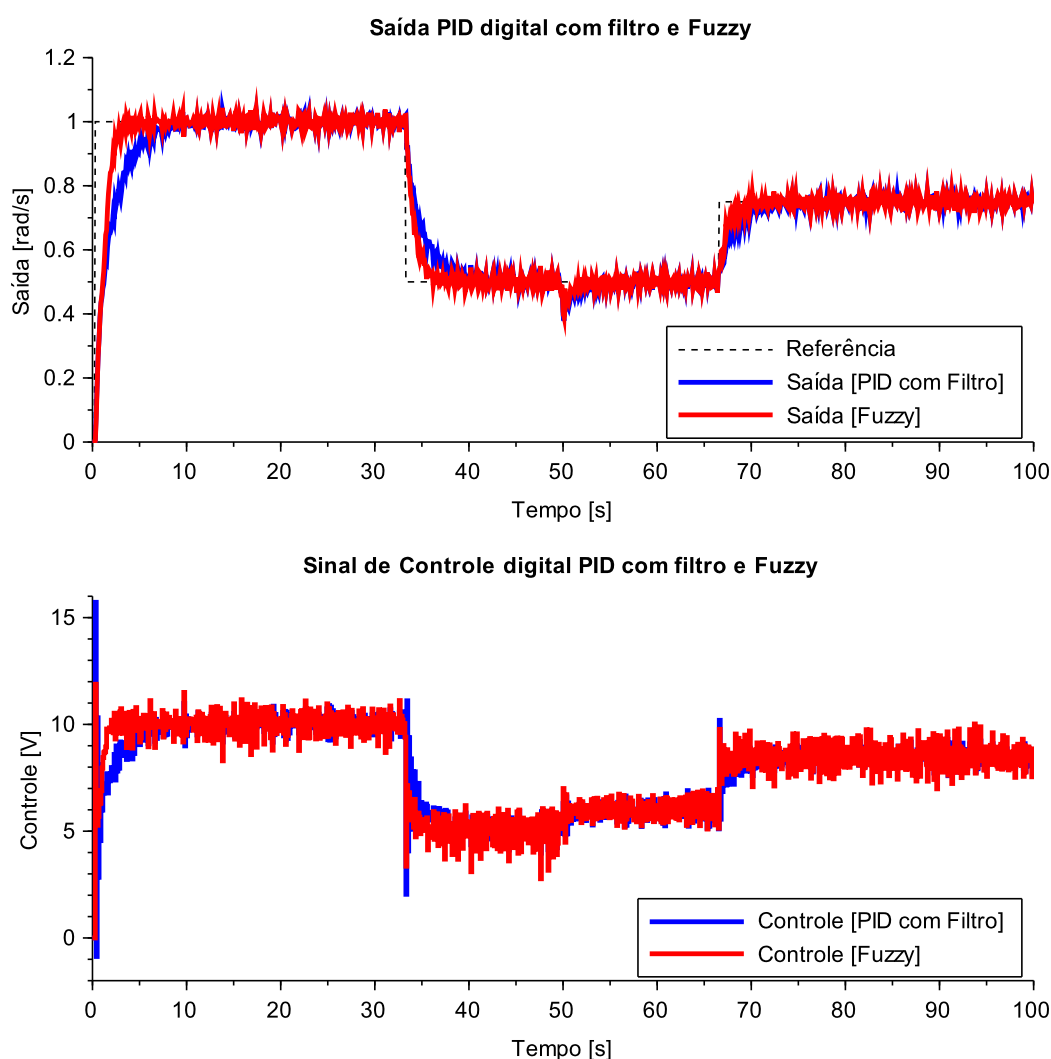
Tabela 11 – Comparação das especificações de resposta transitória dos controladores para o 3º Caso.

Os resultados do 3º caso simulado mostram que a dinâmica apresentada no 1º caso se repetiu, com o controlador $Fuzzy-PID$ apresentando uma resposta mais rápida, exposta pelos índices tempo de subida e de acomodação. Seu sinal de saída e o de controle apresentaram uma menor variação comparados ao PID, cerca de 10,86% menor para o sinal de saída e 59,34% menor para o sinal de controle.

4.4 4º Caso: Resposta a série de degraus do 2º Sistema com ruídos.

As variáveis de entrada e saída do controlador *Fuzzy-PID* e a superfície de controle para essa simulação são as mesmas mostradas na Figura 22, Figura 23 e na Figura 24. A Figura 27 mostra a resposta ao degrau unitário do sistema controlado pelo PID e o sistema controlado pelo controlador *Fuzzy-PID* e o sinal de controle dos mesmos.

Figura 27 – Resposta a série de degraus do 2º Sistema com ruídos.



Fonte: Autor.

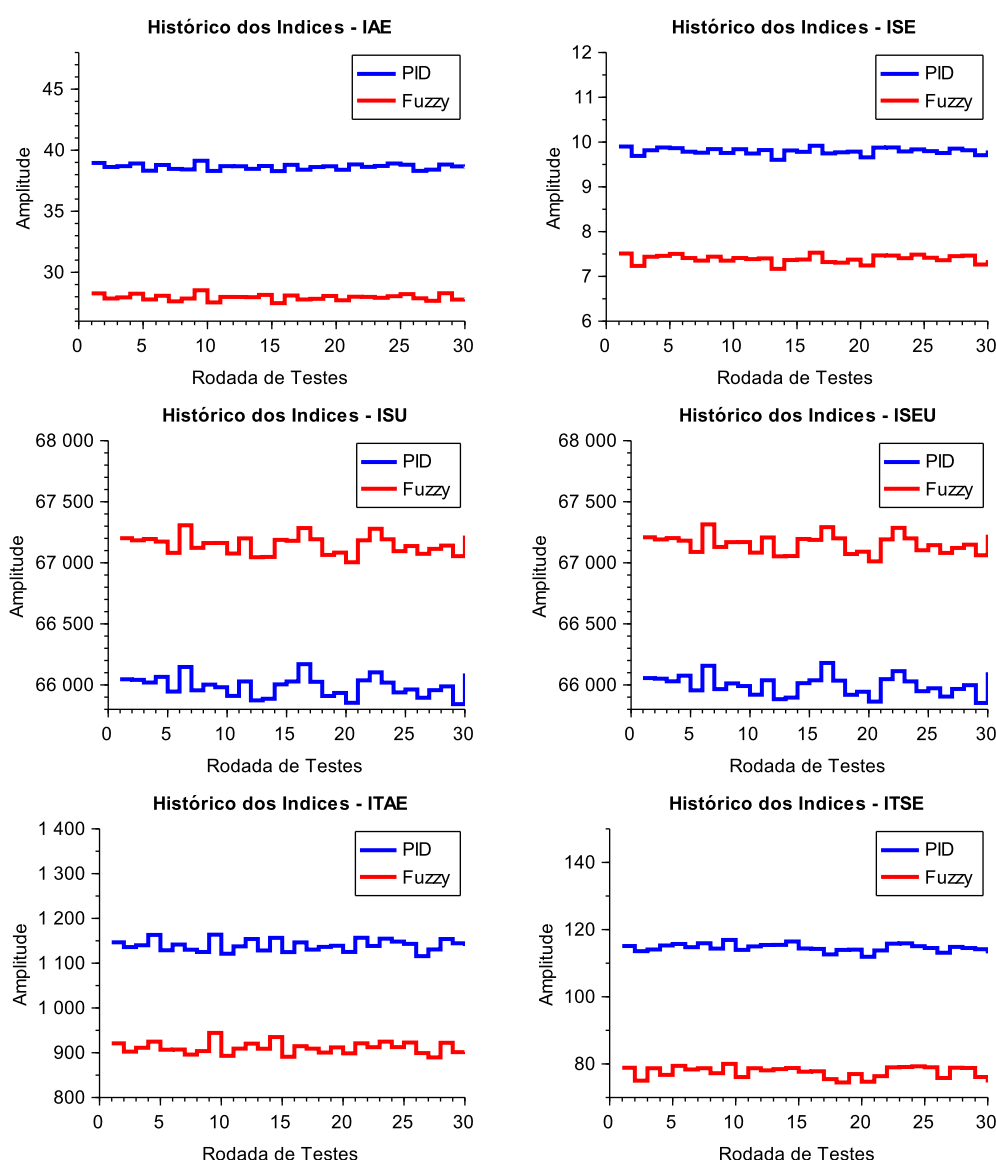
A resposta apresentada pelo sistema controlado pelo *Fuzzy-PID* mostra um desempenho melhor em relação ao PID. O sinal de controle do *Fuzzy-PID* foi maior que o do PID. A média das variâncias dos sinais de saída foram 0,041 e 0,0436 para o PID e o *Fuzzy-PID*, respectivamente, para o sinal de controle foram de 3,745 e 3,993. Como no caso o sistema foi simulado com a adição de um ruído estocástico, foi feita uma média para definir os valores dos índices na Tabela

12; o histórico desses índices é mostrado na Figura 28.

Método de Controle	IAE	ISE	ISU	J	ITAE	ITSE
$PID_{filtrado}$	38,64	9,747	65988,89	65998,69	1139,953	114,591
$Fuzzy - PID$	27,945	7,391	67148,38	67155,78	910,599	77,572

Tabela 12 – Comparação da média dos índices de desempenho entre os controladores PID com Filtro e $Fuzzy-PID$ para o 4º Caso.

Figura 28 – Variação dos índices de desempenho do 4º Caso.



Fonte: Autor.

Os índices IAE, ISE, ITAE e ITSE do controlador $Fuzzy-PID$ apresentaram, respectivamente, o valor médio de 38,268%, 32,548%, 25,187%, 47,722% dos índices do PID. Ao mesmo tempo, para conseguir imprimir esses resultados, foi necessário um esforço médio 1,727% maior

no sinal de controle, indicado pelo índice ISU do *Fuzzy-PID* e um gasto total de energia 1,723% maior, em média, exposto por J.

Os resultados do 4º caso simulado mostraram que com o controlador *Fuzzy-PID* o sistema apresentou uma resposta mais rápida, seu sinal de saída e o de controle apresentaram uma maior variação comparados ao PID, cerca de 6,37% maior para o sinal de saída e 6,6% menor para o sinal de controle.

5 CONCLUSÃO

Este trabalho comparou o desempenho de dois tipos de controladores: o PID (Proporcional-Integral-Derivativo) e o *Fuzzy-PID*. A análise dos resultados obtidos a partir das simulações no software Scilab demonstrou que ambos os controladores apresentam vantagens distintas, dependendo das condições e características do sistema em questão.

O controlador PID, com sua abordagem tradicional e robustez, mostrou um bom desempenho geral nos sistemas, sendo capaz de minimizar oscilações no sinal de controle. No entanto, ele apresenta limitações quando se trata de lidar com incertezas e perturbações. A simulação do PID com filtro também indicou um desempenho consistente, embora com tempos de resposta relativamente mais altos quando comparado ao controlador *Fuzzy-PID*, que demonstrou uma capacidade superior de adaptação a distúrbios.

Por outro lado, o controlador *Fuzzy-PID* destacou-se em cenários que envolvem incertezas e perturbações no processo, mostrando uma resposta mais rápida e eficiente. A redução significativa nos índices de erro, como o ITSE e IAE, e a adaptação às variações do sistema confirmaram as vantagens do controlador *Fuzzy-PID* nos casos simulados. Embora o esforço no sinal de controle e o gasto energético para imprimir essa dinâmica no sistema tenham sido ligeiramente maiores, o desempenho global em termos de velocidade da resposta transitória foi superior ao do PID, inclusive em sistemas com ruído ou variações inesperadas. Esses resultados sugerem que o controlador *Fuzzy-PID* tem grande potencial de aplicação em áreas como automação adaptativa e robótica, em que, além da precisão e rapidez, a maleabilidade e adaptabilidade do sistema a variabilidades externas são altamente necessárias.

Este trabalho reforça que a escolha entre PID e *Fuzzy-PID* depende do contexto operacional. Enquanto o PID é uma solução consolidada para sistemas bem modelados, o *Fuzzy-PID* oferece flexibilidade em ambientes incertos. A dualidade entre precisão e adaptabilidade reflete um princípio maior da engenharia: não há "melhor controle", mas o controle mais adequado ao caso.

Como perspectivas futuras, espera-se aplicar a combinação de controladores PID e *Fuzzy-PID*, como no caso de controladores híbridos, podendo oferecer o melhor dos dois mundos, unindo a precisão do PID com a adaptabilidade do *Fuzzy-PID*. Além disso, a validação experimental em plantas reais e a exploração de integração com técnicas de inteligência artificial, como o uso de redes neurais para a otimização das funções de pertinência, poderiam ampliar ainda mais o alcance e a eficácia dessas abordagens em sistemas de controle avançados.

Referências

ADÃO, A. A. *Controle PID e preditivo: aspectos teóricos e de implantação prática*. 2011. 85 p. Dissertação (Mestrado em Engenharia Elétrica) — Centro de Tecnologia-Universidade Federal do Ceará (UFC), 2011. Disponível em: <<http://repositorio.ufc.br/handle/riufc/2276>>. Citado na página 5.

ASTROM, K.; HAGGLUND, T. *Pid controllers: theory, design and tuning*. *Instrument Society of America*, 1995. Citado na página 6.

ÅSTRÖM, K.; MURRAY, R. *Feedback Systems: An Introduction for Scientists and Engineers, Second Edition*. Princeton University Press, 2021. ISBN 9780691193984. Disponível em: <<https://books.google.com.br/books?id=I50DEAAQBAJ>>. Citado 3 vezes nas páginas 18, 22 e 23.

ÅSTRÖM, K.; WITTENMARK, B. *Adaptive Control: Second Edition*. Dover Publications, 2013. (Dover Books on Electrical Engineering). ISBN 9780486319148. Disponível em: <<https://books.google.com.br/books?id=4CLCAgAAQBAJ>>. Citado 2 vezes nas páginas 1 e 5.

BATESON, R. *Introduction to Control System Technology*. C. E. Merrill Publishing Company, 1980. (Merrill's international series in electrical and electronics publications). ISBN 9780675082556. Disponível em: <<https://books.google.com.br/books?id=BjMoAQAAMAAJ>>. Citado na página 6.

CHAO, C.-T. et al. Equivalence between fuzzy pid controllers and conventional pid controllers. *Applied Sciences*, v. 7, n. 6, 2017. ISSN 2076-3417. Disponível em: <<https://www.mdpi.com/2076-3417/7/6/513>>. Citado 2 vezes nas páginas 19 e 20.

DORF, R.; BISHOP, R. *Sistemas de controle modernos*. LTC, 2009. ISBN 9788521617143. Disponível em: <<https://books.google.com.br/books?id=pT4bQAAACAAJ>>. Citado 6 vezes nas páginas 1, 4, 5, 7, 22 e 23.

ENTERPRISES, S. *Scilab: Free and Open Source Software for Computational Mathematics*. 2024. Acesso em: 10 fev. 2025. Disponível em: <<https://www.scilab.org>>. Citado na página 24.

GOGUEN, J. A. L. a. zadeh. fuzzy sets. information and control, vol. 8 (1965), pp. 338–353. - l. a. zadeh. similarity relations and fuzzy orderings. information sciences, vol. 3 (1971), pp. 177–200. *Journal of Symbolic Logic*, v. 38, n. 4, p. 656–657, 1973. Citado na página 1.

GOMIDE, R. R. G. F. A. C. Modelagem, controle, sistemas e lógica fuzzy. *SBA Controle & Automação*, v. 4, n. 3, p. 97–115, 1994. Disponível em: <<https://www.sba.org.br/revista/volumes/v4n3/v4n3a02.pdf>>. Citado 3 vezes nas páginas 5, 9 e 11.

HALIM, H.; ISMAIL, I. Tree physiology optimization on siso and mimo pid control tuning. *Neural Computing and Applications*, v. 31, 11 2019. Citado na página 6.

HYUN, J.; PARK, C. Compromise: An alternative solution strategy for contradiction problems in the butterfly model. *Procedia CIRP*, v. 39, p. 103–108, 12 2016. Citado na página 4.

OGATA, K. *Engenharia de Controle Moderno*. Pearson Universidades, 2010. ISBN 9788576058106. Disponível em: <<https://books.google.com.br/books?id=iL3FYgEACAAJ>>. Citado 6 vezes nas páginas 1, 5, 6, 22, 23 e 24.

PASQUINI, N. C. As revoluções industriais: uma abordagem conceitual. *Revista Tecnológica da Fatec Americana*, v. 8, p. 29–44, 3 2014. Citado na página 4.

PASSINO, K.; YURKOVICH, S. *Fuzzy Control*. Addison-Wesley, 1998. ISBN 9780201180749. Disponível em: <<https://books.google.com.br/books?id=7eNSAAAAMAAJ>>. Citado 2 vezes nas páginas 2 e 9.

PEÑAS, M. S. *Contribución a los métodos de sintonía de los controladores basados en lógica borrosa*. Tese (Tesis Doctoral) — Facultad de Ciencias Físicas-Universidad Complutense de Madrid, 2002. Disponível em: <<https://hdl.handle.net/20.500.14352/62839>>. Citado na página 5.

PERRONE, G. C. Máquina a vapor — um modelo de usina térmica. *Acervo Museológico dos Laboratórios de Ensino de Física*, 3 2020. Disponível em: <<https://www.ufrgs.br/amlef/2020/03/01/maquina-a-vapor/>>. Citado na página 4.

RAHMAN, Z.-A. S. A. Design a fuzzy logic controller for controlling position of d.c. motor. *International Journal of Computer Engineering In Research Trends*, 07 2017. Citado 2 vezes nas páginas 13 e 14.

SAATCHI, R. Fuzzy logic concepts, developments and implementation. *Information*, v. 15, n. 10, 2024. ISSN 2078-2489. Disponível em: <<https://www.mdpi.com/2078-2489/15/10/656>>. Citado na página 7.

SRIVASTAVA, S.; PANDIT, V. *A NEW SCHEME FOR DIRECT ESTIMATION OF PID CONTROLLER PARAMETERS*. 2012. Citado 2 vezes nas páginas 15 e 27.

WANG, L. *A Course in Fuzzy Systems and Control*. Prentice Hall PTR, 1997. ISBN 9780135408827. Disponível em: <<https://books.google.com.br/books?id=wbJQAAAAMAAJ>>. Citado 2 vezes nas páginas 1 e 5.

ZADEH, L. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338–353, 1965. ISSN 0019-9958. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S001999586590241X>>. Citado 2 vezes nas páginas 5 e 7.

ZHOU, K.; DOYLE, J. *Essentials of Robust Control*. Prentice Hall, 1998. (Prentice Hall Modular Series for Eng). ISBN 9780135258330. Disponível em: <<https://books.google.com.br/books?id=QviHQgAACAAJ>>. Citado na página 1.

A Códigos da Simulação

A.1 Código Principal

Aqui está o código principal utilizado na simulação.

```

clear; xdel(winsid()); clc;
s=%s; // s de Laplace
z=%z; // z da Transf. z
//Carregamento da função de criar sistema fuzzy
//Colocar no diretório atual a localização exata das funções .sci
exec("CriarFuzzy.sci");
exec("SimularSistema.sci");
exec("Perturbacoes.sci");
exec("controlePID.sci");

//Casos para Simulação:
//Caso 1:Resposta da Planta 1 ao Degrau unitario(sem restrições e sem
ruidos)
//Caso 2:Resposta da Planta 1 a Serie de Degraus com ruidos e restrições.
//Caso 3:Resposta da Planta 2 ao Degrau unitario(sem restrições e sem
ruído)
//Caso 4:Resposta da Planta 2 a Serie de Degraus com ruidos.

CASO=1;//Selecione o Caso a Simular
select CASO
case 1 then
// Sistema tempo Continuo
Gs = syslin('c', (4.4)/(s^2+2*s+4));
// Setup da simulação
Ts = 0.1;
tfinal = 30; // tempo total em segundos
N = round(tfinal/Ts); // Total de amostras
t=0:Ts:N*Ts-Ts; // vetor de tempo para os plots
// Sistema tempo Discreto
Gz = ss2tf(dscr(Gs, Ts));
Az = coeff(Gz.den); Az = flipdim(Az, 2);
Bz = coeff(Gz.num); Bz = flipdim(Bz, 2);
kn=length(Az);
// Sintonia PID
kp =0.1; ki = 0.5; kd = 0.05; //tirado do ex silveira
wc=10/Ts;// Frequencia de corte filtro derivativo
M=5;// Numero de funções de pertinencia para cada entrada
// Sinal de referência da malha de controle
r(1:3)=0;r(4:N)=1; // degrau unitário

```

```

    Umax=35*max(abs(r)); // Valor máximo do sinal de saída do controlador.
//Pertuba es #1=ligado #0=desligado
    //[Pert de carga;Pert de ruído; Erro de modelo]
    pert=[0; 0; 0];
//Satura o do sinal de controle #1=ligado #0=desligado
    Sat=0;
    rep=30;

case 2 then
// Sistema tempo Continuo
    Gs = syslin('c', (4.4)/(s^2+2*s+4));
// Setup da simulação
    Ts = 0.1;
    tfinal = 45; // tempo total em segundos
    N = round(tfinal/Ts); // Total de amostras
    t=0:Ts:N*Ts-Ts; // vetor de tempo para os plots
// Sistema tempo Discreto
    Gz = ss2tf(dscr(Gs,Ts));
    Az = coeff(Gz.den); Az = flipdim(Az,2);
    Bz = coeff(Gz.num); Bz = flipdim(Bz,2);
    kn=length(Az);
// Sintonia PID
    kp =0.1; ki = 0.5; kd = 0.05; //tirado do ex silveira
    wc=10/Ts; // Frequencia de corte filtro derivativo
    M=5; // Numero de fun es de pertinencia para cada entrada
// Sinal de referencia da malha de controle
    r(1:3)=0;r(4:N/3)=1;r(N/3+1:2*N/3)=0.5;r(2*N/3+1:N+1)=0.75; //Sinal
    Variante
    Umax=35*max(abs(r)); // Valor máximo do sinal de saída do controlador.
//Pertuba es
    //[Pert de carga;Pert de ruído; Erro de modelo]
    // #1=ligado #0=desligado
    pert=[1; 1; 0];
//Satura o do sinal de controle
    //// #1=ligado #0=desligado
    Sat=1;
    rep=1;

case 3 then
// Sistema tempo Continuo
    Ra=1; // Resistência da armadura (Ohm)
    La=0.5; // Indutância da armadura (H)
    J=0.01; // Momento de inércia (kg.m^2)
    B=0.1; // Atrito viscoso (N.m.s)
    Ke=0.01; // Constante de força contraeletromotriz (V.s/rad)
    Kt=0.01; // Constante de torque (N.m/A)

```

```

Gs = syslin('c', (Kt)/((Ra+ La*s)*(B+ J*s) +Ke*Kt));

// Setup da simulação
Ts = 0.1;
tfinal = 30; // tempo total em segundos
N = round(tfinal/Ts); // Total de amostras
t=0:Ts:N*Ts-Ts; // vetor de tempo para os plots

// Sistema tempo Discreto
Gz = ss2tf(dscr(Gs,Ts));
Az = coeff(Gz.den); Az = flipdim(Az,2);
Bz = coeff(Gz.num); Bz = flipdim(Bz,2);
kn=length(Az);
// Sintonia PID
kp =6; ki = 6/0.6; kd = 6*0.0926;
wc=10/Ts;// Freqüencia de corte filtro derivativo
M=5;

// Sinal de referência da malha de controle
r(1:2)=0; r(3:N)=1; // degrau unitário
Umax=35*max(abs(r)); // Valor máximo do sinal de saída do controlador.
//Perturbações
//[Pert de carga;Pert de ruído; Erro de modelo]
//#1=ligado #0=desligado;
pert=[0; 0; 0];

//Saturação do sinal de controle
////#1=ligado #0=desligado
Sat=0;
rep=30;

case 4 then
// Sistema tempo Continuo
Ra=1; // Resistência da armadura (Ohm)
La=0.5; // Indutância da armadura (H)
J=0.01; // Momento de inércia (kg.m^2)
B=0.1; // Atrito viscoso (N.m.s)
Ke=0.01; // Constante de força contraeletromotriz (V.s/rad)
Kt=0.01; // Constante de torque (N.m/A)

Gs = syslin('c', (Kt)/((Ra+ La*s)*(B+ J*s) +Ke*Kt));

// Setup da simulação
Ts = 0.1;
tfinal = 100; // tempo total em segundos
N = round(tfinal/Ts); // Total de amostras
t=0:Ts:N*Ts-Ts; // vetor de tempo para os plots

```

```

// Sistema tempo Discreto
Gz = ss2tf(dscr(Gs,Ts));
Az = coeff(Gz.den); Az = flipdim(Az,2);
Bz = coeff(Gz.num); Bz = flipdim(Bz,2);
kn=length(Az);
// Sintonia PID
kp =6; ki = 6/0.6; kd = 6*0.0926;
wc=10/Ts;// Frecuencia de corte filtro derivativo
M=5;

// Sinal de referencia da malha de controle
r(1:3)=0;r(4:N/3)=1;r(N/3+1:2*N/3)=0.5;r(2*N/3+1:N+1)=0.75;//Sinal
    Variante
    Umax=35*max(abs(r));// Valor máximo do sinal de saída do controlador.
//Pertuba es
    //[Pert de carga;Pert de ruido; Erro de modelo]
    // #1=ligado #0=desligado;
    pert=[1; 1; 0];

//Satura o do sinal de controle
    //// #1=ligado #0=desligado
    Sat=0;
    rep=1;

end

// Tustin
S0 = kp +ki*Ts/2 +2*kd/Ts;
S1 = ki*Ts -4*kd/Ts;
S2 = -kp +ki*Ts/2 +2*kd/Ts;

// Tustin com filtro derivativo
q0=1 +wc*Ts/2;;
q1=-2;
q2=1 -wc*Ts/2;
Q=[q0;q1;q2];

s0= kp +kp*wc*Ts/2 +ki*Ts/2 +ki*wc*(Ts^2)/4 +kd*wc;
s1= -2*kp +ki*Ts -2*kd*wc;
s2= kp -kp*wc*Ts/2 -ki*Ts/2 +ki*wc*(Ts^2)/4 +kd*wc;
S=[s0;s1;s2];

MISE=0;MISEF=0;
MISU=0;MISUF=0;

```

```

MISEU=0;MISEUF=0;
MIAE=0;MIAEF=0;
MITAE=0;MITAEF=0;
MITSE=0;MITSEF=0;

for rep=1:30

// Condições iniciais
ISE=0;ISEF=0;// Condições para o funcionamento do ISE
ISU=0;ISUF=0;// Condições para o funcionamento do ISU
IAE=0;IAEF=0; // Condições para o funcionamento do IAE
ITAE=0;ITAEF=0; // Condições para o funcionamento do ITAE
ITSE=0;ITSEF=0; // Condições para o funcionamento do ITSE

Sys=zeros(1:kn);Cont=zeros(1:kn);
E=zeros(1:kn);DE=zeros(1:kn);IE=zeros(1:kn);

SysF=zeros(1:kn); ContF=zeros(1:kn);
EF=zeros(1:kn);DEF=zeros(1:kn);IEF=zeros(1:kn);

Fuzzy=criarfuzzy(M,Umax,kp,ki,kd);

// Definindo os Sinais de Perturbação
var_ruido_sensor = 0.0002;
v= grand(N, 1, "nor", 0, sqrt(var_ruido_sensor));
po(1:round(N/2))= 0;po(round(N/2)+1 : N)= -0.1*max(abs(r));
w= grand(N, 1, "nor", 0, sqrt(var_ruido_sensor));

for k=kn+1:N

Sys=[Sys (Sistema(Az,Bz,Cont, Sys,k)+Perturbacoes(Az,po,v,w,k)*pert)];
E=[E r(k)-Sys(k)];
IE=[IE (IE(k-1)+(Ts/2)*[E(k)+E(k-1)])];
DE=[DE ((1/Ts)*(E(k)-E(k-1)))];
Cont=[Cont controlePID(Q,S,Cont,E,k)];

SysF=[SysF (Sistema(Az,Bz,ContF, SysF,k)+Perturbacoes(Az,po,v,w,k)*pert)];
EF=[EF r(k)-SysF(k)];
IEF=[IEF IEF(k-1)+(Ts/2)*[EF(k)+EF(k-1)]];
DEF=[DEF ((1/Ts)*(EF(k)-EF(k-1)))];
ContF=[ContF evalfls([EF(k) IEF(k) DEF(k)], Fuzzy)];

/////Limitação do controlador 0->5
////////////////////////////////////
if Sat==1 then
    if Cont(k)>5 then
        Cont(k)=5;

```

```

elseif Cont(k)<0 then
    Cont(k)=0
end
if ContF(k)>5 then
    ContF(k)=5;
elseif ContF(k)<0 then
    ContF(k)=0
end
end
////////////////////////////////////
// Calculo dos ndices de Desempenho do PID c/ Filtro
ISE= ISE + E(k)^2; // ISE a taxa de acmulo de erro
ISU= ISU + Cont(k)^2; // ISU o quanto de energia gasta no
    processo
ISEU= ISE + ISU;
IAE= IAE + abs(E(k));
ITAE= ITAE + t(k)*abs(E(k));
ITSE= ITSE + t(k)*E(k)^2;

// Calculo dos ndices de Desempenho do PID c/ Fuzzy
ISEF= ISEF + EF(k)^2; // ISE a taxa de acmulo de erro
ISUF= ISUF + ContF(k)^2; // ISU o quanto de energia gasta no
    processo
ISEUF= ISEF + ISUF; //Acumulado das duas taxas combinadas
IAEF= IAEF + abs(EF(k));
ITAEF= ITAEF + t(k)*abs(EF(k));
ITSEF= ITSEF + t(k)*EF(k)^2;

end
MIAE=[MIAE IAE];MIAEF=[MIAEF IAEF];
MISE=[MISE ISE];MISEF=[MISEF ISEF];
MISU=[MISU ISU];MISUF=[MISUF ISUF];
MISEU=[MISEU ISEU];MISEUF=[MISEUF ISEUF];
MITAE=[MITAE ITAE];MITAEF=[MITAEF ITAEF];
MITSE=[MITSE ITSE];MITSEF=[MITSEF ITSEF];

end

MIAE(:,1)=[];MIAEF(:,1)=[];
MISE(:,1)=[];MISEF(:,1)=[];
MISU(:,1)=[];MISUF(:,1)=[];
MISEU(:,1)=[];MISEUF(:,1)=[];
MITAE(:,1)=[];MITAEF(:,1)=[];
MITSE(:,1)=[];MITSEF(:,1)=[];

// Resultados da simula o

```

```

scf(1);
plotvar(Fuzzy, "input", [1 2 3]); //Conjuntos de entrada Fuzzy
scf(2);
plotvar(Fuzzy, "output", 1); //Conjuntos de saída Fuzzy
if M==5 then
PlotInput=scf(1);
for i=1:3
PlotInput.children(i).font_size=2;
PlotInput.children(i).children(1).text=["N"; "NM"; "Z"; "PM"; "P"];
PlotInput.children(i).x_label.font_style = 6; PlotInput.children(i).x_label.
font_size = 3;
PlotInput.children(i).y_label.font_style = 6; PlotInput.children(i).y_label.
font_size = 3;
end
for i=1:5
PlotInput.children(1).children(2).children(i).thickness=3;
PlotInput.children(2).children(2).children(i).thickness=3;
PlotInput.children(3).children(2).children(i).thickness=3;
end
PlotOutput=scf(2);
PlotInput.children(1).font_size=2;
PlotOutput.children(1).children(1).text=["N"; "NL"; "NML"; "NM"; "NMS"; "NS"; "Z"
; "PS"; "PMS"; "PM"; "PML"; "PL"; "P"];
PlotOutput.children(1).x_label.font_style = 6; PlotOutput.children(1).
x_label.font_size = 3;
PlotOutput.children(1).y_label.font_style = 6; PlotOutput.children(1).
y_label.font_size = 3;

fixData=PlotOutput.children(1).children(2).children(1).data;
fixData(length(fixData(:,1))-1,2)=1;
PlotOutput.children(1).children(2).children(1).data=fixData;
for i=1:13
PlotOutput.children(1).children(2).children(i).polyline_style=5;
PlotOutput.children(1).children(2).children(i).thickness=3;
end
end
scf(3);
PlotSurf=gcf();
PlotSurf.figure_size=[800,800];
PlotSurf.figure_name="Superfícies de Controle Fuzzy";
subplot(2,2,1); plotsurf(Fuzzy, [1 2], 1, [0 0 0], 21, 2); //Superfícies Sa da
Fuzzy
PlotSurf=get("current_axes"); PlotSurf.font_size=3;
PlotSurf.title.text="Superfície do Erro com Integral do Erro";
PlotSurf.title.font_style = 8; PlotSurf.title.font_size = 3;
PlotSurf.x_label.font_style = 6; PlotSurf.x_label.font_size = 3;

```

```

PlotSurf.y_label.font_style = 6;PlotSurf.y_label.font_size = 3;
PlotSurf.z_label.text="Sa da";
PlotSurf.z_label.font_style = 6;PlotSurf.z_label.font_size = 3;

subplot(2,2,2);plotsurf(Fuzzy,[1 3],1,[0 0 0],21,2);//Superfícies Sa da
    Fuzzy
PlotSurf=get("current_axes");PlotSurf.font_size=3;
PlotSurf.title.text="Superfície_do_Erro_com_Derivada_do_Erro";
PlotSurf.title.font_style = 8;PlotSurf.title.font_size = 3;
PlotSurf.x_label.font_style = 6;PlotSurf.x_label.font_size = 3;
PlotSurf.y_label.font_style = 6;PlotSurf.y_label.font_size = 3;
PlotSurf.z_label.text="Sa da";
PlotSurf.z_label.font_style = 6;PlotSurf.z_label.font_size = 3;

subplot(2,2,3);plotsurf(Fuzzy,[2 3],1,[0 0 0],21,2);//Superfícies Sa da
    Fuzzy
PlotSurf=get("current_axes");PlotSurf.font_size=3;
PlotSurf.title.text="Superfície_da_Integral_do_Erro_com_a_Derivada_do_Erro
    ";
PlotSurf.title.font_style = 8;PlotSurf.title.font_size = 3;
PlotSurf.x_label.font_style = 6;PlotSurf.x_label.font_size = 3;
PlotSurf.y_label.font_style = 6;PlotSurf.y_label.font_size = 3;
PlotSurf.z_label.text="Sa da";
PlotSurf.z_label.font_style = 6;PlotSurf.z_label.font_size = 3;

// Resultados da simulação
scf(6);
subplot(2,1,1);
Plot1=gcf();
Plot1.figure_size=[800,800];
Plot1.figure_name="Resposta_do_Sistema";
Plot1=get("current_axes");Plot1.font_size=3;
Plot1.title.text="Sa da_PID_digital_com_filtro_e_Fuzzy";
Plot1.x_label.text="Tempo_[s]";Plot1.y_label.text="Sa da_[V]";if CASO==3|
    CASO==4 then Plot1.y_label.text="Sa da_[rad/s]";end
Plot1.title.font_style = 8;Plot1.title.font_size = 3;
Plot1.x_label.font_style = 6;Plot1.x_label.font_size = 3;
Plot1.y_label.font_style = 6;Plot1.y_label.font_size = 3;

plot2d(t,[r(1:length(t)),Sys(1:length(t))',SysF(1:length(t))'],[1,2,5]);
legend('Referência','Sa da_[PID_com_Filtro]','Sa da_[Fuzzy]',4);
set(Plot1.children.children(1),'thickness',3);
set(Plot1.children.children(2),'thickness',3);
set(Plot1.children.children(3),'line_style',3);

subplot(2,1,2);

```

```

Plot2=gcf();
Plot2=get("current_axes");Plot2.font_size=3;
Plot2.title.text="Sinal_de_Controlo_digital_PID_com_filtro_e_Fuzzy";
Plot2.x_label.text="Tempo_[s]";Plot2.y_label.text="Controlo_[V]";
Plot2.title.font_style = 8;Plot2.title.font_size = 3;
Plot2.x_label.font_style = 6;Plot2.x_label.font_size = 3;
Plot2.y_label.font_style = 6;Plot2.y_label.font_size = 3;
plot2d2(t, [Cont', ContF'], [2, 5]);
legend('Controlo_[PID_com_Filtro]', 'Controlo_[Fuzzy]', 4);
set(Plot2.children.children(1), 'thickness', 3);
set(Plot2.children.children(2), 'thickness', 3);

scf(7);
subplot(3,1,1);
Plot3=gcf();
Plot3.figure_size=[800,1000];
Plot3.figure_name="Sinais_dos_Controladores";
Plot3=get("current_axes");Plot3.font_size=3;
Plot3.title.text="Sinal_de_Erro";
Plot3.x_label.text="Tempo_[s]";Plot3.y_label.text="Amplitude";
Plot3.title.font_style = 8;Plot3.title.font_size = 3;
Plot3.x_label.font_style = 6;Plot3.x_label.font_size = 3;
Plot3.y_label.font_style = 6;Plot3.y_label.font_size = 3;
plot2d2(t, [E', EF'], [2, 5]);
legend('PID_com_Filtro', 'Fuzzy', 1);
set(Plot3.children.children(1), 'thickness', 3);
set(Plot3.children.children(2), 'thickness', 3);

subplot(3,1,2);
Plot3=get("current_axes");Plot3.font_size=3;
Plot3.title.text="Sinal_da_Derivada_do_Erro";
Plot3.x_label.text="Tempo_[s]";Plot3.y_label.text="Amplitude";
Plot3.title.font_style = 8;Plot3.title.font_size = 3;
Plot3.x_label.font_style = 6;Plot3.x_label.font_size = 3;
Plot3.y_label.font_style = 6;Plot3.y_label.font_size = 3;
plot2d2(t, [DE', DEF'], [2, 5]);
legend('PID_com_Filtro', 'Fuzzy', 1);
set(Plot3.children.children(1), 'thickness', 3);
set(Plot3.children.children(2), 'thickness', 3);

subplot(3,1,3);
Plot3=get("current_axes");Plot3.font_size=3;
Plot3.title.text="Sinal_da_Integral_do_Erro";
Plot3.x_label.text="Tempo_[s]";Plot3.y_label.text="Amplitude";
Plot3.title.font_style = 8;Plot3.title.font_size = 3;
Plot3.x_label.font_style = 6;Plot3.x_label.font_size = 3;

```

```

Plot3.y_label.font_style = 6;Plot3.y_label.font_size = 3;
plot2d2(t, [IE', IEF'], [2,5]);
legend('PID_com_Filtro', 'Fuzzy', 4);
set(Plot3.children.children(1), 'thickness', 3);
set(Plot3.children.children(2), 'thickness', 3);

if CASO==2|CASO==4 then
scf(8);
subplot(3,2,1);
Plot4=gcf();
Plot4.figure_size=[800,1000];
Plot4.figure_name=" ndices ";
Plot4=get("current_axes");Plot4.font_size=3;
Plot4.title.text="Hist rico_dos_Indices_-_IAE";
Plot4.x_label.text="Rodada_de_Testes";Plot4.y_label.text="Amplitude";
Plot4.title.font_style = 8;Plot4.title.font_size = 3;
Plot4.x_label.font_style = 6;Plot4.x_label.font_size = 3;
Plot4.y_label.font_style = 6;Plot4.y_label.font_size = 3;
plot2d2(1:30, [MIAE', MIAEF'], [2,5]);
legend('PID', 'Fuzzy', 1);
set(Plot4.children.children(1), 'thickness', 3);
set(Plot4.children.children(2), 'thickness', 3);
Plot4=gca();Plot4.data_bounds=[0 min([MIAE,MIAEF])*0.95;rep max([MIAE,MIAEF
    ])*1.2];

subplot(3,2,2);
Plot4=gcf();
Plot4=get("current_axes");Plot4.font_size=3;
Plot4.title.text="Hist rico_dos_Indices_-_ISE";
Plot4.x_label.text="Rodada_de_Testes";Plot4.y_label.text="Amplitude";
Plot4.title.font_style = 8;Plot4.title.font_size = 3;
Plot4.x_label.font_style = 6;Plot4.x_label.font_size = 3;
Plot4.y_label.font_style = 6;Plot4.y_label.font_size = 3;
plot2d2(1:30, [MISE', MISEF'], [2,5]);
legend('PID', 'Fuzzy', 1);
set(Plot4.children.children(1), 'thickness', 3);
set(Plot4.children.children(2), 'thickness', 3);
Plot4=gca();Plot4.data_bounds=[0 min([MISE,MISEF])*0.95;rep max([MISE,MISEF
    ])*1.2];

subplot(3,2,3);
Plot4=gcf();
Plot4=get("current_axes");Plot4.font_size=3;
Plot4.title.text="Hist rico_dos_Indices_-_ISU";
Plot4.x_label.text="Rodada_de_Testes";Plot4.y_label.text="Amplitude";
Plot4.title.font_style = 8;Plot4.title.font_size = 3;

```

```

Plot4.x_label.font_style = 6;Plot4.x_label.font_size = 3;
Plot4.y_label.font_style = 6;Plot4.y_label.font_size = 3;
plot2d2(1:30, [MISU', MISUF'], [2,5]);
legend('PID', 'Fuzzy', 1);
set(Plot4.children.children(1), 'thickness', 3);
set(Plot4.children.children(2), 'thickness', 3);
Plot4=gca();Plot4.data_bounds=[0 min([MISU,MISUF]);rep max([MISU,MISUF])
    *1.01];

subplot(3,2,4);
Plot4=gcf();
Plot4=get("current_axes");Plot4.font_size=3;
Plot4.title.text="Hist rico_dos_Indices_-_ISEU";
Plot4.x_label.text="Rodada_de_Testes";Plot4.y_label.text="Amplitude";
Plot4.title.font_style = 8;Plot4.title.font_size = 3;
Plot4.x_label.font_style = 6;Plot4.x_label.font_size = 3;
Plot4.y_label.font_style = 6;Plot4.y_label.font_size = 3;
plot2d2(1:30, [MISEU', MISEUF'], [2,5]);
legend('PID', 'Fuzzy', 1);
set(Plot4.children.children(1), 'thickness', 3);
set(Plot4.children.children(2), 'thickness', 3);
Plot4=gca();Plot4.data_bounds=[0 min([MISEU,MISEUF]);rep max([MISEU,MISEUF
    ]) *1.01];

subplot(3,2,5);
Plot4=gcf();
Plot4=get("current_axes");Plot4.font_size=3;
Plot4.title.text="Hist rico_dos_Indices_-_ITAE";
Plot4.x_label.text="Rodada_de_Testes";Plot4.y_label.text="Amplitude";
Plot4.title.font_style = 8;Plot4.title.font_size = 3;
Plot4.x_label.font_style = 6;Plot4.x_label.font_size = 3;
Plot4.y_label.font_style = 6;Plot4.y_label.font_size = 3;
plot2d2(1:30, [MITAE', MITAEF'], [2,5]);
legend('PID', 'Fuzzy', 1);
set(Plot4.children.children(1), 'thickness', 3);
set(Plot4.children.children(2), 'thickness', 3);
Plot4=gca();Plot4.data_bounds=[0 min([MITAE,MITAEF]) *0.95;rep max([MITAE,
    MITAEF]) *1.2];

subplot(3,2,6);
Plot4=gcf();
Plot4=get("current_axes");Plot4.font_size=3;
Plot4.title.text="Hist rico_dos_Indices_-_ITSE";
Plot4.x_label.text="Rodada_de_Testes";Plot4.y_label.text="Amplitude";
Plot4.title.font_style = 8;Plot4.title.font_size = 3;
Plot4.x_label.font_style = 6;Plot4.x_label.font_size = 3;
Plot4.y_label.font_style = 6;Plot4.y_label.font_size = 3;

```

```

plot2d2(1:30,[MITSE',MITSEF'],[2,5]);
legend('PID','Fuzzy',1);
set(Plot4.children.children(1),'thickness',3);
set(Plot4.children.children(2),'thickness',3);
Plot4=gca();Plot4.data_bounds=[0 min([MITSE,MITSEF])*0.95;rep max([MITSE,
MITSEF])*1.2];

end
IAE=mean(MIAE);IAEF=mean(MIAEF);
ISE=mean(MISE);ISEF=mean(MISEF);
ISU=mean(MISU);ISUF=mean(MISUF);
ISEU=mean(MISEU);ISEUF=mean(MISEUF);
ITAE=mean(MITAE);ITAEF=mean(MITAEF);
ITSE=mean(MITSE);ITSEF=mean(MITSEF);

mprintf("Indices_de_desempenho:\n");
mprintf("
_____
\n");
mprintf("|Metodo_de_Controle\t_|IAE\t_|ISE\t_|ISU\t_|J
\t_|ITAE\t_|ITSE\t_|
\n");

mprintf("|_____\t_|_____\t_|_____\t_|_____\t_|
_____\t_|_____\t_|_____\t_|
\n");
mprintf("|PID_c/_Filtro\t_|%.3f\t_|%.3f\t_|%.3f\t_|%.3f\t_|%.
.3f\t_|%.3f\t_|
\n", IAE, ISE, ISU, ISEU, ITAE, ITSE);

mprintf("|_____\t_|_____\t_|_____\t_|_____\t_|
_____\t_|_____\t_|_____\t_|
\n");
mprintf("|PID_c/_Fuzzy\t_|%.3f\t_|%.3f\t_|%.3f\t_|%.3f\t_|%.
.3f\t_|%.3f\t_|
\n", IAEF, ISEF, ISUF, ISEUF, ITAEF, ITSEF);
mprintf("|_____\t_|_____\t_|_____\t_|_____\t_|
_____\t_|_____\t_|_____\t_|
\n");

percentual=[(IAE/IAEF -1)*100,(ISE/ISEF -1)*100,(ISU/ISUF -1)*100,(ISEU/
ISEUF -1)*100,(ITAE/ITAEF -1)*100,(ITSE/ITSEF -1)*100,(variance(Cont)/
variance(ContF) -1)*100];
mprintf("\nDiferen a_Percentual:\n");
mprintf("_____\n");
mprintf("|_PID/FUZZY\t_|
\n");
mprintf("|_IAE:_.3f\t_|_ISE:_.3f\t_|_ISU:_.3f\t_|_J:_.3f
\t_|_ITAE:_.3f\t_|_ITSE:_.3f\t_|_Var_Cont:_.3f
\t_|
\n",percentual(1),percentual(2),percentual(3),percentual(4),percentual
(5),percentual(6),percentual(7));

EXPORT=["CONTROLADOR","IAE","ISE","ISU","J","ITAE","ITSE"];

```

```

"PID",sci2exp(IAE),sci2exp(ISE),sci2exp(ISU),sci2exp(ISEU),sci2exp(ITAE),
sci2exp(ITSE);
"FUZZY",sci2exp(IAEF),sci2exp(ISEF),sci2exp(ISUF),sci2exp(ISEUF),sci2exp(
ITAEF),sci2exp(ITSEF)];

//Verifica o do tempo de subida
if CASO==1 | CASO==3 then
for i = 1:length(Sys)
if Sys(i:$) >= max(r)*0.98 & Sys(i:$) <= max(r)*1.02 then
t_acomodacao = t(i);
break;
end
end
for i = 1:length(SysF)
if SysF(i:$) >= max(r)*0.98 & SysF(i:$) <= max(r)*1.02 then
t_acomodacaoF = t(i);
break;
end
end
for i=1:length(Sys)
if Sys(i)<=0.1*Sys(length(Sys))
pmenor=i;
end
if Sys(i)<=0.9*Sys(length(Sys))
pmaior=i;
end
end
temposub=(pmaior-pmenor)*Ts;

for i=1:length(SysF)
if SysF(i)<=0.1*SysF(length(SysF))
pmenor2=i;
end
if SysF(i)<=0.9*SysF(length(SysF))
pmaior2=i;
end
end
temposub2=(pmaior2-pmenor2)*Ts;
mprintf("\n_Tempo_de_Subida_PID:_%%.3f_segundos\n_Tempo_de_Subida_Fuzzy:_%%.3
f_segundos\n",temposub,temposub2);

mprintf("\n_Tempo_de_Acomoda o_PID:_%%.3f_segundos\n_Tempo_de
Acomoda o_Fuzzy:_%%.3f_segundos\n",t_acomodacao,t_acomodacaoF);
end

```

Listing A.1 – Código principal da simulação

A.2 Funções Auxiliares

As funções auxiliares utilizadas na simulação estão listadas abaixo.

A.2.1 Função de Simulação do Sistema

```
function Saida= Sistema (Az,Bz,u,y,k)
    Vy=0;Vu=u(k-1);
    for n=1:(max(size(Az))-1)
        Vy=[Vy;y(k-n)];
    end
    for n=1:(max(size(Bz))-1)
        Vu=[Vu;u(k-n-1)];
    end
    Saida=(-Az)*Vy + Bz*Vu;
endfunction
```

Listing A.2 – Função de Simulação do Sistema

A.2.2 Função de Controle PID

```
function PID = controlePID(Q, S, u, e, k)
    Vu = 0;
    Ve = e(k);
    for n = 1:2
        Vu = [Vu; u(k-n)];
        Ve = [Ve; e(k-n)];
    end
    PID = ((-Q')*Vu + S'*Ve) / (Q' * [1; 0; 0]);
endfunction
```

Listing A.3 – Função de controle PID

A.2.3 Função de Perturbações

```
function Pert = Perturbacoes (Az,po,v,w,k)
    pc=po(k);pr=v(k);
    for n=1:(max(size(Az))-1)
        pc=[pc;po(k-n)];
        pr=[pr;v(k-n)];
    end
    Pert=[Az*pc Az*pr w(k)];
endfunction
```

Listing A.4 – Função de Perturbações

A.2.4 Função de Criação do Sistema Fuzzy

```

//FUNÇÃO DE CRIADOR SIMPLES DE SISTEMA FUZZY
function Fuzzy= criarfuzzy(M, Au, kp, ki, kd)

// Criador de sistema Fuzzy
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
//Au= Ganho estatico do sistema/Ts
Ae=Au/(3*kp);Ai=Au/(3*ki);Ad=Au/(3*kd);

// Criando o sistema Fuzzy
Fuzzy=newfls();          Fuzzy.name="FuzzyPID";
Fuzzy.comment="Sistema_FUZZY_baseado_no_controlador_PID";
Fuzzy.type='ts';        Fuzzy.defuzzMethod="wtaver";
Fuzzy.TNorm='min';     Fuzzy.SNorm='max';
Fuzzy.ImpMethod='min'; Fuzzy.AggMethod='max';

//Definindo os Valores dos Intervalos das Funções de Pertinência;
for k=1:M
    fe(k)=[(2*k-M-1)/(M-1)]*Ae;
    fie(k)=[(2*k-M-1)/(M-1)]*Ai;
    fde(k)=[(2*k-M-1)/(M-1)]*Ad;
end

for k=1:(3*M-2)
    fu(k)=[(2*k-3*M+1)/(3*(M-1))]*Au;
end

//Criando as Funções de Pertinência da Entrada Erro
Fuzzy=addvar(Fuzzy,"input","erro(t)",[fe(1) fe(M)]);
Fuzzy=addmf(Fuzzy,"input",1,1,"zmf",[fe(1) fe(2)]);
for k=1:M-2
    Fuzzy=addmf(Fuzzy,"input",1,k+1,"trimf",[fe(k) fe(k+1) fe(k+2)]);
end
Fuzzy=addmf(Fuzzy,"input",1,M,"smf",[fe(M-1) fe(M)]);

//Criando as Funções de Pertinência da Entrada Integral do Erro
Fuzzy=addvar(Fuzzy,"input","ierro(t)",[fie(1) fie(M)]);
Fuzzy=addmf(Fuzzy,"input",2,1,"zmf",[fie(1) fie(2)]);
for k=1:M-2
    Fuzzy=addmf(Fuzzy,"input",2,k+1,"trimf",[fie(k) fie(k+1) fie(k+2)]);
end
Fuzzy=addmf(Fuzzy,"input",2,M,"smf",[fie(M-1) fie(M)]);

//Criando as Funções de Pertinência da Entrada Derivada do Erro
Fuzzy=addvar(Fuzzy,"input","derro(t)",[fde(1) fde(M)]);
Fuzzy=addmf(Fuzzy,"input",3,1,"zmf",[fde(1) fde(2)]);
for k=1:M-2

```

```

    Fuzzy=addmf(Fuzzy, "input", 3, k+1, "trimf", [fde(k) fde(k+1) fde(k+2)]);
end
    Fuzzy=addmf(Fuzzy, "input", 3, M, "smf", [fde(M-1) fde(M)]);

    //Criando as Funções de Pertinencia de Saída
    Fuzzy=addvar(Fuzzy, "output", "Output", [fu(1) fu(3*M-2)]);
//    Fuzzy=addmf(Fuzzy, "output", 1, 1, "trapmf", [fu(1)*110 fu(1)*100 fu(1)
    fu(2)]);
for k=1:(3*M-2)//
//    Fuzzy=addmf(Fuzzy, "output", 1, k+1, "trimf", [fu(k) fu(k+1) fu(k+2)]);
    Fuzzy=addmf(Fuzzy, "output", 1, k, "constant", [fu(k)]);
end
//    Fuzzy=addmf(Fuzzy, "output", 1, 3*M-2, "trapmf", [fu(3*M-3) fu(3*M-2)
    fu(3*M-2)*100// fu(3*M-2)*110]);

//Criando o Vetor de Regras
rules=[0 0 0 0 0 0];

for i=1:M
for j=1:M
for k=1:M
    rules=[rules; i j k i+j+k-2 1 1];
end
end
end

rules(1,:)=[];

//Adicionando as Regras ao Sistema
Fuzzy=addrule(Fuzzy, rules);

//Limpendo as variáveis não necessárias
clear kp ki kd i j k M Ae Ad Au Ai fe fu fie fde rules
////////////////////////////////////
endfunction

```

Listing A.5 – Função de Criação do Sistema Fuzzy