



UNIVERSIDADE FEDERAL DO PARÁ  
CAMPUS BELÉM  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

MATHEUS MORAIS DA SILVA  
**DM 25/2025**

**METODOLOGIA E SÍNTESE DE CONTROLADORES HÍBRIDOS  
CLÁSSICO-NEURAIS APLICADOS A SISTEMAS MIMO NÃO LINEARES: ESTUDO  
DE CASO EM HELICÓPTERO 2-DOF**

Belém/PA  
2025

MATHEUS MORAIS DA SILVA  
DM 25/2025

**METODOLOGIA E SÍNTESE DE CONTROLADORES HÍBRIDOS  
CLÁSSICO-NEURAI APLICADOS A SISTEMAS MIMO NÃO LINEARES: ESTUDO  
DE CASO EM HELICÓPTERO 2-DOF**

Dissertação submetida ao Programa de Pós-Graduação  
em Engenharia Elétrica da Universidade Federal do  
Pará para a obtenção do título de Mestre em Enge-  
nharia Elétrica.

Orientador: Prof. Dr. Antonio da Silva Silveira.

Belém/PA  
2025

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD  
Sistema de Bibliotecas da Universidade Federal do Pará  
Gerada automaticamente pelo módulo Ficat, mediante os dados fornecidos pelo(a)  
autor(a)

---

S586m Silva, Matheus Morais da.  
Metodologia e síntese de controladores híbridos clássico-  
neurais aplicados a sistemas mimo não lineares : estudo de  
caso em helicóptero 2-dof / Matheus Morais da Silva. — 2025.  
105 f. : il. color.

Orientador(a): Prof. Dr. Antonio da Silva Silveira  
Dissertação (Mestrado) - Universidade Federal do Pará,  
Instituto de Tecnologia, Programa de Pós-Graduação em  
Engenharia Elétrica, Belém, 2025.

1. Controle Híbrido. 2. Redes Neurais Artificiais. 3.  
Alocação de Polos. 4. Sistemas Multivariáveis Não  
Lineares. 5. Identificação por MQR. I. Título.

CDD 629.8312

---



UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

## **ATA DA APRESENTAÇÃO E DEFESA DE DISSERTAÇÃO DE MESTRADO**

ATA DA 825ª SESSÃO DE APRESENTAÇÃO DE DEFESA DE DISSERTAÇÃO DE MESTRADO PARA CONCESSÃO DE GRAU DE MESTRE EM ENGENHARIA ELÉTRICA NA ÁREA DE **SISTEMAS DE ENERGIA ELÉTRICA** REALIZADA ÀS NOVE HORAS E TRINTA MINUTOS DO DIA ONZE DE SETEMBRO DE DOIS MIL E VINTE E CINCO, VIA GOOGLE MEET, INTITULADA: **SÍNTESE DE CONTROLADORES HÍBRIDOS CLÁSSICO-NEURAIIS PARA SISTEMAS MIMO NÃO LINEARES: ESTUDO DE CASO EM HELICÓPTERO 2-DOF** APRESENTADA DURANTE **58** MINUTOS PELO CANDIDATO **MATHEUS MORAIS DA SILVA** DIANTE DA BANCA EXAMINADORA APROVADA PELO COLEGIADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA DO INSTITUTO DE TECNOLOGIA DA UNIVERSIDADE FEDERAL DO PARÁ, ASSIM CONSTITUÍDA: **PROF. DR. ANTONIO DA SILVA SILVEIRA** (ORIENTADOR - PPGE/ITEC/UFPA), **PROF. DR. ROBERTO CÉLIO LIMÃO DE OLIVEIRA** (AVALIADOR INTERNO - PPGE/ITEC/UFPA), **PROF. DR. WALTER BARRA JÚNIOR** (AVALIADOR INTERNO - PPGE/ITEC/UFPA), **PROF. DR. JOÃO VIANA DA FONSECA NETO** (AVALIADOR EXTERNO - UFMA) E **PROF.ª DR.ª REJANE DE BARROS ARAÚJO** (AVALIADORA EXTERNA - IFPA). CONCLUÍDOS OS TRABALHOS DE APRESENTAÇÃO E ARGUIÇÃO, A BANCA EXAMINADORA DECIDIU PELA **APROVAÇÃO** DO CANDIDATO. FOI CONCEDIDO UM PRAZO DE TRINTA DIAS, PARA O CANDIDATO EFETUAR AS CORREÇÕES SUGERIDAS PELA COMISSÃO EXAMINADORA E APRESENTAR O TRABALHO EM SUA REDAÇÃO DEFINITIVA. ESTA ATA NÃO VALE COMO OUTORGA DE GRAU DE MESTRADO, DE ACORDO COM O DEFINIDO NA RESOLUÇÃO 072/2004-CONSEPE. E, PARA CONSTAR, FOI LAVRADA A PRESENTE ATA, QUE VAI ASSINADA PELOS MEMBROS DA COMISSÃO E PELO CANDIDATO.

Documento assinado digitalmente



ANTONIO DA SILVA SILVEIRA  
Data: 16/09/2025 16:22:00-0300  
Verifique em <https://validar.iti.gov.br>

---

**Prof. Dr. Antonio da Silva Silveira**  
(Orientador – PPGE/ITEC/UFPA)

Documento assinado digitalmente



ROBERTO CELIO LIMAO DE OLIVEIRA  
Data: 16/09/2025 18:58:47-0300  
Verifique em <https://validar.iti.gov.br>

---

**Prof. Dr. Roberto Célio Limão de Oliveira**  
(Avaliador Interno - PPGE/ITEC/UFPA)

( X ) **VIDE VERSO:** Em caso de alteração do título pela Comissão Examinadora.



UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Documento assinado digitalmente



WALTER BARRA JUNIOR

Data: 16/09/2025 17:19:57-0300

Verifique em <https://validar.iti.gov.br>

---

**Prof. Dr. Walter Barra Júnior**  
(Avaliador Interno – PPGEE/ITEC/UFPA)

Documento assinado digitalmente



JOAO VIANA DA FONSECA NETO

Data: 16/09/2025 18:42:07-0300

Verifique em <https://validar.iti.gov.br>

---

**Prof. Dr. João Viana da Fonseca Neto**  
(Avaliador Externo – UFMA)

Documento assinado digitalmente



REJANE DE BARROS ARAUJO

Data: 16/09/2025 21:56:36-0300

Verifique em <https://validar.iti.gov.br>

---

**Prof.<sup>a</sup> Dr.<sup>a</sup> Rejane de Barros Araújo**  
(Avaliadora Externa – IFPA)

Documento assinado digitalmente



MATHEUS MORAIS DA SILVA

Data: 16/09/2025 16:28:38-0300

Verifique em <https://validar.iti.gov.br>

CANDIDATO: \_\_\_\_\_

**Por sugestão da Comissão Examinadora, o novo título passa a ser:**

**METODOLOGIA E SÍNTESE DE CONTROLADORES HÍBRIDOS CLÁSSICO-NEURAI APLICADOS A SISTEMAS MIMO NÃO LINEARES: ESTUDO DE CASO EM HELICÓPTERO 2-DOF**

Documento assinado digitalmente



ANTONIO DA SILVA SILVEIRA

Data: 16/09/2025 16:23:10-0300

Verifique em <https://validar.iti.gov.br>

---

**Prof. Dr. Antonio da Silva Silveira**  
(Orientador – PPGEE/ITEC/UFPA)

**( X ) VIDE VERSO:** Em caso de alteração do título pela Comissão Examinadora.

## **AGRADECIMENTOS**

Agradeço ao meu pai, Joabe Morais da Silva, pelo suporte e orientações na vida e no período do mestrado, pois sem os incentivos, ensinamentos e carinho não haveria como chegar a esta etapa da minha vida. A minha avó Maria de Nazaré Morais da Silva, que me encorajou a seguir os meus sonhos e afetos advindos de um bom coração, apesar de enfrentar várias complicações devido à sua idade avançada.

Aos meus amigos da Universidade Federal do Pará, Carlos, Paulo, Allan, Victor, Marcus, Marcelo, Rufin e Éder por todo o esforço, conselhos e tempo dados para me auxiliar na produção desta pesquisa, além das conversas e momentos de descontração. Aos meus amigos de longa data, Daniel, Lucas, Bruno, Fábio, que sem dúvida são irmãos que a vida me deu, e desejo levar esses vínculos de amizade para toda a vida, sempre me fazendo rir e dando conselhos.

Em especial, ao meu orientador Prof. Dr. Antonio Silveira, que teve um grande impacto nas minhas concepções, no meu amadurecimento pessoal e na tomada de decisões, dando conselhos e conhecimentos importantes para lapidar as ideias desta obra, bem como sobre concepções sobre a vida acadêmica e profissional. E agradeço aos demais não citados aqui, por todas as oportunidades e auxílios dados a mim para tornar este trabalho possível, que sabem o quanto fizeram parte da minha vida.

Agradeço a UFPA e a CAPES pelo suporte acadêmico e financeiro prestado durante o meu período como mestrando.

*"A vida me ensinou a nunca desistir.  
Nem ganhar, nem perder, mas procurar evoluir."  
(Dias de Luta, Dias de Glória - Charlie Brown Jr., 2005)*

## RESUMO

Esta dissertação apresenta um estudo de caso no desenvolvimento e aplicação de uma metodologia de controle híbrido baseada na integração entre estruturas clássicas descentralizadas do tipo RST-PI por redes neurais artificiais do tipo Perceptron Multicamadas (MLP), aplicada ao controle de posição de um helicóptero didático com dois graus de liberdade (H2DOF). A planta estudada possui características de um sistema multivariável (MIMO) não linear com acoplamentos dinâmicos entre os eixos de guinada e arfagem, sujeito a perturbações e variações paramétricas que dificultam a obtenção de modelos analíticos precisos e a sintonia eficiente de controladores convencionais locais. Visando superar essas limitações, primeiramente é realizada a identificação do sistema por meio do algoritmo de Mínimos Quadrados Recursivos (MQR) em espaço de estados, possibilitando a obtenção de modelos lineares aproximados para diferentes regiões operacionais, onde é efetuada a seleção do melhor modelo estimado por uma proposta de algoritmo de busca pela função custo do método MQR. A partir desse modelo estimado linearizado e médio, são projetados os controladores clássicos nas estruturas RST-PI e RST-PID por meio da técnica de alocação de polos, com os quais são realizados testes em simulação e no protótipo físico, avaliando-se os resultados das respostas de trajetória por métricas de desempenho como a Integral do Erro Médio Quadrático (ISE), Integral do Sinal de Controle Quadrático (ISU) e as variâncias dos sinais de erro e controle, além dos índices de robustez gerados pela análise das curvas de sensibilidade. Com base nos dados coletados dos experimentos no processo real, a rede neural é treinada de maneira *offline* com o objetivo de identificar, a cada instante, a região de operação da planta e adaptar dinamicamente os ganhos dos controladores, promovendo uma predição de melhoria no desempenho transitório e com menor esforço de controle. Os resultados experimentais indicaram que a abordagem híbrida proposta é capaz de reduzir oscilações, suavizar o sinal de controle e melhorar a resposta em malhas sujeitas a variações abruptas e efeitos de acoplamento. A metodologia desenvolvida demonstra viabilidade prática e técnica, indicando uma direção de como a aplicação de inteligência artificial pode otimizar estratégias de controle clássico em sistemas multivariáveis não lineares com características incertas e variantes.

**Palavras-chave:** Controle Híbrido. Redes Neurais Artificiais. Helicóptero 2-DOF. Alocação de Polos. Identificação por MQR. Controle RST-PI. Sistemas Multivariáveis Não Lineares.

## ABSTRACT

This dissertation presents the development and application of a hybrid control methodology based on the integration of decentralized classical RST-PI structures with artificial neural networks of the Multilayer Perceptron (MLP) type, applied to the position control of a didactic two-degrees-of-freedom (2-DOF) helicopter system. The studied plant exhibits characteristics of a nonlinear multivariable system with dynamic coupling between the yaw and pitch axes, subject to disturbances and parametric variations that hinder the derivation of accurate analytical models and the efficient tuning of local conventional controllers. To overcome these limitations, system identification was initially carried out using the Recursive Least Squares (RLS) algorithm in state space, allowing for the estimation of linearized models for different operating regions. A model selection strategy based on a cost function from the RLS method was proposed to choose the most suitable model. Based on the estimated linear model, classical controllers with RST-PI and RST-PID structures were designed using pole placement techniques. These controllers were tested in both simulations and real-time experiments, and their performance was evaluated using metrics such as mean integral of the squared error (ISE), mean integral of the squared control signal (ISU), variance of control and error signals, and robustness indicators based on sensitivity function analysis. Using experimental data collected from the real process, a neural network was trained offline to identify the plant's operating region at each instant and dynamically adapt the controller gains, aiming to enhance transient response and reduce control effort. The experimental results demonstrated that the proposed hybrid approach effectively reduced oscillations, smoothed the control signal, and improved tracking performance under abrupt variations and coupling effects. The developed methodology proved to be both technically and practically viable, offering a promising direction for the application of artificial intelligence to enhance classical control strategies in nonlinear multivariable systems with uncertain and time-varying dynamics. Future work includes exploring neural-based adaptation in centralized or more sophisticated controllers.

**Keywords:** Hybrid Control. Artificial Neural Networks. 2-DOF Helicopter. Pole Placement. Recursive Least Squares Identification. RST-PI Control. Nonlinear Multivariable Systems.

## LISTA DE FIGURAS

Figura 1 – Protótipo H2-DOF. . . . .	24
Figura 2 – Localização dos potenciômetros responsáveis pelos sinais analógicos de sensoriamento de posição angular. . . . .	25
Figura 3 – Representação dos eixos de rotação do Helicóptero 2-DOF. . . . .	26
Figura 4 – Diagrama Simplificado de Corpo Livre do Helicóptero 2-DOF. . . . .	28
Figura 5 – Fluxograma do algoritmo MQR para obter o vetor de estimação $\hat{\theta}$ . . . . .	36
Figura 6 – Esquema do mapeamento das regiões de estabilidade do sistema contínuo para o sistema discreto: (a) domínio $s$ ; e (b) domínio $z$ . . . . .	37
Figura 7 – Diagrama de blocos de um sistema de controle. . . . .	38
Figura 8 – Estrutura PID Paralelo Digital. . . . .	41
Figura 9 – Diagrama de blocos simplificado do sistema em malha fechada discreto. . . . .	42
Figura 10 – Estrutura RST na Forma Canônica Discreta. . . . .	43
Figura 11 – Estrutura de Projeto Descentralizado para os Controladores Clássicos. . . . .	49
Figura 12 – Estrutura da Rede Neural Perceptron de Múltiplas Camadas (MLP). . . . .	55
Figura 13 – Estrutura de Projeto da Rede Neural (6-8-6). . . . .	60
Figura 14 – Estrutura da Síntese do Controlador Clássico Neural. . . . .	61
Figura 15 – Trajetória de Referência para os Experimentos no H2-DOF. . . . .	62
Figura 16 – Aproximação da Trajetória pela Combinação das Referências Angulares. . . . .	63
Figura 17 – Ensaio em Malha Aberta. . . . .	67
Figura 18 – Tratamento dos Dados do Ensaio em Malha Aberta. . . . .	68
Figura 19 – Resultado Comparativo da Saída Simulada pelo Melhor Modelo Estimado. . . . .	70
Figura 20 – Validação do Melhor Modelo Estimado com Novos Dados do H2-DOF. . . . .	71
Figura 21 – Comparação da resposta ao degrau no sistema MIMO. . . . .	73
Figura 22 – Análise de resposta em frequência. . . . .	74
Figura 23 – Estimativa de Conversão dos Sinais de Saída de Tensão para Ângulo. . . . .	76
Figura 24 – Resposta ao Degrau em Malha Aberta da Planta e da Dinâmica das Especificações Desejada. . . . .	77
Figura 25 – Resultados Obtidos via Simulação com a Sintonia dos Controladores PID no Modelo Estimado. . . . .	78
Figura 26 – Resultados Obtidos via Experimento Real com a Sintonia dos Controladores PID no Protótipo H2-DOF. . . . .	78
Figura 27 – Curvas de Sensibilidade dos Projetos dos Controladores PID. . . . .	79
Figura 28 – Resposta ao degrau da planta e da dinâmica desejada reajustada. . . . .	80

Figura 29 – Resultados Obtidos via Simulação com a Nova Sintonia para os Controladores PID no Modelo Estimado. . . . .	81
Figura 30 – Resultados Obtidos via Experimento Real com a Nova Sintonia para os Controladores PID no Protótipo H2-DOF. . . . .	81
Figura 31 – Curvas de Sensibilidade do Nova Sintonia dos Controladores PID. . . . .	82
Figura 32 – Resposta ao Degrau da Planta e da Dinâmica Desejada (Projeto 1). . . . .	83
Figura 33 – Resposta ao Degrau da Planta e da Dinâmica Desejada (Projeto 2). . . . .	83
Figura 34 – Resultados Obtidos via Simulação com os Controladores PI do Projeto 1 no Modelo Estimado. . . . .	84
Figura 35 – Resultados Obtidos via Simulação com os Controladores PI do Projeto 2 no Modelo Estimado. . . . .	84
Figura 36 – Curvas de Sensibilidade dos Controladores PI do Projeto 1. . . . .	85
Figura 37 – Curvas de Sensibilidade dos Controladores PI do Projeto 2. . . . .	85
Figura 38 – Resultados Obtidos via Experimento Real com os Controladores PI do Projeto 1 no Protótipo H2-DOF. . . . .	86
Figura 39 – Resultados Obtidos via Experimento Real com os Controladores PI do Projeto 2 no Protótipo H2-DOF. . . . .	86
Figura 40 – Base de dados de Entrada ( $x_j$ ). . . . .	88
Figura 41 – Desempenho da Rede Neural nas Etapas de Treinamento, Validação e Teste. . . . .	89
Figura 42 – Resultados Obtidos via Simulação com os Controladores PI Neurais no Modelo Estimado. . . . .	90
Figura 43 – Resultados Obtidos via Experimento Real com os Controladores PI Neurais no Protótipo H2-DOF. . . . .	91
Figura 44 – Predição dos Termos RST pela RNA nos Experimentos Simulado e Real. . . . .	91
Figura 45 – Adaptação Ganhos dos Controladores PI nos Experimento Simulado e Real. . . . .	92
Figura 46 – Comparação Entre os Controladores PI dos Experimentos Reais. . . . .	93
Figura 47 – Variação dos Índices de Robustez Produzidos pela Adaptação dos Ganhos no Controlador PI-NN nos Experimentos Simulados Usando o Método Monte Carlo com 1000 Ruídos Gaussianos de Variância $\sigma_v^2$ . . . . .	95
Figura 48 – Espectro dos Índices de Robustez Produzidos pela Adaptação dos Ganhos no Controlador PI-NN no Experimento Real. . . . .	96
Figura 49 – Trajetória Tridimensional dos Ângulos Produzidos pelos Controladores PI. . . . .	97

## LISTA DE TABELAS

Tabela 1 – Índices de Identificação do $MQR_{SS}$ por malha do processo. . . . .	70
Tabela 2 – Índices de Validação do $MQR_{SS}$ por malha do processo. . . . .	71
Tabela 3 – Parâmetros Desejados para Sintonia dos Controladores PID. . . . .	76
Tabela 4 – Índices dos Controladores Obtidos nos Experimentos Simulado e Real	79
Tabela 5 – Parâmetros Desejados Reajustados para nova sintonia dos Controladores PID . . . . .	80
Tabela 6 – Índices dos Novos Controladores Obtidos nos Experimentos Simulado e Real. . . . .	82
Tabela 7 – Parâmetros Desejados para Sintonia dos Controladores PI do Projeto 1 e 2. . . . .	83
Tabela 8 – Índices dos Controladores PI Obtidos nos Experimentos Simulado e Real com os Parâmetros do Projeto 1. . . . .	87
Tabela 9 – Índices dos Controladores PI Obtidos nos Experimentos Simulado e Real com os Parâmetros do Projeto 2. . . . .	87
Tabela 10 – Índices de Desempenho Obtidos no Treinamento da Rede com a Base de Dados dos Projetos dos Controladores PI. . . . .	88
Tabela 11 – Índices dos Controladores Neurais PI Obtidos nos Experimentos Simulado e Real. . . . .	92
Tabela 12 – Comparação dos índices de desempenho dos controladores PI (Experimento Real). . . . .	94
Tabela 13 – Comparação percentual dos índices de desempenho dos controles PI Fixos em relação ao PI-NN (referência). . . . .	94

## LISTA DE ABREVIATURAS E SIGLAS

UFPA	Universidade Federal do Pará
LACOS	Laboratório de Automação e Controle de Sistemas
VANT	Veículos Aéreos Não Tripulados
UAV	Unmanned Aerial Vehicle
ROV	Remotely Operated Vehicle
MIMO	Multiple Input, Multiple Output
SISO	Single Input, Single Output
DOF	Degree of Freedom
H2-DOF	Helicóptero com Dois Graus de Liberdade
PID	Proportional Integral Derivative
PI	Proportional Integral
RST	Reference Signal Tracking
RST-PI	Reference Signal Tracking for PI
RST-PID	Reference Signal Tracking for PID
RNA	Redes Neurais Artificiais
NN	Neural Network
BP	Backpropagation
MLP	Multilayer Perceptron
PID-NN	Proportional Integral Derivative with Neural Network
PI-NN	Proportional Integral with Neural Network
E-FRIT	Extended Fast Recursive Input Training
NN-E-FRIT	Neural Network with Extended Fast Recursive Input Training
GDNN	General Dynamic Neural Networks
FFNN	Feedforward Neural Network
RNR	Redes Neurais Recorrentes

RNN	Recurrent Neural Networks
RL	Reinforcement Learning
MSE	Mean Squared Error
RMSE	Root Mean Squared Error
CCW	Counter-Clockwise
CW	Clockwise
LTI	Linear Time-Invariant
DSP	Digital Signal Processors
MRAC	Model Reference Adaptive Control
LQR	Linear-Quadratic Regulator
LQG	Linear-Quadratic Gaussian
DCL	Diagrama de Corpo Livre
RLS	Recursive Least Squares
MQR	Mínimos Quadrados Recursivos
ARX	AutoRegressive with eXogenous input
NARX	Nonlinear AutoRegressive with eXogenous input
IA	Inteligência Artificial
RL	Reinforcement Learning
SS	Space State
RGA	Relative Gain Array
FOPDT	First Order Plus Dead Time
ZOH	Zero Order Hold
ISE	Integral Squared Error
ISU	Integral Squared Control Signal
GM	Gain Margin
PM	Phase Margin

GPC          Generalized Predictive Controller

LSTM        Long Short-Term Memory

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>17</b>
1.1	VISÃO GERAL	17
1.2	JUSTIFICATIVA	20
1.3	OBJETIVOS	22
<b>1.3.1</b>	<b>Objetivo Geral</b>	<b>22</b>
<b>1.3.2</b>	<b>Objetivos Específicos</b>	<b>22</b>
1.4	METODOLOGIA	22
<b>2</b>	<b>PROTÓTIPO DE HELICÓPTERO COM DOIS GRAUS DE LIBERDADE (H2-DOF)</b>	<b>24</b>
2.1	HELICÓPTERO 2-DOF (EQUIVALENTE DINÂMICO)	25
<b>2.1.1</b>	<b>Diagrama de Corpo Livre (DCL)</b>	<b>27</b>
2.2	IDENTIFICAÇÃO POR MÍNIMOS QUADRADOS RECURSIVO	29
<b>2.2.1</b>	<b>Aproximação em Espaço de Estados</b>	<b>29</b>
2.2.1.1	Identificação aplicada ao sistema H2-DOF	31
2.3	ALGORITMO DE SELEÇÃO DO MELHOR MODELO ESTIMADO PELO MQR	32
<b>3</b>	<b>TEORIA DE CONTROLE</b>	<b>37</b>
3.1	CONTROLE PID	39
3.2	PROJETO DO CONTROLADOR POR ALOCAÇÃO DE POLOS	43
<b>3.2.1</b>	<b>Considerações de Projeto para o Modelo do Sistema Multivariável</b>	<b>45</b>
3.2.1.1	Controlabilidade e Observabilidade	45
3.2.1.2	RGA e pré-compensação	46
<b>3.2.2</b>	<b>Considerações para o Projeto na Estrutura RST-PID</b>	<b>49</b>
<b>3.2.3</b>	<b>Considerações para o Projeto na Estrutura RST-PI</b>	<b>52</b>
3.3	REDES NEURAIS ARTIFICIAS	54
<b>3.3.1</b>	<b>Algoritmo <i>Backpropagation</i></b>	<b>55</b>
3.4	PROJETO DE SÍNTESE DO CONTROLADOR CLÁSSICO NEURAL (HIBRIDIZAÇÃO)	58
3.5	ANÁLISE DE DESEMPENHO E ROBUSTEZ	63
<b>3.5.1</b>	<b>Índices de Desempenho</b>	<b>63</b>
<b>3.5.2</b>	<b>Índices de Robustez</b>	<b>64</b>
<b>4</b>	<b>SIMULAÇÕES E RESULTADOS</b>	<b>66</b>
4.1	IDENTIFICAÇÃO	66
<b>4.1.1</b>	<b>Análise de Modelagem</b>	<b>67</b>
<b>4.1.2</b>	<b>Modelo na Forma Polinomial</b>	<b>71</b>
4.2	CONTROLE	75
<b>4.2.1</b>	<b>Análise de Projeto dos Controladores PID</b>	<b>76</b>

4.2.2	Análise de Projeto dos Controladores PI . . . . .	82
4.2.3	Análise do Projeto de Síntese do Controlador Neural . . . . .	87
4.2.4	Análise Comparativa . . . . .	92
5	CONCLUSÃO . . . . .	98
	Referências . . . . .	100

# 1 INTRODUÇÃO

## 1.1 VISÃO GERAL

A teoria de controle desempenha um papel fundamental na produção de sistemas automatizados com tecnologias cada vez mais complexas, que vêm demandando a presença contínua de mais sensores e atuadores refinados em seus processos, com o objetivo de se obter uma maior eficiência e rapidez nos resultados no ambiente industrial e comercial. Neste contexto, no setor aeroespacial, os Veículos Aéreos Não Tripulados (VANT), como os drones e helicópteros, se caracterizam por sua complexidade desafiadora para aplicação de controle e modelagem matemática, uma vez que esses veículos se configuram como sistemas de múltiplas entradas e múltiplas saídas (*Multiple Input, Multiple Output* - MIMO), onde cada atuador pode influenciar simultaneamente nos graus de liberdade (*Degree of Freedom* - DOF) do sistema, como os de posição, altitude e orientação (Stevens; Lewis; Johnson, 2016). Essa forte interdependência entre as variáveis de estado, aliadas as dinâmicas não lineares e incertezas externas ao sistema, impõe desafios significativos para o desenvolvimento de controladores clássicos a parâmetros fixos capazes de garantir estabilidade e precisão, sem gerar degradações de desempenho, nas mudanças dos segmentos das referências.

Neste trabalho, propõe-se o uso de redes neurais como ferramentas de adaptação em tempo real aplicadas a sistemas de controle baseados em estruturas clássicas, preservando a simplicidade e a interpretabilidade dos controladores PI e, ao mesmo tempo, incorporando uma solução flexível e robusta para o controle de um protótipo de veículo aéreo não tripulado, com dinâmicas simplificadas e próximas as de um helicóptero, com apenas dois graus de liberdade (H2-DOF).

Do ponto de vista analítico, segundo Åström e Wittenmark (2013) a teoria de controle clássico foi consolidada entre as décadas 60 e 80, onde houve o aumento exponencial do poder computacional devido à corrida espacial, que possibilitou a introdução de computadores digitais no controle de processos. Entretanto, apesar da importância da teoria de controle clássico, a evolução subsequente para técnicas mais sofisticadas, como o controle ótimo e o controle adaptativo, permitiu ampliar a abordagem de problemas cada vez mais complexos em sistemas não-lineares e sujeitos a incertezas paramétricas (Franklin; Powell; Emami-Naeini, 2019). Estes avanços teóricos não apenas ampliaram o escopo de aplicação dos sistemas de controle, mas também prepararam o terreno para sua integração com técnicas emergentes de inteligência artificial.

Em paralelo à evolução da teoria de controle, o desenvolvimento das Redes Neurais Artificiais (RNA) também seguiu uma trajetória notória. Partindo disso, o modelo pioneiro de McCulloch e Pitts (1943), estabeleceu as bases matemáticas para a modelagem de neurônios artificiais, o campo evoluiu para o Perceptron de Rosenblatt (1958)

e, após um período de relativo desinteresse provocado pela identificação de suas limitações fundamentais que foram descritas e corrigidas por Minsky e Papert (1969), ressurgiu com vigor na década de 1980 graças ao desenvolvimento do algoritmo *Back-propagation* (BP) para o treinamento de redes neurais do tipo Perceptron Multicamadas (Multilayer Perceptron - MLP) (Rumelhart; Hinton; Williams, 1986; Werbos, 1990). A demonstração formal do teorema da aproximação universal de Cybenko (1989) e Hornik (1991) representaram um marco teórico fundamental, ao estabelecer de forma definitiva a capacidade das RNAs de representar comportamentos não-lineares complexos (Bishop; Nasrabadi, 2006).

Assim, segundo Samad et al. (2020), o controlador PID (Proporcional-Integral-Derivativo) ainda é a estratégia mais utilizada na indústria, tendo uma representatividade de 91% nas malhas de processo, onde até suas variações simplificadas, como o PI são usadas devido à sua simplicidade de implementação e por serem economicamente mais rentáveis. No entanto, sua eficácia pode ser degradada em sistemas com alta não-linearidade, dinâmicas variantes ou sujeitos a incertezas não modeladas (Skogestad; Postlethwaite, 2005). A partir dessa limitação, pesquisadores passaram a investigar formas de complementar a estrutura clássica com técnicas de inteligência artificial e métodos de otimização, entre elas encontra-se o uso de redes neurais de retropropagação, recorrentes (RNR) e aprendizado por reforço (*Reinforcement Learning* - RL), denominando-os de controladores inteligentes.

Em Hernández et al. (2016) desenvolveram um sistema de autoajuste neural aplicado ao controle de veículos submarinos autônomos, onde se apresentou uma arquitetura de rede neural que ajustava dinamicamente os ganhos PID conforme a mudança dos parâmetros de ROV (*Remotely Operated Vehicle*) e as perturbações ambientais, onde foi considerado um processo com 6 graus de liberdade. Os experimentos em ambientes simulados e reais mostraram que o sistema neural foi capaz de manter a estabilidade da embarcação mesmo sob perturbações hidrodinâmicas intensas, com um erro médio quadrático (MSE) de 50% menor, em relação ao controle PID a parâmetros fixos, e 3,038% mais eficiente em termos energéticos.

Posteriormente, Zribi, Chtourou e Djemel (2018) apresentaram uma proposta de controlador do tipo PID projetado com redes neurais (denominado PID-NN), especificamente, para lidar com processos não lineares. A rede neural utiliza um método de otimização baseado em gradiente descendente, onde testes conduzidos em um processo isotérmico (CSTR) e outro não isotérmico revelaram uma redução significativa no tempo de acomodação de 17% e MSE de 11%, em comparação ao PID convencional. Os resultados evidenciam as redes neurais como uma ferramenta promissora de adaptação do termo *momentum* para ajustes em tempo real dos parâmetros PID, melhorando o desempenho em processos dinâmicos e com alta incerteza.

Além disso, abordagem projetada em Kinoshita, Wakitani e Ohno (2018) apre-

senta uma contribuição baseada em aprendizado offline por meio do algoritmo E-FRIT (*Extended Fast Recursive Input Training*), aplicando-a ao ajuste de ganhos PID durante a operação. O sistema neural treinado (NN-E-FRIT) possui uma norma de avaliação com um termo de penalidade para alterações das variáveis manipuladas, que promete controladores mais estáveis em comparação ao método convencional. Aplicado a um sistema de controle de nível com não linearidades e atrito estático, o controlador proposto obteve uma melhoria de 23% no tempo de resposta em relação ao método PID com E-FRIT fixo. Desta forma, o trabalho destaca o potencial das redes neurais em acelerar e refinar o processo de sintonia e propõe o estudo futuro da estabilidade estrita do sistema de controle, que é um tópico importante para sistemas de controle.

No campo da interpretabilidade, Günther et al. (2020) propõem o uso de redes neurais dinâmicas interpretáveis (*General Dynamic Neural Networks - GDNN*) como ferramenta para ajuste de parâmetros PID. O diferencial da proposta reside na capacidade da rede não apenas ajustar os parâmetros automaticamente, mas também de fornecer explicabilidade das decisões tomadas, facilitando a aceitação por engenheiros de controle. O sistema foi testado em 16 cenários-padrão da indústria (*benchmarks*), e superou o controlador PID tradicional em 15 desses casos, obtendo melhorias médias no desempenho global, pelo termo da raiz do erro médio quadrático (*Root Mean Squared Error - RMSE*).

Em estudos mais recentes, Sharifi e Alasty (2023) aplicaram uma estratégia de rede neural híbrida baseada em RL com estrutura *model-free-actor-critic*, voltada para o controle de atitude em quadricópteros. Nesta abordagem, os ganhos do controlador PID são ajustados online com base nas interações do processo com o ambiente, sem a necessidade de um modelo matemático preciso do sistema. Os testes indicaram uma melhora de até 35% no RMSE, além de reduções significativas nas respostas transitórias do sistema e no tempo de estabilização frente às perturbações externas. A estrutura neural demonstrou rápida adaptação, mesmo em cenários de forte variabilidade aerodinâmica, reforçando a utilidade dessas técnicas em sistemas dinâmicos e altamente instáveis.

Em complemento aos estudos recentes, Costa (2024) em sua dissertação de mestrado, aplicou redes neurais artificiais para sintonia de controladores PI e PID em motores de corrente contínua utilizados em ambientes industriais. A metodologia consistiu no treinamento das redes com base em conjuntos estabilizantes de parâmetros obtidos por simulações em malha fechada. Os resultados alcançaram acurácia superior a 95% para o PI e 97,5% para o PID, demonstrando excelente capacidade de generalização e confiabilidade nos ajustes realizados. Além disso, os controladores ajustados pela RNA apresentaram respostas com menor oscilação, redução no tempo de subida e melhoria no consumo energético do motor, quando comparados aos métodos de sintonia tradicionais, como Ziegler-Nichols.

Portanto, tais evidências reforçam o potencial de aplicação prática de técnicas de aprendizado de máquina em conjunto à teoria clássica de controle no contexto atual, em processos de diversas áreas.

## 1.2 JUSTIFICATIVA

Como esta dissertação de mestrado apresenta uma abordagem de proposição entre controladores da literatura de controle clássico (PID e PI) sintetizados por alocação de polos (via estrutura RST, *Reference-Signal-Tracking*) e a hibridização destes controladores com o uso de redes neurais. Para se projetar e aplicar esses controladores, via simulação e no experimento real, faz-se necessário o uso de métodos de modelagem matemática de sistemas para se estipular um modelo, em que não se tem conhecimento pleno dos parâmetros e das leis físicas que regem o sistema, sendo considerado como um processo de caixa preta, com o objetivo de compreender as dinâmicas dos sistemas, que a partir dos parâmetros estimados torna-se possível analisar o comportamento do processo (Ogata, 1995).

Além disso, a convergência entre dois campos do conhecimento apresentados neste estudo ocorre como resposta a desafios práticos cada vez mais complexos no controle de sistemas dinâmicos, particularmente, em situações onde a dinâmica do sistema apresenta características não-lineares significativas e difíceis de modelar de forma analítica para gerar um modelo que consiga representar a dinâmica total do processo.

Dessa forma, os processos que são apresentados nesta pesquisa possuem condições operacionais que variam de forma substancial durante a operação de controle. Assim, é possível verificar que estes sistemas possuem incertezas nos parâmetros dos seus modelos, que os caracterizam como não lineares. Neste contexto, o objetivo principal da rede neural é produzir uma ação de ajuste adaptativo dos ganhos dos controladores PI-RST, de forma eficaz para satisfazer as especificações do projeto, ao ajustar os ganhos de controle com base nas mudanças nas regiões de operação do processo para reduzir possíveis sobressinais, variações no sinal de controle, variações da saída quando na presença de perturbações, que dificultam o seguimento de referência.

O estudo e a aplicação dessa integração têm produzido resultados positivos em áreas que englobam a supervisão e aquisições de dados, em sistemas industriais e comerciais, apesar do uso destas ferramentas não possuir uma metodologia concreta entre os autores. Desta maneira, trabalhos como o de Zhou, Liu, Yu et al. (2010) apresentam visão comparativa no controle de temperatura de um sistema de aquecimento solar ativo em uma residência, onde a inserção da teoria de controle PID possibilitou aprimorar o objetivo do projeto feito anteriormente apenas com uma rede neural de retropropagação, em que foi averiguado que o controlador PID neural,

para uma referência de 18 °C, desempenhou sem sobressinais e com redução das perdas energéticas ocasionadas por aquecimento desnecessário, enquanto a técnica do projeto anterior apresentou um sobressinal de 7,2%, com uma temperatura de pico no valor de 19,33 °C.

Ademais, estudos mais condensados, como o de Madebo et al. (2024), apresentam uma contribuição relevante ao propor uma arquitetura de controle híbrida para um UAV, combinando RNA com Controle Adaptativo por Modelo de Referência (Model Reference Adaptive Control - MRAC). A pesquisa destaca-se pelo desenvolvimento de um modelo dinâmico baseado no método de Newton-Quaternion, que elimina singularidades matemáticas e incorpora efeitos aerodinâmicos complexos. A solução implementou uma abordagem inovadora ao integrar redes neurais feedforward (FFNN) para controle de posição somadas a Redes Neurais Recorrentes para regulação de atitude, complementada por um mecanismo de aprendizado online via retropropagação. Os resultados experimentais demonstram a superioridade do método proposto em relação a controladores LQR (Linear-Quadratic Regulator) e MRAC convencionais, evidenciando maior precisão no rastreamento de trajetórias e robustez frente a distúrbios externos e variações paramétricas, mantendo ainda um perfil de esforço de controle energeticamente eficiente.

Como essa hibridização é uma tecnologia de controle em estado de fluxo, não havendo uma forma consolidada do que ela é ou mesmo do que será, de tal forma que se justifica esta pesquisa pelas contribuições para a área de controle de processos, em um protótipo de veículo aéreo não tripulado, que objetiva produzir dinâmicas aproximadas as de um helicóptero com dois graus de liberdade (H2-DOF), espera-se que as contribuições desta pesquisa também possam dar *insights*, utilizando estas metodologias de controladores neurais que estão sendo exploradas no decorrer do século XXI, sobre as possíveis aplicações das redes neurais que estão ganhando espaço em tempos modernos.

Por fim, a metodologia adaptativa destes controladores neurais hibridizados, além de fazer as leituras de variáveis do processo em malha fechada, para que se possa produzir uma adaptação dos ganhos do controlador a cada iteração, permite a tentativa da rede neural de prever a mudança de regiões geradas pelos pontos de operação, com base no treinamento offline com dados operacionais obtidos nos experimentos reais com os controladores locais, para verificar se há melhora no desempenho de controle em relação às estratégias clássicas que utilizam os ganhos fixos.

## 1.3 OBJETIVOS

### 1.3.1 Objetivo Geral

Projetar sistemas de controle de posição na estrutura RST-PI integrados com redes neurais de retropropagação, para um processo não-linear de duas entradas e duas saídas.

### 1.3.2 Objetivos Específicos

- Apresentar o modelo do H2-DOF e o estudo completo dos controladores clássicos e neural, referenciando suas características e principais componentes.
- Estabelecer uma metodologia para síntese do controlador neural e possíveis formas de aprimoramento, bem como a estrutura de controle a ser utilizada, considerando incertezas paramétricas presentes no processo a ser controlado.
- Construir e simular os controladores propostos, em ambiente de simulação, a partir dos modelos lineares e não lineares do processo.
- Testar os controladores no processo real e analisar os resultados obtidos.

## 1.4 METODOLOGIA

Esta dissertação de mestrado tem como principal objetivo produzir um estudo de caso e investigar estratégias de controladores clássicos com redes neurais no processo não-linear denominado H2-DOF. Assim, é necessário controlar os ângulos de guinada e arfagem para se realizar a trajetória desejada de forma autônoma, frente às perturbações internas e externas no processo.

Primeiramente, é investigado o método de identificação no processo H2-DOF de tal forma que, nesse estudo prévio, com base na aquisição de dados obtida pela comunicação serial em um ESP-32, será produzido um modelo linearizado pelos Mínimos Quadrados Recursivo (MQR). Assim, por meio da pesquisa bibliográfica produzida sobre este método de identificação, a modelagem do processo, incertezas do processo e as influências do acoplamento dinâmico dos atuadores deste sistema sobre as malhas de controle serão descritas.

De posse do modelo matemático, serão desenvolvidas simulações em ambiente computacional fazendo uso do software Matlab/Simulink, de modo a desenvolver um simulador com as características não lineares do sistema, objetivando assemelhar a resposta simulada ao sistema real.

Na etapa seguinte, serão estudados os projetos dos controladores e a análise de robustez às incertezas paramétricas, que podem ser caracterizadas como ruído de

sensor na saída. Dessa forma, para a organização do problema, serão estabelecidas as ferramentas matemáticas necessárias, onde no problema de estabilização é utilizado o conceito de alocação de polos no projeto de controle para produzir controladores locais com dinâmicas diferentes para os pontos de operação que serão testados em simulação e na planta real.

A estrutura de controle do sistema é definida como a regulação dos estados dinâmicos do sistema por meio da realimentação, onde tal estrutura é expandida em seu modelo a fim de atender a mais objetivos de controle além da regulação, como seguimento de referência e rejeição de perturbação. Por fim, é realizada a síntese do controlador, onde a estrutura de controle é particionada como dois sistemas de uma entrada e uma saída (Single Input, Single Output - SISO), por meio de uma metodologia de pré-compensação do ganho direto das malhas do sistema MIMO, para realização do projeto de forma desacoplada para uma dada faixa de frequências de interesse.

Na etapa de simulação, são analisados os comportamentos do modelo não-linear com controladores sintonizados por alocação de polos, em estruturas RST-PI e RST-PID, utilizando índices de desempenho, que estão relacionados a esforço de controle e seguimento de referência. Nessas simulações, serão realizados testes de variação de referência e perturbações na saída. Além disso, é feita a utilização dos dados resultantes da simulação para treinar a rede neural com o objetivo de gerar um controlador neural que adapta os ganhos internos dos controladores clássicos, por meio do reconhecimento das mudanças de regiões de operação, a fim de atenuar a degradação de desempenho do controlador provocada pelos diferentes pontos de operação no processo não-linear. Portanto, por meio de variáveis como as referências (do ângulo de guinada e arfagem), os sinais de saída e o erro das malhas de controle, que são verificados a cada instante pela rede neural para adaptar os ganhos dos controladores.

Enfim, na etapa experimental será realizada a aplicação dos controladores no sistema real, com a aquisição dos dados resultantes para a aplicação da metodologia do controlador neural e discussão dos resultados obtidos dos controladores em termos de desempenho e possíveis formas de melhoria e refinamento dessa metodologia para trabalhos futuros.

As partes subsequentes do trabalho estão organizadas da seguinte maneira: no capítulo 2 se apresenta o processo e o método de identificação. No capítulo 3 mostra-se a Teoria de Controle, relacionada às técnicas de controladores RST, RST-PI e RST-PID com sintonia dos ganhos do projeto via alocação de polos para o modelo estimado do protótipo H2-DOF. Ainda no capítulo 3, é apresentada a rede neural de avanço do tipo MLP, que é treinada pelo algoritmo BP, para a síntese de projeto do controlador neural. O capítulo 4 traz os resultados obtidos e discussões do desempenho para cada caso. No capítulo 5, as conclusões e propostas para trabalhos futuros são apresentados.

## 2 PROTÓTIPO DE HELICÓPTERO COM DOIS GRAUS DE LIBERDADE (H2-DOF)

Neste capítulo será estudada a dinâmica de uma planta didática com dois graus de liberdade (2-DOF), desenvolvida com base em trabalhos anteriores em Breganon (2009) e Nascimento (2016), onde o presente protótipo do sistema foi desenvolvido no Laboratório de Automação e Controle de Sistemas (LACOS), na Universidade Federal do Pará, pelo seguinte pesquisador: Eng. Haroldo Martins Ramos Filho. Desta forma, o objetivo do protótipo produzido é servir como uma ferramenta didática para aplicações focadas em controle e identificação, além de incentivar pesquisas que englobem sistemas MIMO e controle não-linear. Uma foto do protótipo é mostrada na Figura 1.

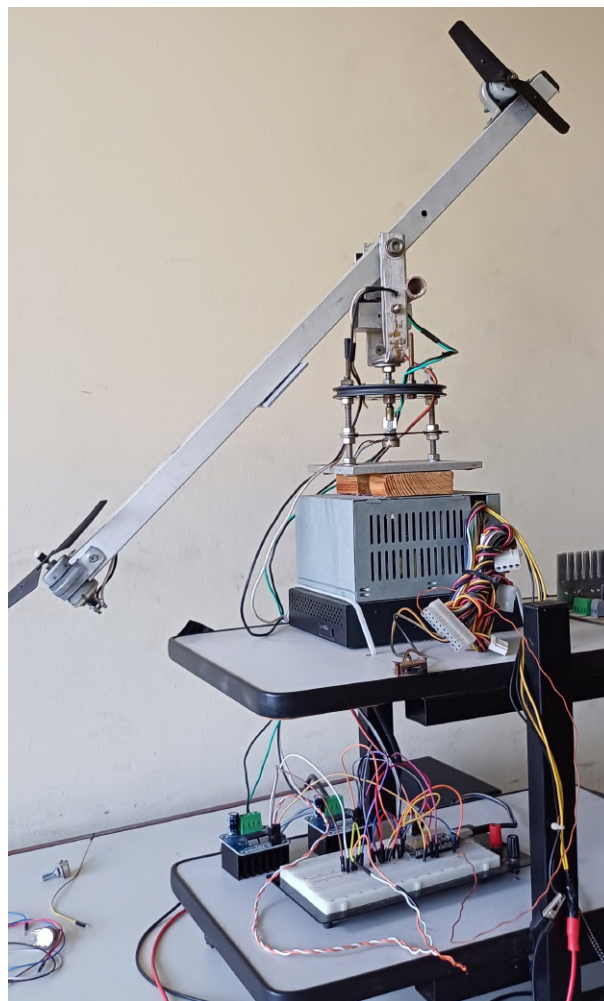


Figura 1 – Protótipo H2-DOF.

Fonte: Autor.

O Helicóptero 2-DOF possui dois atuadores baseados em motores de corrente contínua com hélices acopladas em seus eixos, que são responsáveis pela mudança dos ângulos de guinada e arfagem, sendo fixados em uma haste de metal que está acoplada a uma base mecânica que possui rolamentos nos eixos vertical e lateral para reduzir as influências causadas por atritos mecânicos. Além disso, o processo

tem potenciômetros fixados no final dos eixos para captação das mudanças angulares produzidas pelo acionamento dos rotores da planta, onde o sinal analógico é enviado para um microcontrolador ESP-32 responsável pela aquisição e envio de dados, que pode ser verificado na Figura 2.

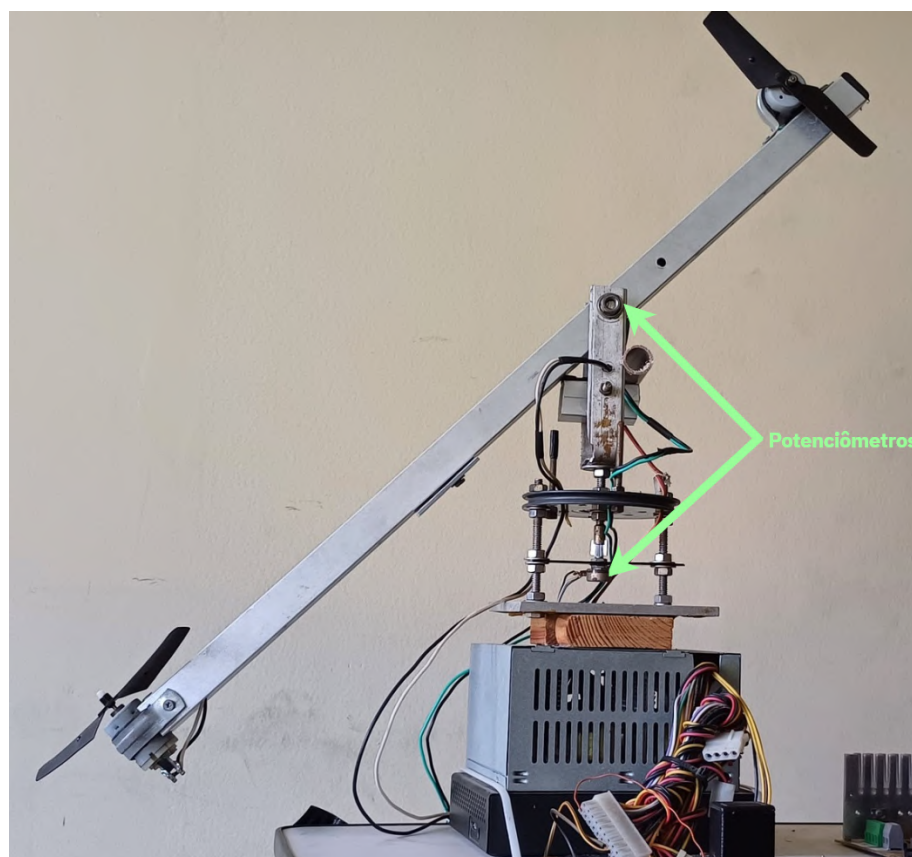


Figura 2 – Localização dos potenciômetros responsáveis pelos sinais analógicos de sensoriamento de posição angular.

Fonte: Autor.

Desta maneira, essa planta didática pode ser descrita como um equivalente dinâmico de um helicóptero de pequeno porte com dois graus de liberdade (H2-DOF), onde o movimento de rolagem não pode ser modificado devido à fixação da haste no eixo que produz esta ação, assim, limitando os graus de liberdade do corpo aos eixos responsáveis pelos movimentos de arfagem e guinada (Nascimento, 2016).

## 2.1 HELICÓPTERO 2-DOF (EQUIVALENTE DINÂMICO)

Considerado um sistema dinâmico complexo, o helicóptero não tripulado com dois graus de liberdade apresenta características multivariáveis, não lineares e de forte acoplamento entre suas variáveis (Zhu et al., 2018; CRUZ et al., 2019). Assim, torna-se um desafio representar a totalidade de sua dinâmica em um modelo matemático único, que pode ser explorado por meio de metodologias de identificação comumente

aplicadas em veículos aéreos não tripulados de pequeno porte. Um dos métodos utiliza fundamentos teóricos para derivar equações diferenciais, que podem ser lineares ou não lineares. Já a outra abordagem consiste em realizar experimentações para obter dados de entrada e saída, como os provenientes de testes de voo reais, os quais podem ser utilizados por Redes Neurais ou Mínimos Quadrados Recursivo, para a estimação dos parâmetros de um modelo (Wu et al., 2022). No entanto, este segundo método com redes neurais costuma exigir conhecimento técnico avançado ou uma base extensa de dados empíricos, que não serão utilizados neste trabalho para fins de estimação paramétrica do modelo.

O diagrama de eixos de coordenadas do sistema H2-DOF, apresentado na Figura 3, mostra os três graus de liberdade que um helicóptero comumente possui. Desta maneira, ao abordar o funcionamento do equivalente dinâmico, é possível extrair conceitos básicos relacionados às dinâmicas produzidas nos eixos a partir do acionamento dos atuadores do helicóptero. Entretanto, como expresso anteriormente, para manter a equivalência dinâmica em relação ao protótipo Helicóptero 2-DOF, não há rotação em torno do eixo longitudinal (eixo x), reduzindo a ação do corpo em apenas dois graus de liberdade.

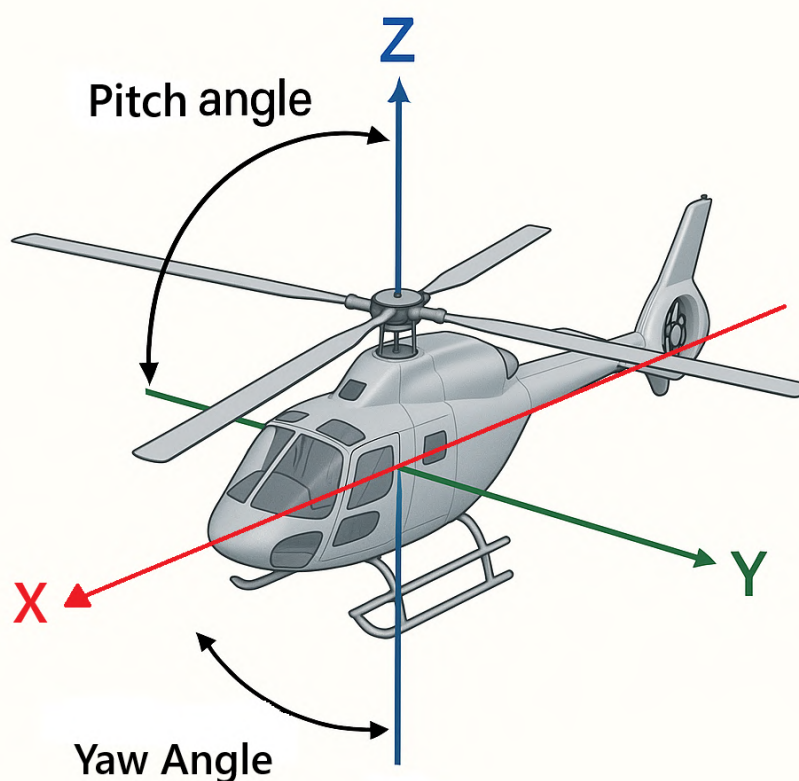


Figura 3 – Representação dos eixos de rotação do Helicóptero 2-DOF.

Fonte: Autor com auxílio de IA.

Portanto, o modelo do processo H2-DOF é projetado para simular duas dinâmicas principais de um helicóptero real: arfagem (pitch) e guinada (yaw), correspondentes aos movimentos em torno dos eixos lateral (Y) e vertical (Z), respectivamente. Embora o modelo H2-DOF não represente a dinâmica em torno do eixo longitudinal (X), que no helicóptero real é responsável pelo rolamento (roll), torna-se útil para estudo de sistemas MIMO, já que há acoplamento dinâmico entre os eixos Y e Z. Esse efeito de acoplamento cruzado exige que o controlador considere a influência mútua entre os dois movimentos, tornando o problema mais desafiador e realista do ponto de vista do controle.

Assim, o funcionamento de um helicóptero envolve principalmente dois rotores: o rotor principal (primário) e o rotor de cauda (secundário), sendo ambos fundamentais para o controle e estabilidade da aeronave em voo. O rotor principal, localizado na parte superior do helicóptero, é responsável por gerar sustentação e permitir o controle nos eixos lateral (Y) e longitudinal (X). Por meio da variação cíclica das pás, o rotor principal induz movimentos de arfagem (inclinação para frente e para trás) em torno do eixo Y, e rolamento (inclinação lateral) em torno do eixo X, possibilitando deslocamentos e manobras em múltiplas direções (Quanser, 2011; González; Rivera; Gómez, 2012).

Por sua vez, o rotor de cauda que é montado na extremidade traseira da fuselagem, atua na rotação em torno do eixo vertical (Z). Sua principal função é compensar o torque gerado pela rotação do rotor principal, ao evitar que a fuselagem gire no sentido oposto. Assim, ao controlar a potência do rotor de cauda, o piloto realiza a guinada, isto é, o movimento de rotação do helicóptero em torno do eixo Z, que permite a orientação do nariz da aeronave para a esquerda ou para a direita. Logo, esses dois rotores geram as dinâmicas tridimensionais que viabilizam o voo controlado e estável do helicóptero (Oh; Green, 2004; Patel et al., 2017).

Ademais, é notório destacar que há limitações operacionais e físicas relacionadas à produção dos ângulos de arfagem e guinada (no protótipo), onde o ângulo de arfagem pode ter uma variação máxima de  $\pm 65^\circ$  e para o ângulo de guinada uma variação máxima de  $\pm 180^\circ$  (Aras; Kaynak, 2014). Dessa forma, o estudo para produzir um modelo linearizado MIMO é feito com base na análise da aquisição de dados obtidos das entradas e saídas do H2-DOF, em malha aberta, com o objetivo de considerar as grandezas e os limites operacionais.

### 2.1.1 Diagrama de Corpo Livre (DCL)

No contexto de sistemas dinâmicos, como o helicóptero com dois graus de liberdade, o DCL é particularmente útil para simplificar os principais elementos que influenciam os movimentos de arfagem e guinada. Ao representar o helicóptero como um corpo isolado, é possível desenhar as forças de sustentação geradas pelos rotores, o peso da estrutura, os torques de acionamento e as resistências aerodinâmicas, bem

como as direções dos eixos longitudinal (X), lateral (Y) e vertical (Z).

Dessa forma, a representação na Figura 4 pode facilitar a formulação das equações de movimento utilizando as Leis de Newton ou o formalismo de Lagrange, contribuindo para a modelagem matemática do sistema. Com base nisso, o engenheiro pode projetar sistemas de controle mais precisos, antecipando como o processo irá reagir a diferentes entradas e perturbações (Patel et al., 2017; CRUZ et al., 2019; Aras; Kaynak, 2014).

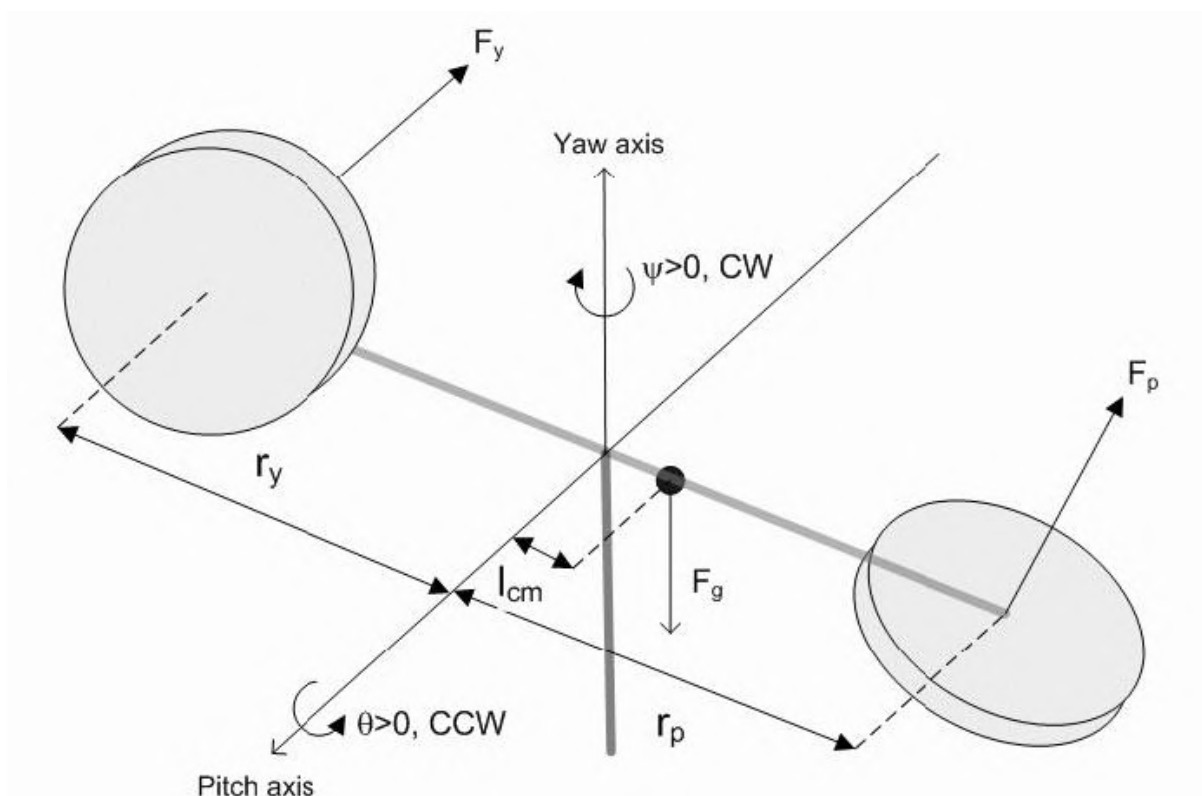


Figura 4 – Diagrama Simplificado de Corpo Livre do Helicóptero 2-DOF.

Fonte: Aras e Kaynak (2014).

A partir da observação das variáveis contidas na Figura 4 pode-se inferir as seguintes informações listadas a seguir:

- O ângulo de arfagem  $\theta$  e o ângulo de guinada  $\psi$  representam os dois graus de liberdade presentes no processo real.
- Quando o nariz do helicóptero sobe ao fazer um movimento anti-horário (CCW), em relação ao eixo lateral (Y), obtém-se  $\theta > 0$ . Enquanto isso, o ângulo de guinada torna-se positivo ( $\psi > 0$ ) quando a rotação do helicóptero é no sentido horário (CW) em torno do eixo vertical (Z).
- As forças  $F_p$  e  $F_y$  são aplicadas nos eixos de arfagem e guinada, respectivamente, pelo acionamento dos rotores.

- Os torques atuam a uma distância  $r_p$  e  $r_y$  do respectivo eixo, enquanto a força gravitacional  $F_g$  puxa para baixo o nariz do helicóptero.
- O centro de massa é localizado a uma certa distância do eixo de inclinação ao longo do comprimento do corpo do helicóptero, logo  $L_{cm}$  é a distância entre o centro de massa e a intersecção dos eixos de arfagem e guinada.

## 2.2 IDENTIFICAÇÃO POR MÍNIMOS QUADRADOS RECURSIVO

### 2.2.1 Aproximação em Espaço de Estados

De acordo com Silveira et al. (2020) e Dutra, Silveira e Pereira (2021) o algoritmo de estimação apresentado nesta sub-seção advém da necessidade de identificar sistemas combinados para fornecer um modelo em que o controlador interaja de forma direta e centralizada com as variáveis de estado, como os controladores lineares quadráticos. Entretanto, também é possível trabalhar com o modelo identificado de forma descentralizada ao aplicar metodologias de desacoplamento de malha, de forma a considerar os estados e as suas influências, que são descritos em capítulos posteriores.

Assim, ao considerar que todas as variáveis de estado são medidas ou estimadas, o estimador paramétrico dos mínimos quadrados recursivos pode ser usado para identificar os parâmetros do sistema da seguinte forma:

$$\underbrace{\begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix}}_{\mathbf{x}(k)} = \underbrace{\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix}}_A \underbrace{\begin{bmatrix} x_1(k-1) \\ \vdots \\ x_n(k-1) \end{bmatrix}}_{\mathbf{x}(k-1)} + \underbrace{\begin{bmatrix} b_{11} & \cdots & b_{1l} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nl} \end{bmatrix}}_B \underbrace{\begin{bmatrix} u_1(k-1) \\ \vdots \\ u_l(k-1) \end{bmatrix}}_{\mathbf{u}(k-1)} \quad (1)$$

$$+ \underbrace{\begin{bmatrix} \gamma_{11} & \cdots & \gamma_{1n} \\ \vdots & \ddots & \vdots \\ \gamma_{n1} & \cdots & \gamma_{nn} \end{bmatrix}}_\Gamma \underbrace{\begin{bmatrix} w_1(k-1) \\ \vdots \\ w_n(k-1) \end{bmatrix}}_{\mathbf{w}(k-1)}$$

$$\underbrace{\begin{bmatrix} y_1(k) \\ \vdots \\ y_j(k) \end{bmatrix}}_{\mathbf{y}(k)} = \underbrace{\begin{bmatrix} I_{j \times j} & 0_{j \times (n-j)} \end{bmatrix}}_C \underbrace{\begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix}}_{\mathbf{x}(k)} + \underbrace{\begin{bmatrix} v_1(k) \\ \vdots \\ v_j(k) \end{bmatrix}}_{\mathbf{v}(k)} \quad (2)$$

onde  $\mathbf{x}(k)$  é o vetor de estado,  $\mathbf{u}(k)$  é a entrada do sistema,  $\mathbf{y}(k)$  é a saída do sistema.  $A$  é a matriz de estado,  $B$  é a matriz de entrada,  $\Gamma$  é a matriz relacionada à entrada do vetor de ruído Gaussiano  $\mathbf{w}(k)$ ,  $C$  é a matriz que associa os estados à saída medida do processo e  $\mathbf{v}(k)$  é o ruído de medição.

A solução para a estimação dos parâmetros do modelo em espaço de estados, usando os mínimos quadrados recursivos, reside na determinação dos vetores de

parâmetros estimados (3) (Silveira et al., 2020; SILVA et al., 2023).

$$\hat{\theta} = \begin{bmatrix} \hat{\theta}_1^T \\ \vdots \\ \hat{\theta}_n^T \end{bmatrix} = \begin{bmatrix} \hat{a}_{11} & \cdots & \hat{a}_{1n} & \hat{b}_{11} & \cdots & \hat{b}_{1l} & \hat{y}_{11} & \cdots & \hat{y}_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \hat{a}_{n1} & \cdots & \hat{a}_{nn} & \hat{b}_{n1} & \cdots & \hat{b}_{nl} & \hat{y}_{n1} & \cdots & \hat{y}_{nn} \end{bmatrix}^T \quad (3)$$

De acordo com Dutra, Silveira e Pereira (2021), supondo que a saída medida  $y_1(k)$  corresponda à variável de estado  $x_1(k)$ , a estimativa do próximo estado,  $x_2(k)$ , pode ser calculada pela metodologia *backward difference* de  $x_1(k)$ , e assim por diante. Portanto, para uma melhor estimação de um modelo de ordem superior, as outras variáveis virtuais de estado podem ser calculadas pela aproximação da derivada discreta, em relação à variável real, em função do tempo de amostragem  $T_s$  em (4).

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{bmatrix} = \begin{bmatrix} y(k) \\ \frac{x_1(k) - x_1(k-1)}{T_s} \\ \vdots \\ \frac{x_{n-1}(k) - x_{n-1}(k-1)}{T_s} \end{bmatrix} \quad (4)$$

Por sua vez, as observações futuras são baseadas no seguinte vetor de regressores:

$$\varphi_{ls}^T(k) = \left[ \mathbf{x}(k-1) \right]_{1 \times n}^T \quad \left[ \mathbf{u}(k-1) \right]_{1 \times l}^T \quad \left[ \mathbf{w}(k-1) \right]_{1 \times n}^T \quad (5)$$

Os vetores de regressores, de um total de  $N$  amostras, são organizados para formar uma matriz de regressores, retratado como:

$$\Phi = \begin{bmatrix} \varphi_{ls}^T(k) \\ \vdots \\ \varphi_{ls}^T(N) \end{bmatrix}_{N \times (2n+l)} \quad (6)$$

Assim, o algoritmo permite uma identificação recursiva de  $\hat{\theta}$  em (3), a cada instante  $k$  com os dados, onde os estados estimados podem ser calculados em (7).

$$\hat{x}(k) = \Phi^T(k) \hat{\theta}(k-1) \quad (7)$$

A saída do modelo estimado e o erro de predição são calculados em (8) e (9).

$$\hat{y}(k) = C \hat{x}(k) \quad (8)$$

$$e(k) = x(k) - \hat{x}(k) \quad (9)$$

O ganho do estimador em (10) permite que as matrizes de parâmetros sejam atualizadas por (11).

$$\mathbf{K}(k) = \frac{P(k-1)\varphi(k)}{1 + \varphi^T(k)P(k-1)\varphi(k)} \quad (10)$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \mathbf{K}(k)e(k) \quad (11)$$

Sabe-se que  $w(k)$  é calculado a cada iteração de  $k$ , pelo algoritmo MQR, com base no erro de estimação dos estados e considerando inicialmente como um vetor com valores nulos, pode-se obter essa estimação em (12). Ademais, é aplicada a mesma lógica para o ruído de medição  $v(k)$  com base no erro entre a saída real e a saída estimada em (13).

$$w(k) = x(k) - \varphi^T(k)\hat{\theta}(k) \quad (12)$$

$$v(k) = y(k) - \hat{y}(k) \quad (13)$$

Portanto, a matriz de covariância do estimador é atualizada por (14) e o ciclo do algoritmo é repetido de forma sucessiva a cada incremento de  $k$ , até o final do experimento.

$$P(k) = \left[ I - \mathbf{K}(k)\varphi^T(k) \right] P(k-1) \quad (14)$$

### 2.2.1.1 Identificação aplicada ao sistema H2-DOF

Ao considerar as dinâmicas descritas na seção 2.1 e na subseção 2.1.1, sobre o processo real, é considerado o vetor de estados  $\mathbf{x}^T$  da seguinte forma:

$$\mathbf{x}^T(k) = \left[ \psi(k) \quad \theta(k) \quad \psi_{spd}(k) \quad \theta_{spd}(k) \right] \quad (15)$$

onde,  $\psi_{spd}$  é a derivada discreta do ângulo de guinada e  $\theta_{spd}$  é a derivada discreta do ângulo de arfagem.

Desta maneira, é possível verificar que apesar do H2-DOF possuir leituras de apenas dois estados, o uso das derivadas discretas em (4) são consideradas no projeto para tentar se obter uma melhor estimação da dinâmica do protótipo, mesmo que o protótipo não tenha sensores de medição analógica para estes estados. Ademais, com as observações produzidas na Figura 4, é considerado que as tensões aplicadas nos rotores do processo compõem o vetor de entrada do sistema em (16).

$$\mathbf{u}^T(k) = \left[ u_l(k) \quad u_v(k) \right] \quad (16)$$

onde, todas as entradas possuem uma faixa de operação  $[0,5]$ , sendo  $u_l(k)$  a tensão aplicada ao rotor de impulso lateral e  $u_v(k)$  a tensão aplicada ao rotor de impulso vertical.

A partir das equações (15) e (16), é feita a substituição em (5) para produzir o vetor de regressores inicial, que será estimado pelo algoritmo da seguinte forma:

$$\varphi_{ls}^T(k) = \begin{bmatrix} \psi(k) & \theta(k) & \psi_{spd}(k) & \theta_{spd}(k) & u_I(k) & u_V(k) & w_\psi(k) & w_\theta(k) \\ w_{\psi_{spd}}(k) & w_{\theta_{spd}}(k) \end{bmatrix} \quad (17)$$

Estas as etapas do algoritmo de MQR são reproduzidas até o fim do experimento com a convergência dos parâmetros e a atualização da matriz  $\hat{\theta}$ , que podem ser verificadas com detalhes no fluxograma da Figura 5.

Portanto, ao fim do fluxograma da Figura 5 é possível extrair a estrutura generalizada que representará o modelo do H2-DOF, pelas considerações feitas nas equações (15), (16) e (17), que pode ser descrito como:

$$\underbrace{\begin{bmatrix} \hat{x}_1(k) \\ \hat{x}_2(k) \\ \hat{x}_3(k) \\ \hat{x}_4(k) \end{bmatrix}}_{\hat{\mathbf{x}}(k)} = \underbrace{\begin{bmatrix} \hat{a}_{11} & \hat{a}_{12} & \hat{a}_{13} & \hat{a}_{14} \\ \hat{a}_{21} & \hat{a}_{22} & \hat{a}_{23} & \hat{a}_{24} \\ \hat{a}_{31} & \hat{a}_{32} & \hat{a}_{33} & \hat{a}_{34} \\ \hat{a}_{41} & \hat{a}_{42} & \hat{a}_{43} & \hat{a}_{44} \end{bmatrix}}_{\hat{A}} \underbrace{\begin{bmatrix} \hat{x}_1(k-1) \\ \hat{x}_2(k-1) \\ \hat{x}_3(k-1) \\ \hat{x}_4(k-1) \end{bmatrix}}_{\hat{\mathbf{x}}(k-1)} + \underbrace{\begin{bmatrix} \hat{b}_{11} & \hat{b}_{12} \\ \hat{b}_{21} & \hat{b}_{22} \\ \hat{b}_{31} & \hat{b}_{32} \\ \hat{b}_{41} & \hat{b}_{42} \end{bmatrix}}_{\hat{B}} \underbrace{\begin{bmatrix} u_1(k-1) \\ u_2(k-1) \end{bmatrix}}_{\mathbf{u}(k-1)} \quad (18)$$

$$+ \underbrace{\begin{bmatrix} \hat{\gamma}_{11} & \hat{\gamma}_{12} & \hat{\gamma}_{13} & \hat{\gamma}_{14} \\ \hat{\gamma}_{21} & \hat{\gamma}_{22} & \hat{\gamma}_{23} & \hat{\gamma}_{24} \\ \hat{\gamma}_{31} & \hat{\gamma}_{32} & \hat{\gamma}_{33} & \hat{\gamma}_{34} \\ \hat{\gamma}_{41} & \hat{\gamma}_{42} & \hat{\gamma}_{43} & \hat{\gamma}_{44} \end{bmatrix}}_{\hat{\Gamma}} \underbrace{\begin{bmatrix} \hat{w}_1(k-1) \\ \hat{w}_2(k-1) \\ \hat{w}_3(k-1) \\ \hat{w}_4(k-1) \end{bmatrix}}_{\hat{\mathbf{w}}(k-1)}$$

$$\underbrace{\begin{bmatrix} \hat{y}_1(k) \\ \hat{y}_2(k) \end{bmatrix}}_{\hat{\mathbf{y}}(k)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}}_{\hat{C}} \underbrace{\begin{bmatrix} \hat{x}_1(k) \\ \hat{x}_2(k) \\ \hat{x}_3(k) \\ \hat{x}_4(k) \end{bmatrix}}_{\hat{\mathbf{x}}(k)} + \underbrace{\begin{bmatrix} \hat{v}_1(k) \\ \hat{v}_2(k) \end{bmatrix}}_{\hat{\mathbf{v}}(k)} \quad (19)$$

### 2.3 ALGORITMO DE SELEÇÃO DO MELHOR MODELO ESTIMADO PELO MQR

A presente pesquisa propõe um procedimento computacional baseado na formulação de modelos no espaço de estados para a simulação e avaliação dos diversos modelos previamente estimados a partir de dados experimentais pelo estimador da seção 2.2. A estrutura computacional, desenvolvida no ambiente MATLAB, é concebida com o intuito de selecionar e validar, dentre os modelos estimados disponíveis gerados a cada iteração  $k$  pelo vetor  $\hat{\theta}$  do algoritmo MQR, aquele que melhor representa a

dinâmica do sistema real, a partir da comparação entre as saídas estimadas ( $\hat{y}$ ) e as saídas medidas ( $y$ ), com a minimização da função custo em (21) relacionada ao erro médio quadrático do algoritmo MQR (20) (Aström; Wittenmark, 2008).

$$J_m = \frac{1}{2} \left( \sum_{k=1}^N (y_1(k) - \hat{y}_{1,m}(k))^2 + (y_2(k) - \hat{y}_{2,m}(k))^2 \right) \quad (20)$$

$$J_{MSE} = \min [J_1 \quad J_2 \quad J_3 \quad J_4 \dots J_N] \quad (21)$$

onde,  $k$  é a posição de cada elemento do vetor das saídas e  $m$  é o respectivo modelo estimado pelo algoritmo MQR em (3).

Os modelos utilizados estão organizados na forma matricial, em que os vetores  $A_{est}$ ,  $B_{est}$  e  $\Gamma_{est}$  armazenam os parâmetros estimados ( $\hat{A}$ ,  $\hat{B}$  e  $\hat{\Gamma}$ ) de cada modelo, empilhados verticalmente em blocos de quatro linhas, correspondentes à dimensão do espaço de estados do sistema considerado. A matriz de saída  $C$  é mantida fixa, visto que o mapeamento dos estados para as saídas permanece inalterado em todos os modelos.

Cada modelo é simulado utilizando a formulação discreta clássica do espaço de estados, conforme apresentada na equação abaixo:

$$\begin{cases} \hat{x}(k) = \hat{A}\hat{x}(k-1) + \hat{B}u(k-1) + \hat{\Gamma}\hat{w}(k-1) \\ \hat{y}(k) = C\hat{x}(k) + \hat{v}(k) \end{cases} \quad (22)$$

em que, para o processo H2-DOF:

- $\hat{x}(k) \in \mathbb{R}^4$  representa o vetor de estados no instante de tempo  $k$ .
- $u(k) \in \mathbb{R}^2$  é o vetor de entradas reais conhecidas aplicadas ao sistema estimado.
- $\hat{y}(k) \in \mathbb{R}^2$  corresponde ao vetor de saídas estimadas.

A cada iteração do laço principal de simulação, são extraídas as submatrizes associadas a um modelo estimado, utilizando uma linguagem similar ao MATLAB e a lógica de endereçamento por blocos, definindo que os “dois pontos” generalizam o acesso as colunas das matrizes. Considerando um total de  $n_{blocos}$  modelos disponíveis, cada modelo  $m$  é acessado através do índice:

$$i_m = (n_{blocos} - 1)4 + 1 : n_{blocos} \cdot 4 \quad (23)$$

Desta forma, gerado o índice em (23), é possível acessar o modelo que é extraído para as submatrizes:

$$\begin{aligned} A_{aux} &= A_{est}(i_m, :) \\ B_{aux} &= B_{est}(i_m, :) \\ \Gamma_{aux} &= \Gamma_{est}(i_m, :) \end{aligned} \quad (24)$$

Para os parâmetros do modelo estimado extraídos em (24), realiza-se a simulação da trajetória dos estados e das saídas estimadas do modelo para o vetor de entrada  $u$  do processo real, que foi usado no estimador MQR no vetor de regressores em (16). As saídas simuladas de todos os modelos são armazenadas em uma matriz denominada  $y_{sim}$ , contendo os pares de saídas estimadas ( $\hat{y}_{1m}$  e  $\hat{y}_{2m}$ ) para cada instante de tempo  $k$ .

Após a simulação de todos os modelos, é realizada a etapa de avaliação, onde é usada a equação (20) e o processo é repetido para todos os modelos  $m$ , gerando um vetor de erros  $J_{MSE}$ , que armazena o  $J_m$  de cada simulação. O modelo que apresentar o menor valor verificado no vetor em (21) é selecionado como o modelo mais adequado para representar a dinâmica dos dados do sistema real.

O melhor modelo escolhido, representado pelas matrizes  $A_{best}$ ,  $B_{best}$  e  $\Gamma_{best}$ , é utilizado em uma nova etapa de validação, com um conjunto de dados não utilizado anteriormente. O objetivo desta fase é verificar a capacidade de generalização do melhor modelo escolhido, comparando as saídas simuladas com os novos dados reais por meio de gráficos com uma medida estatística adicional, para verificar o grau de correlação linear entre as saídas estimadas e reais, para ambas as variáveis de saída do sistema, denominada de Coeficiente de Correlação Múltipla ( $R^2$ ) em (25). Segundo Coelho e Coelho (2004), para muitas aplicações práticas, valores de  $R^2$  entre 0,8 e 1,0 podem ser considerados suficientes para o processo estimado pela identificação, enquanto um valor igual a 1 indica uma exata adequação do modelo para os dados medidos do processo.

$$R^2 = 1 - \frac{\sum_{k=1}^N [y(k) - \hat{y}_{best}(k)]^2}{\sum_{k=1}^N [y(k) - \bar{\hat{y}}_{best}]^2} \quad (25)$$

onde,  $\bar{\hat{y}}_{best}$  é a média do vetor de saídas do melhor modelo escolhido ( $\hat{y}_{best}$ ).

Por fim, um pseudocódigo foi expresso no Algoritmo 1, para complementar a lógica de programação utilizada para realizar esse procedimento e auxiliar futuras pesquisas de identificação com o estimador MQR, em espaço de estados, para sistemas multivariáveis e produzir possíveis melhorias para otimizar o algoritmo de seleção.

**Algoritmo 1** Simulação, Avaliação e Validação de Modelos Estimados

---

```

1:  $nBlocos \leftarrow$  número de linhas de  $A_{est} \div 4$ 
2: Inicializar  $ysim \leftarrow []$  ▷ para armazenar todas as saídas simuladas
3: for  $bloco = 1$  até  $nBlocos$  do
4:   Limpar variáveis  $x, y, u$ 
5:    $idx \leftarrow (bloco-1) \cdot 4 + 1 : bloco \cdot 4$ 
6:   Extrair  $A_{aux}, B_{aux}, \Gamma_{aux}$  a partir de  $A_{est}, B_{est}, \Gamma_{est}$ 
7:   Extrair ruídos  $w_{aux} \leftarrow w(:, :, bloco)$  e  $v_{aux} \leftarrow v(:, :, bloco)$ 
8:    $x(:, :, 1) \leftarrow [0 \ 0 \ 0 \ 0]^T$ 
9:    $y(:, :, 1) \leftarrow C \cdot x(:, :, 1)$ 
10:   $u(:, :, 1) \leftarrow [u_1(1); u_2(1)]$ 
11:  for  $i = 2$  até  $N$  do
12:     $u(:, :, i) \leftarrow [u_1(i); u_2(i)]$ 
13:     $x(:, :, i) \leftarrow A_{aux} \cdot x(:, :, i-1) + B_{aux} \cdot u(:, :, i-1) + \Gamma_{aux} \cdot w(:, :, i-1)$ 
14:     $y(:, :, i) \leftarrow C \cdot x(:, :, i) + v(:, :, i)$ 
15:  end for
16:  Empilhar  $y$  em  $ysim$ 
17: end for
18: Inicializar vetor de erros  $EEST \leftarrow []$ 
19: for  $m = 1$  até  $nBlocos$  do
20:    $ysim_1 \leftarrow ysim(2m-1, :)$  ▷ Saída 1 do modelo  $m$ 
21:    $ysim_2 \leftarrow ysim(2m, :)$  ▷ Saída 2 do modelo  $m$ 
22:    $J_m \leftarrow \frac{1}{2} \sum [(y_1 - ysim_1)^2 + (y_2 - ysim_2)^2]$ 
23:   Armazenar  $EEST \leftarrow [EEST; J_m]$ 
24: end for
25:  $best\_ind \leftarrow \arg \min(EEST)$ 
26:  $y_{best,1} \leftarrow ysim(2 \cdot best\_ind - 1, :)$ 
27:  $y_{best,2} \leftarrow ysim(2 \cdot best\_ind, :)$ 
28: Calcular  $J_{m_1} \leftarrow \frac{1}{2} \sum (y_1 - y_{best,1})^2$ 
29: Calcular  $J_{m_2} \leftarrow \frac{1}{2} \sum (y_2 - y_{best,2})^2$ 
30: Calcular  $J_{MSE} \leftarrow \min(EEST)$ 
31:  $idx_{best} \leftarrow (best\_ind - 1) \cdot 4 + 1 : best\_ind \cdot 4$ 
32: Extrair blocos  $A_{best}, B_{best}, \Gamma_{best}$  de acordo com  $idx_{best}$ 
33: Validação com nova base de dados
34: Carregar sinais de entrada  $u_1, u_2$  e saídas reais  $y_1, y_2$ 
35: Normalizar sinais removendo suas médias
36: Inicializar  $x(:, :, 1) \leftarrow [0 \ 0 \ 0 \ 0]^T, u(:, :, 1)$  e  $y(:, :, 1)$ 
37: for  $k = 2$  até  $N_{val}$  do
38:    $u(:, :, k) \leftarrow [u_1(k); u_2(k)]$ 
39:    $x(:, :, k) \leftarrow A_{best} \cdot x(:, :, k-1) + B_{best} \cdot u(:, :, k-1) + \Gamma_{best} \cdot w(:, :, i-1)$ 
40:    $y(:, :, k) \leftarrow C \cdot x(:, :, k) + v(:, :, k)$ 
41: end for
42: Plotar saídas simuladas e reais
43: Calcular Coeficiente de Correlação Múltipla  $R^2$  entre saídas as simuladas e as reais:
44: Calcular  $R_1^2$  e  $R_2^2$  referente a cada saída, e gerar  $R_{médio}^2 \leftarrow (R_1^2 + R_2^2)/2$ 

```

---

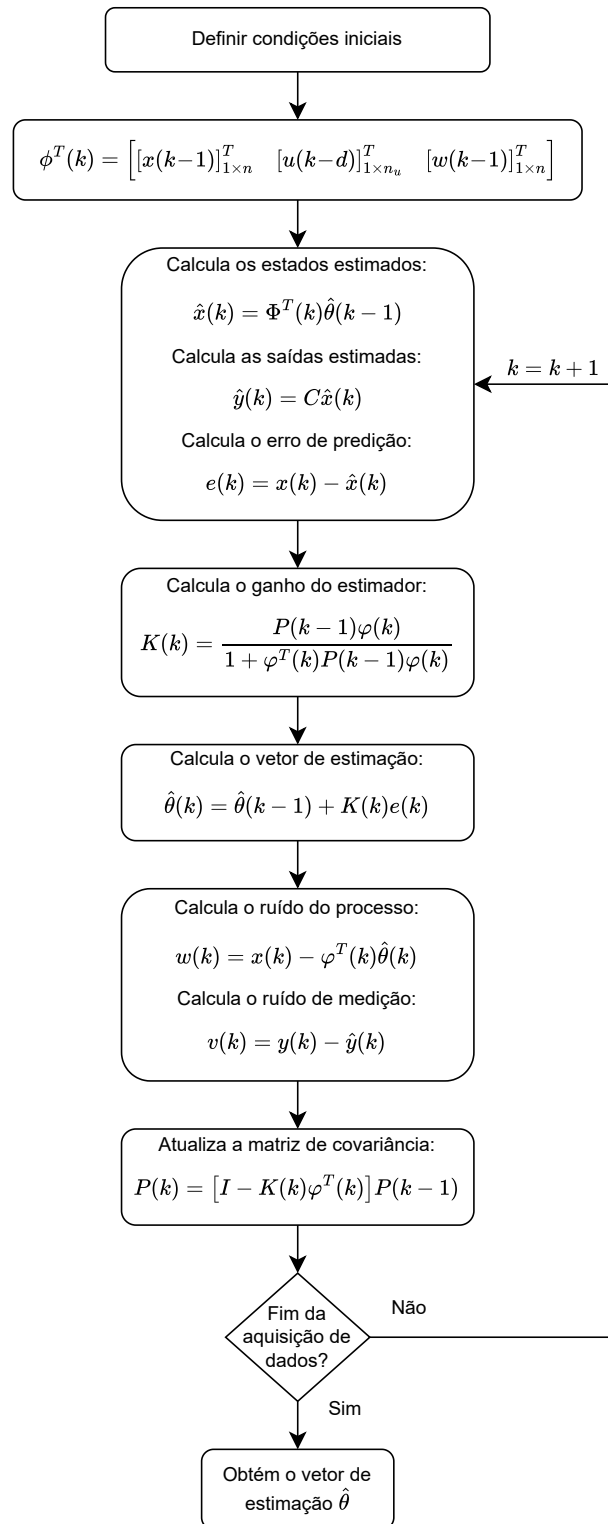


Figura 5 – Fluxograma do algoritmo MQR para obter o vetor de estimação  $\hat{\theta}$ .

Fonte: Dutra, Silveira e Pereira (2021) adaptado.

### 3 TEORIA DE CONTROLE

A teoria de controle abrange uma ampla gama de estratégias e topologias que podem ser formuladas tanto no domínio do tempo contínuo quanto no domínio do tempo discreto. O domínio contínuo da frequência, representado pela variável  $s$ , é tradicionalmente associado à transformada de Laplace e à análise de sistemas que operam com sinais contínuos no tempo. Por outro lado, o domínio discreto, representado pela variável  $z$ , está relacionado à transformada  $z$  e é essencial na modelagem e projeto de sistemas digitais, os quais operam com sinais amostrados e implementações computacionais.

O estudo e a análise de sistemas nos domínios  $s$  e  $z$  são amplamente explorados por autores como Oppenheim (1999), onde enfatizam a importância dessas ferramentas para o entendimento do comportamento dinâmico de sistemas lineares e invariantes no tempo (LTI), seja em tempo contínuo ou discreto. Ademais, o projeto de controladores digitais no domínio  $z$  baseia-se em princípios semelhantes aos utilizados no domínio  $s$  para sistemas de tempo contínuo, mas com suas devidas modificações ao levar em conta a natureza de tempo discreto do sistema. Portanto, a ideia concentra-se em ajustar os zeros e polos do controlador, que antes eram representados em um semiplano infinito, agora em um plano de círculo unitário, ambos detalhados na Figura 6.

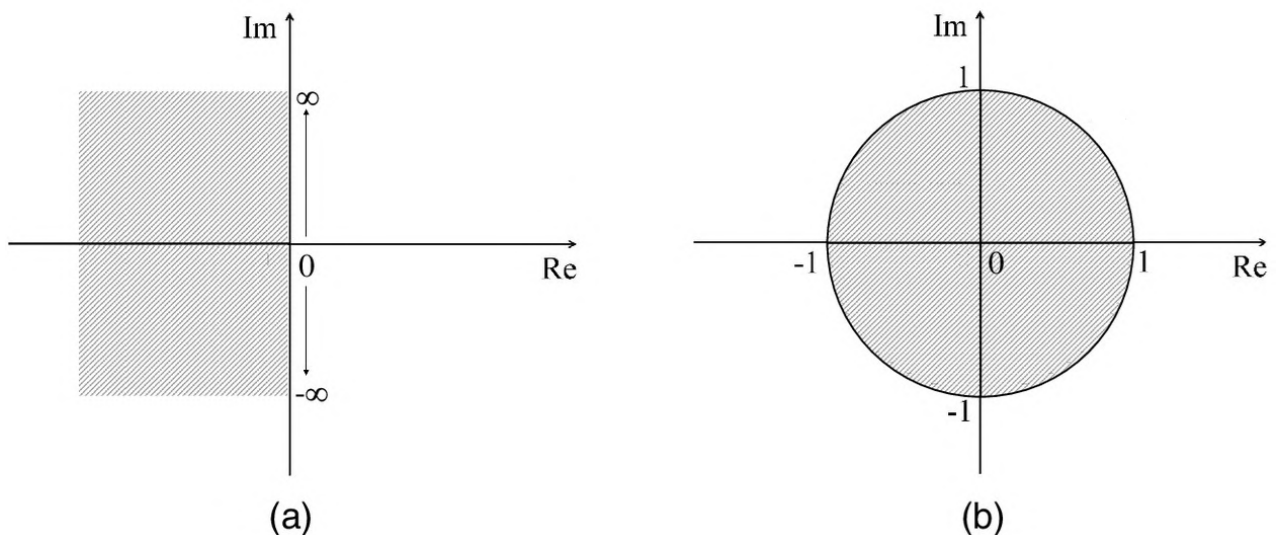


Figura 6 – Esquema do mapeamento das regiões de estabilidade do sistema contínuo para o sistema discreto: (a) domínio  $s$ ; e (b) domínio  $z$ .

Fonte: Tang et al. (2021).

Desta maneira, esta área do conhecimento representa um campo fundamental dentro da Engenharia de Controle e Automação, especialmente em cenários de aplicações industriais, aeroespaciais e robóticas. Essa teoria engloba desde a análise até a produção do projeto em sistemas cujos sinais são manipulados por dispositivos digitais,

como microcontroladores, DSPs (Digital Signal Processors) ou sistemas embarcados em geral (Åström; Wittenmark, 2013; Seborg; Edgar; Mellichamp, 2003). Entretanto, apesar dos estudos de sistemas contínuos e digitais terem a possibilidade de serem feitos de maneira independente, uma abordagem inicial que relacione ambos os domínios é extremamente valiosa. Isso se deve à necessidade de compreender como os sinais contínuos são convertidos em sinais discretos e como esse processo impacta o comportamento do sistema de controle (Åström; Wittenmark, 2013).

Assim, segundo Landau e Zito (2006) e Åström e Wittenmark (2013), um sistema de controle digital trabalha a partir da ideia de instantes discretos, onde é possível, por meio de um computador, produzir uma perspectiva de uma variável do mundo real com base na periodicidade do *clock* de processamento do algoritmo, conforme o diagrama simplificado na Figura 7.

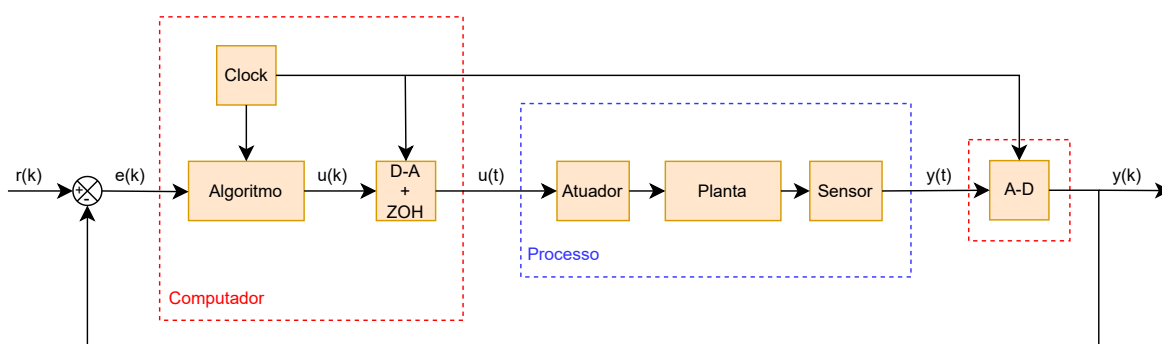


Figura 7 – Diagrama de blocos de um sistema de controle.

Fonte: Autor

Desta forma, na Figura 7 pode-se observar que o processo a ser controlado é um sistema no tempo contínuo formado pelo conjunto de atuador, planta e sensor. Destarte, a saída medida em tempo contínuo  $y(t)$  é convertida em uma sequência de valores numéricos  $y(k)$  por um conversor A-D (análogo-digital), em instantes de amostragem  $k$ , definidos pelo relógio de sincronização (*clock*). Logo, essa sequência é comparada com a sequência de referência  $y_r(k)$ , que resulta nos valores de  $e(k)$ .

Posteriormente, com base nos valores discretos de  $e(k)$  que são processados pelo computador, o algoritmo gerará a sequência de sinais de controle  $u(k)$  correspondentes. Então, por meio de um conversor D-A (digital-analógico), essa sequência é convertida em um sinal analógico, que geralmente é mantido constante entre os instantes de amostragem por um segurador de ordem zero (ZOH).

É válido destacar que o segurador de ordem zero pode ser descrito como um *buffer* presente nos conversores D-A, onde o sinal analógico produzido é segurado até o momento em que uma nova informação de geração seja solicitada. Além disso, com o avanço da tecnologia e a velocidade de resposta dos computadores, é possível desprezar os intervalos de tempo nas operações que vão da conversão A-D, proces-

samento do algoritmo até a conversão D-A. Entretanto, o intervalo de tempo em que o segurador permanece esperando a solicitação de um novo ciclo de interação  $k$  para a conversão D-A, é uma restrição característica dos projetos de controle digital e que difere da metodologia aplicada na análise contínua (Silveira, 2024; Landau; Zito, 2006).

Embora os projetos dos controladores discretos sejam responsáveis por grandes avanços, em termos de aumento de desempenho nos processos industriais, estes não podem ser considerados uma solução universal para todos os tipos de processos, como em sistemas de potência que necessitam ser projetados com tempos de amostragem muito pequenos. Então, se o tempo de amostragem  $T_s$  (descrito como o intervalo de tempo do segurador) for muito menor que o tempo do ciclo de trabalho do processador, o controlador pode não garantir robustez de forma satisfatória, e em casos mais graves pode causar instabilidade total no processo.

Na sequência deste capítulo, será abordado o projeto dos controladores para o sistema 2-DOF utilizando a estratégia de alocação de polos, onde primeiramente é definida a ferramenta matemática para a estruturação do projeto. Ademais, é apresentada a estrutura do sistema de controle a ser implementada com a utilização de artifícios como um pré-compensador digital de ganho direto e a matriz de ganhos relativos (RGA), que viabiliza a projeção de dois controladores SISO ao invés de um projeto de controlador MIMO.

### 3.1 CONTROLE PID

O controlador PID é a tecnologia dominante nas malhas de controle, onde problemas de controle simples podem ser resolvidos de maneira satisfatória, desde que os requisitos de desempenho não sejam muito elevados. Desta maneira, quando sistemas de malha fechada necessitam ser acrescidos de um bloco compensador, existem diversas implementações de compensadores que variam desde um simples filtro de 1ª ordem até complexos arranjos com equações de ordem maior. Dentre estes, o controlador PID possui três formas de atuação, cada uma com um efeito diferente ao sistema (Ogata; Souza; Tannuri, 2011; Seborg; Edgar; Mellichamp, 2003).

A combinação da quantidade de cada uma das ações do controlador PID faz com que o controlador, em conjunto com a planta, forneça uma resposta adequada a uma determinada variação na entrada. Dessa forma, a lei de controle do controlador PID é dada por:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (26)$$

onde,  $K_p$  é o ganho proporcional,  $K_i$  é o ganho integral,  $K_d$  é o ganho derivativo,  $u(t)$  é o sinal de controle e  $e(t)$  é o sinal de erro da malha de controle.

Dessa forma, para gerar o sinal de controle  $u(t)$ , a parcela proporcional do controlador ( $K_p$ ) calcula uma saída que é proporcional ao sinal de erro atual. Destarte, esse valor proporcional é produzido pela diferença entre a referência atual  $y_r(t)$  e o valor atual da variável de processo  $y(t)$ . A parcela integral do controlador objetiva uma ação que é proporcional ao sinal de erro cumulativo, onde sua saída é proporcional à soma dos sinais de erro atuais e anteriores. Por fim, o componente derivativo do controlador calcula uma ação que é proporcional à taxa de variação do sinal de erro  $e(t)$ , em que a saída produz uma ação proporcional à diferença entre os sinais de erro atuais e passados.

No entanto, para a implementação da malha de controle em um sistema real, por meio de computadores e com maior eficiência, é necessária a projeção de um controlador PID digital. Para o projeto do controlador PID digital, deve-se ter uma correta escolha do período de amostragem,  $T_s$ , e dos coeficientes  $K_p$ ,  $K_i$  e  $K_d$  (Aguirre, 2007; Coelho; Jeronymo; Araujo, 2019).

O ajuste dos parâmetros do controlador PID pode ser um dos principais desafios para se estabelecer um controle satisfatório para o processo. Por isso, quando o projetista não possui muito conhecimento sobre o comportamento dinâmico do sistema a ser controlado, embora haja apenas três parâmetros, o controle deve satisfazer critérios complexos dentro dos limites do controlador para se alcançar um desempenho de controle desejado, de acordo com as especificações do projeto.

Com o objetivo de se estabelecer uma relação de conversão entre o controlador PID contínuo e o controlador PID digital, pode-se considerar a transformada de Laplace aplicada em (26) para obter a lei de controle PID no domínio da frequência em (27).

$$U(s) = K_p E(s) + \frac{K_i}{s} E(s) + K_d s E(s) \quad (27)$$

Posteriormente, com base em Landau e Zito (2006) e Silveira (2024), o domínio discreto permite múltiplas representações de um mesmo sistema contínuo, cujo elo entre contínuo e discreto é definido pelo período de amostragem e o método de discretização. Então, é possível aproximarmos a equação (27) pelo método *Backward Difference* a seguir:

$$s := \frac{1 - z^{-1}}{T_s} \quad (28)$$

Desta maneira, ao substituir (28) em (27) é possível gerar a síntese do controlador PID digital de forma mais simples e direta, obtendo-se

$$\frac{U(z)}{E(z)} = K_p + K_i \frac{T_s}{1 - z^{-1}} + K_d \frac{1 - z^{-1}}{T_s} \quad (29)$$

É notório verificar que a síntese do controlador PID digital está na topologia paralela em (29). Desta forma, essa topologia é considerada didaticamente como o

“padrão” na literatura de Controle e Sistemas, pois a partir dela são derivadas outras topologias (como I+PD, PI+D e outras), que objetivam solucionar problemas, a depender do tipo de processo que necessita ser controlado. Portanto, apesar desta estrutura possuir uma característica mais impulsiva, em relação a outras topologias (Ogata; Souza; Tannuri, 2011), sua estrutura de projeto é mais simples de ser implementada e pode ser representada pelo diagrama de blocos na Figura 8.

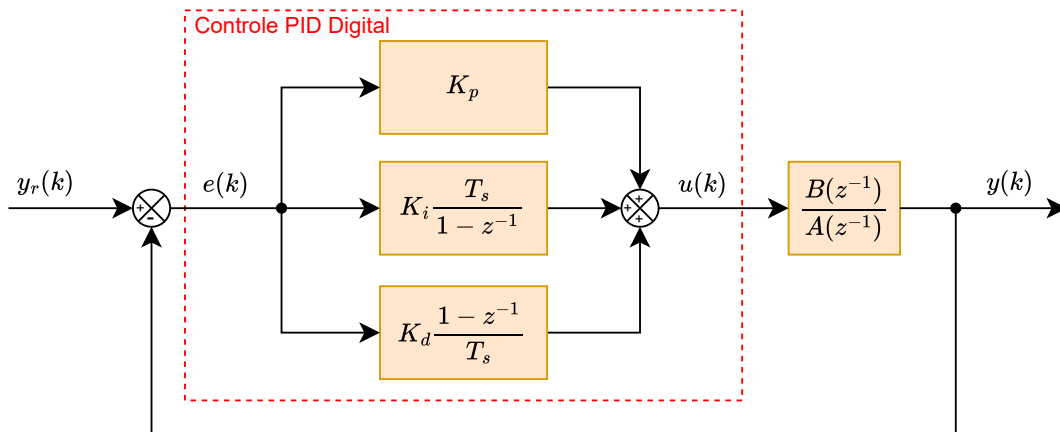


Figura 8 – Estrutura PID Paralelo Digital.

Fonte: Autor.

$y_r(k)$ ,  $e(k)$ ,  $B(z^{-1})$ ,  $A(z^{-1})$ ,  $u(k)$  e  $y(k)$  são, respectivamente, sinal de referência, erro, numerador da função de transferência discreta, denominador da função de transferência discreta, sinal de controle e sinal de saída.

A partir disso, ao observar a estrutura da Figura 8 e a Equação (29), é possível dar continuidade ao equacionamento do controlador para a sua forma filtrada, como segue:

$$\frac{U(z)}{E(z)} = \frac{K_p - K_p z^{-1} + K_i T_s + \left(\frac{K_d}{T_s}\right) (1 - 2z^{-1} + z^{-2})}{1 - z^{-1}} \quad (30)$$

$$\frac{U(z)}{E(z)} = \frac{\left(K_p + K_i T_s + \frac{K_d}{T_s}\right) + \left(-K_p - 2\frac{K_d}{T_s}\right) z^{-1} + \left(\frac{K_d}{T_s}\right) z^{-2}}{1 - z^{-1}} \quad (31)$$

Em paralelo a isso, se aplicarmos a transformada  $z$  inversa sobre a forma filtrada da equação (31) é possível expressar a equação a diferenças, no tempo discreto  $k$ , que é usada para a implementação desse controlador. Ademais, esta síntese é causal, devido à solução de  $u(k)$  depender integralmente das informações anteriores (de  $u(k - 1)$ ,  $e(k - 1)$ ,  $e(k - 2)$ ) e do estado atual do erro de seguimento de referência  $e(k)$ , como pode ser observado em:

$$u(k) = u(k - 1) + s_0 e(k) + s_1 e(k - 1) + s_2 e(k - 2) \quad (32)$$

$$\begin{cases} s_0 = K_p + K_i T_s + \frac{K_d}{T_s} \\ s_1 = -K_p - 2\frac{K_d}{T_s} \\ s_2 = \frac{K_d}{T_s} \end{cases} \quad (33)$$

onde,  $s_0$ ,  $s_0$  e  $s_0$ , são os termos associados ao projeto via alocação de polos pela estrutura RST, que são calculados para sintonizar os ganhos do controlador PID, denominados de "máscara do controlador".

Por fim, com o objetivo de encontrar valores adequados para os parâmetros do PID, é considerado o diagrama de blocos de um sistema discreto, na Figura 9.

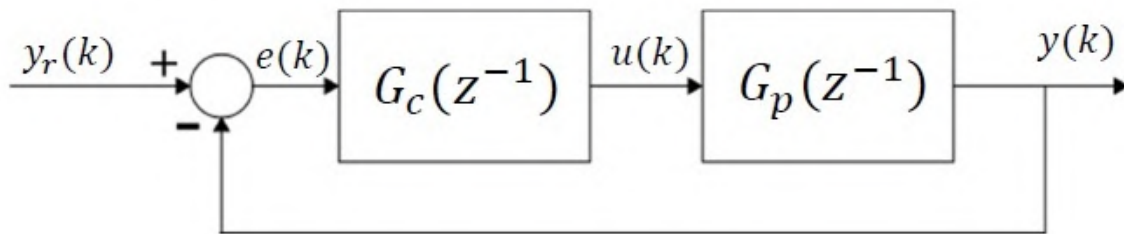


Figura 9 – Diagrama de blocos simplificado do sistema em malha fechada discreto.

Fonte: Araújo, Jeronimo e Coelho (2019).

Com base nisso, nota-se que a saída do sistema possui o rastreamento da referência,  $y_r(k)$ , e tem a forma:

$$y(k) = \frac{G_c(z^{-1})G_p(z^{-1})}{1 + G_c(z^{-1})G_p(z^{-1})} y_r(k) \quad (34)$$

onde,  $G_c(z^{-1})$  e  $G_p(z^{-1})$  representam as funções de transferência do controlador e da planta discretas, respectivamente.

Nota-se que o controlador depende da especificação da resposta em malha fechada em (34) para assegurar que a realização do controlador seja factível, ao calcular a sintonia dos ganhos, como apresentado na estrutura em (32) e (33).

Portanto, existem técnicas para síntese dos ganhos  $K_p$ ,  $K_i$  e  $K_d$ , que podem ser produzidas por métodos de ajuste manual, como sintonias por tabela (Ziegler-Nichols, Cohen-Coon, IMC) ou pela utilização de softwares para ajuste direto do PID (*Auto-tuning*). Logo, nesta pesquisa, a técnica escolhida para projetar e sintonizar os controladores clássicos discretos é feita com base na estrutura RST pelo método de Alocação de Polos (*Pole-Placement*) (Landau; Zito, 2006; Ogata; Souza; Tannuri, 2011).

### 3.2 PROJETO DO CONTROLADOR POR ALOCAÇÃO DE POLOS

O projeto do controlador por alocação de polos consiste na utilização de uma estrutura de controle bastante difundida na indústria, conhecida como RST, que se fundamenta na definição de três polinômios posicionados em diferentes pontos da malha de controle. Na Figura 10 mostra-se a estrutura do controlador na forma canônica discreta e, ao contrário do que ocorre em um controlador PID, por exemplo, onde a atuação do controlador é feita pelo sinal de erro da planta, o controlador RST possibilita que os sinais do valor real da variável controlada e o sinal de referência sejam tratados de forma independente (Landau; Zito, 2006; Araujo, M. et al., 2017).

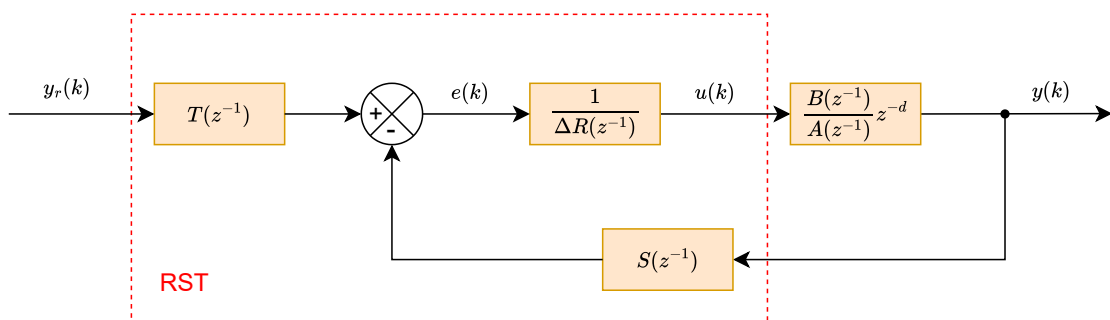


Figura 10 – Estrutura RST na Forma Canônica Discreta.

Fonte: Autor

Os três polinômios  $R(z^{-1})$ ,  $S(z^{-1})$  e  $T(z^{-1})$  atuam, respectivamente, na ação de controle, na realimentação da saída e no tratamento do sinal de referência.

A partir da Figura 10 é possível expressar a lei de controle do controlador RST, que pode ser descrita como:

$$\Delta R(z^{-1})u(k) = T(z^{-1})y_r(k) - S(z^{-1})y(k) \quad (35)$$

$$u(k) = \frac{T(z^{-1})}{\Delta R(z^{-1})}y_r(k) - \frac{S(z^{-1})}{\Delta R(z^{-1})}y(k) \quad (36)$$

Ademais,  $\Delta$  é um termo que representa o operador diferença discreta,  $\Delta = 1 - z^{-1}$ , que posicionado no denominador em (36) opera como um integrador discreto  $1/\Delta$ , que garante erro nulo para a saída em regime permanente e rejeição de perturbação de carga.

A forma presente na equação (36) permite que o projeto do controlador seja dividido em duas partes: a relação de realimentação da saída  $y(k)$  por meio dos polinômios  $R(z^{-1})$  e  $S(z^{-1})$ , e a relação do rastreamento da referência  $y_r(k)$  por meio do polinômio  $T(z^{-1})$  e  $R(z^{-1})$  (Franklin; Powell; Workman, 1997).

Os polinômios  $R(z^{-1})$ ,  $S(z^{-1})$  e  $T(z^{-1})$  são estruturados da seguinte forma:

$$\begin{aligned}
R(z^{-1}) &= 1 + r_1 z^{-1} + r_2 z^{-2} + \dots \\
S(z^{-1}) &= s_0 + s_1 z^{-1} + s_2 z^{-2} + \dots \\
T(z^{-1}) &= t_0 + t_1 z^{-1} + t_2 z^{-2} + \dots
\end{aligned} \tag{37}$$

A substituição da lei de controle (36) na equação da planta resulta na seguinte expressão para a saída  $y(k)$ :

$$y(k) = G_p(z^{-1}) \cdot u(k) = \frac{B(z^{-1})z^{-d}}{A(z^{-1})} \cdot \left( \frac{T(z^{-1})}{\Delta R(z^{-1})} y_r(k) - \frac{S(z^{-1})}{\Delta R(z^{-1})} y(k) \right) \tag{38}$$

Multiplicando ambos os lados por  $A(z^{-1})\Delta R(z^{-1})$  para eliminar os denominadores e reorganizando a equação (38), obtém-se:

$$A(z^{-1})\Delta R(z^{-1})y(k) + B(z^{-1})z^{-d}S(z^{-1})y(k) = B(z^{-1})z^{-d}T(z^{-1})y_r(k) \tag{39}$$

Assim, ao analisar a equação (39) é possível obter a função de transferência do sistema em malha fechada da (34), que também pode ser descrita como uma relação da saída  $y(k)$  sobre o sinal de referência  $y_r(k)$ , segundo a expressão:

$$G_{mf}(z^{-1}) = \frac{y(k)}{y_r(k)} = \frac{B(z^{-1})z^{-d}T(z^{-1})}{A(z^{-1})\Delta R(z^{-1}) + B(z^{-1})z^{-d}S(z^{-1})} \tag{40}$$

Desta forma, ao considerar as estruturas das Figuras 10 e 8, é possível observar uma pequena diferença, onde considera-se uma planta com atraso puro de  $d$  amostras. Esse atraso deve ser explicitamente considerado no projeto do controlador para garantir a estabilidade do sistema e satisfazer o equacionamento para o projeto.

Enfim, a condição considerada no projeto é de que o denominador da função de transferência em malha fechada seja igual a um polinômio pré-especificado  $H_{cl}(z^{-1})$ , cujas raízes determinam os polos desejados do sistema. Isso leva à chamada equação Diofantina:

$$A(z^{-1})\Delta R(z^{-1}) + B(z^{-1})z^{-d}S(z^{-1}) = H_{cl}(z^{-1}) \tag{41}$$

A resolução dessa equação permite determinar os coeficientes dos polinômios  $R(z^{-1})$  e  $S(z^{-1})$ . Já o polinômio  $T(z^{-1})$  é normalmente escolhido para garantir o rastreamento adequado da referência. Uma escolha comum é:

$$T(z^{-1}) = H_o(z^{-1}) \tag{42}$$

onde,  $H_o(z^{-1})$  é o polinômio associado ao comportamento em regime permanente, que muitas vezes é igual ao polinômio de observador ou a um fator de  $H_{cl}(z^{-1})$  (Franklin; Powell; Workman, 1997).

Portanto, esta estrutura pode ser trabalhada para representar controladores clássicos, com diferentes topologias, ao fazer a equivalência das parcelas da estrutura RST, como  $S(z^{-1}) = T(z^{-1})$ . Desta forma, o projeto deste controlador discreto utilizou o modelo em sua forma filtrada (31) para desenvolver a sintonia do controlador PI e PID por aproximação Backward, expresso em (32) e (33), para satisfazer o objetivo do projeto.

### 3.2.1 Considerações de Projeto para o Modelo do Sistema Multivariável

Antes de prosseguir para as considerações dos projetos dos controladores (RST-PID e RST-PI), é necessário descrever as aproximações produzidas com base no modelo estimado em (18) e (19). Desta forma, sabe-se que o processo H2-DOF é uma planta didática bem explorada, onde autores como Patel et al. (2017) e Madebo et al. (2024) optaram por trabalhar com o modelo não linear da planta, enquanto outros autores como CRUZ et al. (2019) e Khalid et al. (2017) optaram por usar um modelo linearizado para projetar os seus controladores, onde cada tipo de modelo exige uma análise que difere em sua metodologia para se gerar resultados.

Na presente pesquisa, é utilizado um modelo linearizado para se aplicar ferramentas de descentralização de malhas de controle, com o uso da pré-compensação pelo ganho DC da planta, descrito por Stevens, Lewis e Johnson (2016). Assim, ao aplicar essa metodologia, são feitas análises sobre as dinâmicas desacopladas para as malhas de arfagem e guinhada, pelo método da matriz de ganhos relativos (RGA - *Relative Gain Array*) presente em Skogestad e Postlethwaite (2005), para que se possa aplicar reduções de ordem do tipo FOPDT (First Order Plus Dead Time) e projetar dois controladores baseados no caso SISO para controlar o sistema acoplado.

Com isso, o sistema MIMO do processo H2-DOF, em (18) e (19), precisa ser convertido da representação em espaço de estados para a forma polinomial em função de transferência. Para isso, inicialmente é realizada uma análise sobre as características do sistema multivariável estimado, para que se possa verificar se é realizável ou não a aplicação de controle, em termos de controlabilidade e observabilidade, enquanto sistema MIMO para o uso em trabalhos futuros.

#### 3.2.1.1 Controlabilidade e Observabilidade

Para o sistema no espaço de estados de (18) e (19) são usadas as matrizes  $A$  e  $B$  que são relacionadas à matriz de controlabilidade, que nesta pesquisa possui ordem  $n = 4$  em (43) (Stevens; Lewis; Johnson, 2016).

$$C_o = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix} \quad (43)$$

onde,  $C_o$  é a matriz de controlabilidade.

Desta maneira, o sistema é considerado como completamente controlável se e somente se o valor calculado a partir de  $C_o$  possuir valor igual à ordem  $n$  do sistema, como em (44).

$$\rho_{C_o} = n = 4 \quad (44)$$

Paralelo a isso, a observabilidade do sistema é verificada com o uso das matrizes A e C que produzem a relação para a matriz de observabilidade, calculada por:

$$O_b = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \quad (45)$$

onde,  $O_b$  é a matriz de observabilidade.

Destarte, o sistema será completamente observável se e somente se o valor calculado a partir de  $O_b$  seguir a mesma premissa de  $C_o$ , a seguir:

$$\rho_{O_b} = n = 4 \quad (46)$$

Portanto, após a realização dessa análise na planta MIMO, prossegue-se para a aplicação das metodologias de pré-compensação e RGA, no intuito de gerar os modelos polinomiais que serão usados nos projetos de controladores clássicos. Assim, esta etapa é concebida apenas como uma contribuição complementar para futuros trabalhos com o uso de controladores centralizados, tendo em vista que, na presente pesquisa, os projetos dos controladores são descentralizados e não possuem a necessidade de fazer esta análise.

### 3.2.1.2 RGA e pré-compensação

No intuito de projetar as malhas de controle descentralizadas do processo, é necessário considerar o efeito *cross-coupling* gerado pelas influências que relacionam as entradas de  $u$  para as saídas de  $y$  no projeto (Skogestad; Postlethwaite, 2005). Assim, a matriz de ganhos relativos apresentada por Bristol (1966) e Skogestad e Postlethwaite (2005) é definida como uma ferramenta para medir a interação entre as variáveis do sistema MIMO e analisar o sistema em regime estacionário, em que este pode ser projetado para frequências próximas à de corte do sistema ( $\omega = 0$  rad/s) (CRUZ et al., 2019).

Primeiramente, para utilizar o RGA é necessário converter o modelo estimado para a forma polinomial em função de transferência, pela equação em (47), assim, é verificado que o modelo é uma matriz de funções de transferências representada em (48), onde as funções da diagonal principal representam as dinâmicas diretas de

entradas/saídas e a diagonal secundária as dinâmicas indiretas devido ao acoplamento entre entradas/saídas.

$$G_p(z) = C(zI - \hat{A})^{-1} \hat{B} + D \quad (47)$$

$$G_p(z) = \begin{bmatrix} G_{p_{11}}(z) & G_{p_{12}}(z) \\ G_{p_{21}}(z) & G_{p_{22}}(z) \end{bmatrix} = \begin{bmatrix} G_{(u_1, y_1)}(z) & G_{(u_2, y_1)}(z) \\ G_{(u_1, y_2)}(z) & G_{(u_2, y_2)}(z) \end{bmatrix} \quad (48)$$

A partir disso, Skogestad e Postlethwaite (2005) consideram fatores relevantes para que o projeto descentralizado não seja comprometido devido a estas influências, assim, para que o RGA apresente uma relação satisfatória para o projeto descentralizado, é necessário que a relação direta entre entradas/saídas (como  $u_1 \rightarrow y_1$ ) seja próxima de 1, enquanto a relação indireta das entradas/saídas (como  $u_1 \rightarrow y_2$ ) seja próxima de 0.

Então, seja  $G_p(z)$  a matriz de função de transferência do sistema discreto ou contínuo, em regime permanente, quando  $z = 1$  para sistemas discretos ou  $s = 0$  para sistemas contínuos, a matriz RGA é denotada por  $\Lambda$  e definida a seguir:

$$f_{RGA}(G_p) = \Lambda(G_p(z)) = G_p(z) \circ (G_p(z)^{-1})^T \quad (49)$$

onde,  $\circ$  denota a multiplicação elemento a elemento,  $(G_p^{-1})^T$  é a transposta da inversa de  $G_p$  e  $G_p \in \mathbb{R}^{n \times n}$  deve ser **quadrada e não singular** para a definição ser válida.

$$\Lambda(G_p) = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} = \begin{bmatrix} \lambda_{11} & 1 - \lambda_{11} \\ 1 - \lambda_{11} & \lambda_{11} \end{bmatrix}; \lambda_{11} = \frac{1}{1 - \frac{G_{p_{12}} G_{p_{21}}}{G_{p_{11}} G_{p_{22}}}} \quad (50)$$

onde,  $\lambda_{11}$  é o termo que relaciona as influências do processo acoplado da matriz de funções de transferência  $G_p(z)$  com seus elementos  $G_{p_{ij}}$ .

É válido destacar que o RGA nesta pesquisa é usado meramente como uma ferramenta de análise, para averiguar os empareiramentos das relações diretas e indiretas de entradas/saídas do sistema, sendo também uma alternativa à mera pré-compensação do ganho DC. Com isso, pode-se constatar que as dinâmicas dos acoplamentos do sistema podem comprometer o desempenho de controladores projetados individualmente para cada malha. Para mitigar esses efeitos, aplica-se uma pré-compensação escalar ( $P_C$ ) baseada na inversa da matriz de ganho DC do sistema ( $K_S$ ), representado pela resposta em regime permanente do sistema a uma entrada constante (isto é,  $\lim_{z \rightarrow 1} G(z)$  no domínio discreto), calculado com base nas matrizes de parâmetros estimados de (18) e (19) tem-se:

$$K_S = C(I - \hat{A})^{-1} \hat{B} + D = \begin{bmatrix} K_{S_{11}} & K_{S_{12}} \\ K_{S_{21}} & K_{S_{22}} \end{bmatrix} \quad (51)$$

$$P_C = K_S^{-1} = \begin{bmatrix} P_{C_{11}} & P_{C_{12}} \\ P_{C_{21}} & P_{C_{22}} \end{bmatrix} \quad (52)$$

A motivação para essa abordagem está na padronização da resposta estática do sistema, garantindo que o efeito de uma entrada de referência constante produza uma saída com módulo unitário após a compensação. Assim, o sistema pré-compensado passa a ter ganho unitário em regime permanente, o que simplifica o projeto do controlador e melhora a previsibilidade do comportamento do sistema em malha fechada (Stevens; Lewis; Johnson, 2016).

Em termos práticos, essa compensação implementada em (52), por meio de um ganho escalar, é aplicada à entrada do sistema. Desta forma, esse ganho  $K_S$  calculado em malha aberta com o modelo identificado, tem a entrada pré-compensada da seguinte forma:

$$\begin{aligned} \mathbf{x}(k) &= \hat{A}\mathbf{x}(k-1) + \hat{B}(P_c)u(k-1) \\ \mathbf{y}(k) &= C\mathbf{x}(k) + Du(k) \end{aligned} \quad (53)$$

Desta forma, ao multiplicar  $P_c$  com a matriz de parâmetros estimados da entrada ( $\hat{B}$ ) em (54), é possível extrair um novo sistema pré-compensado em (55), de forma a atender à proposição descrita, para se projetar os controladores com base somente nas funções de transferência da diagonal principal ( $G_{P_{C11}}$  e  $G_{P_{C22}}$ ), que produzem as dinâmicas diretas das entradas/saídas e podem ser observadas na estrutura de controle em malha fechada da Figura 11.

$$B_{comp} = \hat{B}P_c \quad (54)$$

$$\begin{aligned} \mathbf{x}_{P_c}(k) &= \hat{A}\mathbf{x}_{P_c}(k-1) + B_{comp}\mathbf{u}(k-1) \\ \mathbf{y}_{P_c}(k) &= C\mathbf{x}_{P_c}(k) + D\mathbf{u}(k) \end{aligned} \Rightarrow G_{P_c}(z) = \begin{bmatrix} G_{P_{C11}}(z) & G_{P_{C12}}(z) \\ G_{P_{C21}}(z) & G_{P_{C22}}(z) \end{bmatrix} \quad (55)$$

onde  $B_{comp}$  é a matriz  $\hat{B}$  pré-compensada por  $P_c$ , associada à entrada  $u(k)$  do sistema pré-compensado no espaço de estados, cuja conversão resulta na matriz de funções de transferência  $G_{P_c}$ , obtida conforme a equação (47).

Na Figura 11 observa-se que, embora as funções de transferência  $G_{P_{C12}}$  e  $G_{P_{C21}}$  estejam presentes na malha fechada, o projeto de controladores descentralizados com base em  $G_{P_{C11}}$  e  $G_{P_{C22}}$  permanece factível. Isso se deve à normalização do ganho estático global promovida pela pré-compensação  $P_c$ , a qual condiciona a entrada  $u(k)$  a exercer uma influência predominante, com ganho próximo de 1, nas malhas diretas ( $u_1 \rightarrow y_1$  e  $u_2 \rightarrow y_2$ ), enquanto reduz significativamente a influência cruzada, com pesos próximos de zero nas malhas indiretas. Dessa forma, garante-se uma ação de controle predominantemente desacoplada e equilibrada entre as variáveis do sistema, e pode ser verificado ao aplicar o método da matriz de ganhos relativos (RGA) após a pré-compensação do sistema (Skogestad; Postlethwaite, 2005).

Por fim, é necessário aproximar as dinâmicas de  $G_{P_{C11}}$  e  $G_{P_{C22}}$  para uma ordem inferior, pois a ordem destas plantas em (56) é superior à grandeza de ordem  $n$  usada para os projetos dos controladores PI e PID (Landau; Zito, 2006). Desta maneira, a

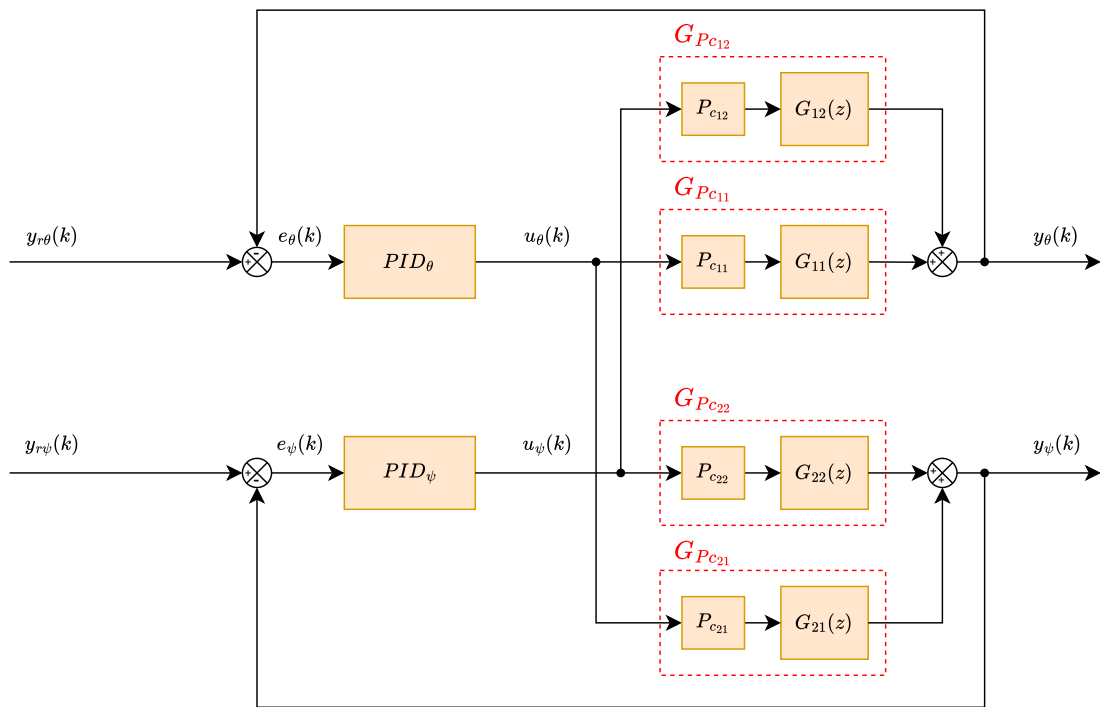


Figura 11 – Estrutura de Projeto Descentralizado para os Controladores Clássicos.

Fonte: Autor.

fim de permitir a síntese dos polinômios da estrutura RST, aplicam-se aproximações por modelagem de primeira ordem com atraso de transporte (FOPDT - *First Order Plus Dead Time*) de Ziegler-Nichols que são discretizadas via ZOH em (57) (Coelho; Coelho, 2004).

$$G_{Pc_{11}}(z), G_{Pc_{22}}(z) = \frac{b_0 z^{-1} + b_1 z^{-2} + b_2 z^{-3} + b_3 z^{-4}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} + a_4 z^{-4}} \quad (56)$$

$$G(s) = \frac{Ke^{-\theta s}}{\tau s + 1} \xrightarrow{\text{ZOH}} G(z) = \frac{b_0 z^{-d}}{1 + a_1 z^{-1}} \quad (57)$$

### 3.2.2 Considerações para o Projeto na Estrutura RST-PID

O uso da estrutura RST permite uma separação clara entre os objetivos de estabilidade (via alocação de polos em  $H_{cl}(z^{-1})$ ) e de rastreamento (via  $T(z^{-1})$ ). O atraso  $d$  da planta impacta diretamente o grau mínimo necessário dos polinômios  $R(z^{-1})$  e  $S(z^{-1})$ , aumentando a complexidade do controlador. Entretanto, essa abordagem permite o controle de uma ampla gama de plantas, inclusive aquelas com atraso, com boa capacidade de modelagem e desempenho (Ogata, 1995; Landau et al., 2011; Åström; Wittenmark, 2013).

Com as considerações desenvolvidas na seção 3.1 e 3.2, para que a lei de controle  $u(k)$  da estrutura RST em (35) seja equivalente às características da lei de controle  $u(k)$  do controlador PID discreto em (32), é necessário considerar os termos R,S e T da seguinte forma:

$$\begin{cases} S(z^{-1}) = T(z^{-1}) \\ R(z^{-1}) = 1 \\ S(z^{-1}) = s_0 + s_1 z^{-1} + s_2 z^{-2} \end{cases} \quad (58)$$

Substituindo as considerações em (58) na equação (35), e sabendo-se que o erro  $e(k)$  é a diferença entre os sinais  $y_r(k)$  e o  $y(k)$ , é possível desenvolver a estrutura equivalente a seguir:

$$e(k) = y_r(k) - y(k) \quad (59)$$

$$\Delta u_{RST}(k) = S(z^{-1})y_r(k) - S(z^{-1})y(k) \quad (60)$$

$$u_{RST}(k) - u_{RST}(k-1) = S(z^{-1})e(k) \quad (61)$$

$$u_{RST}(k) = u_{RST}(k-1) + s_0 e(k) + s_1 e(k-1) + s_2 e(k-2) \quad (62)$$

A partir disso, é possível produzir uma "máscara do controlador", que é necessária para sintonizar os ganhos do controlador PID com a estrutura de projeto RST por alocação de polos (Araujo, M. et al., 2017). Então, partindo da equação (33) para isolar os ganhos  $K_p$ ,  $K_i$  e  $K_d$ , em termos de  $s_0$ ,  $s_1$ ,  $s_2$  e  $T_s$ , são produzidas as equações a seguir:

$$s_2 = \frac{K_d}{T_s} \rightarrow K_d = s_2 T_s \quad (63)$$

Ao substituir o ganho  $K_d$  da equação (63) no termo  $s_1$  da equação (33), temos:

$$s_1 = -K_p + 2 \frac{(s_2 T_s)}{T_s} \rightarrow K_p = -s_1 - 2s_2 \quad (64)$$

Desta maneira, ao isolar o ganho  $K_p$  em (64), é possível encontrar o ganho  $K_i$  ao substituir (63) e (64) no termo  $s_0$  da equação (33), descrito como:

$$s_0 = (-s_1 - 2s_2) + K_i T_s + \frac{(s_2 T_s)}{T_s} \rightarrow K_i = \frac{s_0 + s_1 + s_2}{T_s} \quad (65)$$

Para sintonizar os ganhos pelos termos  $s_0$ ,  $s_1$  e  $s_2$ , o projeto do controlador RST-PID deve permitir: a seleção das especificações de tolerância do sobressinal ( $M_p$ ) e do tempo de subida ( $T_r$ ), em malha fechada. Então, ao considerar uma equação na forma de controle incremental de segunda ordem, é necessário um modelo característico

de segunda ordem mais um polo auxiliar do observador, para satisfazer a igualdade polinomial do projeto, descrito por (66).

$$G_{mf}(s) = \frac{\omega_n^2}{(s^2 + 2\zeta\omega_n s + \omega_n^2)} \cdot \frac{1}{(s + \alpha)} \xrightarrow{ZOH} \frac{b_0 z^{-1} + b_1 z^{-2} + b_2 z^{-3}}{(1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3})} \quad (66)$$

onde,  $\zeta$  é o fator de amortecimento desejado,  $\omega_n$  é a frequência natural desejada (em rad/s) e  $\alpha$  é a posição dinâmica escolhida do polo auxiliar, de forma que não afete a dinâmica dominante do processo.

Destarte, para se calcular os termos  $\zeta$  e  $w_n$  são usadas fórmulas derivadas de  $M_p$  e  $T_r$  em (67) e (68).

$$\zeta = \frac{-\ln\left(\frac{M_p}{100}\right)}{\sqrt{\pi^2 + \left(\ln\left(\frac{M_p}{100}\right)\right)^2}} \quad (67)$$

$$w_n = \frac{\pi - \arccos(\zeta) \cdot \frac{\pi}{180}}{T_r \sqrt{1 - \zeta^2}} \quad (68)$$

A planta considerada no projeto tem a dinâmica aproximada de primeira ordem e do tipo-0, como descrita em (57) com um atraso de  $d = 1$ . Então, tem-se o acréscimo de um polo do integrador discreto ( $\Delta$ ) que é relacionado à malha de controle, descrito como:

$$G_p(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{b_0 z^{-1}}{1 + a_1 z^{-1}} \quad (69)$$

$$\Delta A(z) = 1 + \bar{a}_1 z^{-1} + \bar{a}_2 z^{-2} \quad (70)$$

onde,  $\bar{a}_1$  e  $\bar{a}_2$  são parâmetros do polinômio aumentado  $\Delta A(z)$ .

O polinômio desejado de malha fechada,  $H_{cl}(z^{-1})$ , pode ser composto por: polinômios de especificação de malha fechada desejada ( $H_c(z^{-1})$ ) e o polinômio do observador ( $H_o(z^{-1})$ ). Dessarte, para satisfazer a igualdade polinomial estabelecida pela equação Diofantina, ao observar a equação (66) considera-se um polinômio de segunda ordem para  $H_c(z^{-1})$  e um polinômio de primeira ordem para  $H_o(z^{-1})$ , onde  $G_c(z^{-1})$  precisa ser assintoticamente estável para o polinômio do observador  $H_o(z^{-1})$  funcionar, como é descrito a seguir:

$$H_c(z^{-1}) = 1 + h_{c1} z^{-1} + h_{c2} z^{-2} \quad (71)$$

$$H_o(z^{-1}) = 1 + h_o z^{-1} \quad (72)$$

$$H_{cl}(z^{-1}) = H_c(z^{-1})H_o(z^{-1}) = (1 + h_{c1}z^{-1} + h_{c2}z^{-2})(1 + h_0z^{-1}) \quad (73)$$

Portanto, para extrair os termos necessários para sintonizar ( $s_0$ ,  $s_1$  e  $s_2$ ) os ganhos do controlador PID, usa-se as considerações presentes em (66), (69), (70) e (73) para serem substituídas na equação (41) e fazer o equacionamento polinomial. Então, é produzida a igualdade polinomial em função da ordem  $n$  (em  $z^{-n}$ ), para se isolar os termos de  $S(z^{-1})$ , e expressar as equações a seguir:

$$s_0 = \frac{H_{c1} + H_0 - \bar{a}_1}{b_0} \quad (74)$$

$$s_1 = \frac{H_{c2} + H_{c1}H_0 - \bar{a}_2}{b_0} \quad (75)$$

$$s_2 = \frac{H_{c2}H_0}{b_0} \quad (76)$$

### 3.2.3 Considerações para o Projeto na Estrutura RST-PI

Como descrito na seção 3.1 pela Figura 8 e expressa na equação (32), o controlador PI é uma estrutura oriunda do controlador PID, onde a estrutura RST também pode ser utilizada para sintonizar os ganhos do controlador PI. Assim, apesar da ausência do termo derivativo, esse tipo de estrutura também é bem comum em sistemas industriais devido à sua simplicidade e bom desempenho para processos com dinâmica dominante de primeira ordem, onde a principal diferença em relação ao projeto PID está na escolha da forma dos polinômios que compõem a estrutura RST.

A partir disso, ao zerar a parcela do ganho derivativo em (27) e aproximar sua estrutura para a forma discreta pelo método *Backward Difference* em (28), tem-se a lei de controle  $u(k)$  do controlador PI discreto, como:

$$u(k) = u(k-1) + s_0e(k) + s_1e(k-1) \quad (77)$$

$$\begin{cases} s_0 = K_p + K_i T_s \\ s_1 = -K_p \end{cases} \quad (78)$$

Desta forma, é considerado que os termos do projeto do controlador por alocação de polos são obtidos ao impor mudanças na forma do polinômio  $S(z^{-1})$ , em que o sistema em malha fechada possua características desejadas de estabilidade e desempenho, sem a necessidade de um termo derivativo, descrito como:

$$\begin{cases} S(z^{-1}) = T(z^{-1}) \\ R(z^{-1}) = 1 \\ S(z^{-1}) = s_0 + s_1z^{-1} \end{cases} \quad (79)$$

A “máscara do controlador” pode ser produzida ao aplicar a mesma lógica do projeto anterior (Seção 3.3.2), assim, ao isolar os ganhos  $K_p$  e  $K_i$  em função dos termos  $s_0$  e  $s_1$ , é possível obter as equações para sintonizar os ganhos do controlador PI em (80) e (81).

$$K_p = -s_1 \quad (80)$$

$$K_i = \frac{s_0 + s_1}{T_s} \quad (81)$$

Embora as considerações de projeto do controlador RST-PID (Seção 3.3.2) tenham sido realizadas pela seleção da tolerância de sobressinal e o tempo de subida em malha fechada, o projeto do controlador RST-PI é realizado ao considerar o sistema de primeira ordem em (82) mais um polo do observador  $H_o(z^{-1})$ , que difere do modelo característico utilizado em (66). Desta maneira, o projeto é produzido para permitir a seleção dinâmica desejada pela constante de tempo de malha fechada ( $\tau_{mf}$ ) e uma relação de posição  $\sigma$ , entre o polo auxiliar do observador  $H_o(z^{-1})$  e o polo gerado pela seleção de  $\tau_{mf}$ .

$$G_{mf}(s) = \frac{Ke^{-\theta s}}{(\tau_{mf}s + 1)} \cdot \frac{1}{(s + \sigma)} \xrightarrow{ZOH} \frac{b_0z^{-1} + b_1z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}} \quad (82)$$

Ademais, a planta descrita em (69) é utilizada para este projeto, onde a expansão de  $\Delta A(z)$  da equação (70) se mantém, logo é possível descrever o polinômio desejado de malha fechada  $H_{cl}(z^{-1})$  como:

$$H_c(z^{-1}) = 1 + h_{c1}z^{-1} \quad (83)$$

$$H_o(z^{-1}) = 1 + h_o z^{-1} \quad (84)$$

$$H_{cl}(z^{-1}) = H_c(z^{-1})H_o(z^{-1}) = (1 + h_{c1}z^{-1})(1 + h_o z^{-1}) \quad (85)$$

Portanto, pode-se observar que as mudanças feitas para o projeto do controlador RST-PI em (79), (82) e (85) satisfazem a igualdade polinomial, quando são substituídas em (41). Desta forma, o equacionamento matemático é realizado, como no projeto anterior, para se isolar os termos em função das grandezas de  $z^{-n}$ , em que são extraídos os termos  $s_0$  e  $s_1$  para sintonizar os ganhos em (80) e (81) a seguir:

$$s_0 = \frac{h_{c1} - \bar{a}_1}{b_0} \quad (86)$$

$$s_1 = \frac{h_o - \bar{a}_2}{b_0} \quad (87)$$

### 3.3 REDES NEURAIS ARTIFICIAS

As Redes Neurais Artificiais (RNAs) são sistemas fundamentais em aplicações de classificação de dados complexos. Desta maneira, de acordo com Wingerter, Santos e Barbosa (2020), quando as RNAs são configuradas como modelos computacionais baseados em modelagem matemática, bioinspiradas pelo funcionamento do sistema nervoso central humano, estas são capazes de realizar o aprendizado de máquina para reconhecer padrões e executar tarefas complexas que, de outra forma, seriam desafiadoras de se resolver apenas com a programação tradicional baseada em regras comuns (*if-then, if-else, for, while*).

Nos últimos anos, as RNAs têm sido comumente utilizadas em áreas de visão computacional, processamento de linguagem e previsão de séries temporais. Entre as arquiteturas clássicas utilizadas, o Perceptron de Múltiplas Camadas (MLP) é uma classe de redes neurais de avanço (*feedforward*) totalmente conectadas, onde os neurônios são organizados em camadas sequenciais: camada de entrada, camadas ocultas e camada de saída. Dessa forma, cada neurônio de uma camada está conectado a todos os neurônios da próxima camada, permitindo que a informação se propague pela rede.

Ademais, apesar das MLPs serem uma estrutura clássica e mais simples, sua capacidade de atender demandas logísticas regionais, descritas por Guo et al. (2021), demonstra que essas redes podem capturar eficientemente variações sazonais e tendências de longo prazo nos dados logísticos. De modo similar, Nosrati, Khalili e Gerdooei (2021) mostraram que as MLPs podem ser treinadas para aprender sobre a dinâmica de um processo, ao destacar o desempenho dessas redes em gerar simulações precisas sobre dinâmicas complexas nos processos de manufatura. Portanto, as redes MLPs continuam sendo uma estrutura de uso viável no campo das RNAs, oferecendo soluções eficazes para uma ampla gama de problemas de classificação e regressão (Nosrati; Khalili; Gerdooei, 2020).

Segundo Cybenko (1989), as redes neurais artificiais MLPs possuem aproximadores universais em suas camadas, para auxiliar na classificação dos dados, onde uma única camada oculta é suficiente para aproximar uniformemente qualquer função contínua. O teorema da aproximação universal pressupõe que a função contínua a ser aproximada é dada e que uma camada oculta de tamanho ilimitado está disponível para a aproximação, onde ambas as premissas são violadas na maioria das aplicações práticas de perceptrons multicamadas (Haykin, 2009). Entretanto, Haykin (2009) descreve que essa visão de aproximação universal é importante do ponto de vista teórico, pois fornece a ferramenta matemática necessária para a viabilidade de redes *feedforward* com uma única camada oculta, para uma classe de soluções aproximadas. Destarte, apesar do teorema não ser construtivo, sem ele, a solução para o problema seria impraticável. Portanto, o teorema não especifica como determinar um perceptron

multicamadas com as propriedades de aproximação declaradas.

A partir disso, em situações de classificações complexas, a aproximação torna-se desafiadora de ser aprimorada em um ponto, sem piorá-la em outro ponto. Desta forma, uma MLP com duas camadas ocultas pode tornar a classificação mais manejável para qualquer tipo de aproximação de função, como: *sigmoid*, *tansig* e *logsig* (Haykin, 2009).

Essas redes são comumente treinadas na fase de retropropagação com algoritmos, como o *Backpropagation* ou suas variantes (*Levenberg-Marquardt*, *Quick-Drop*), que são métodos usados para treinar estruturas de redes MLPs, consistindo em um aprendizado supervisionado pela correção do erro entre a saída desejada e a saída prevista pela rede, para então retropropagar o erro e atualizar os parâmetros da rede, como apresentados na estrutura simplificada da Figura 12 (Pádua; Leon; Ludermir, 2007; Haykin, 2009).

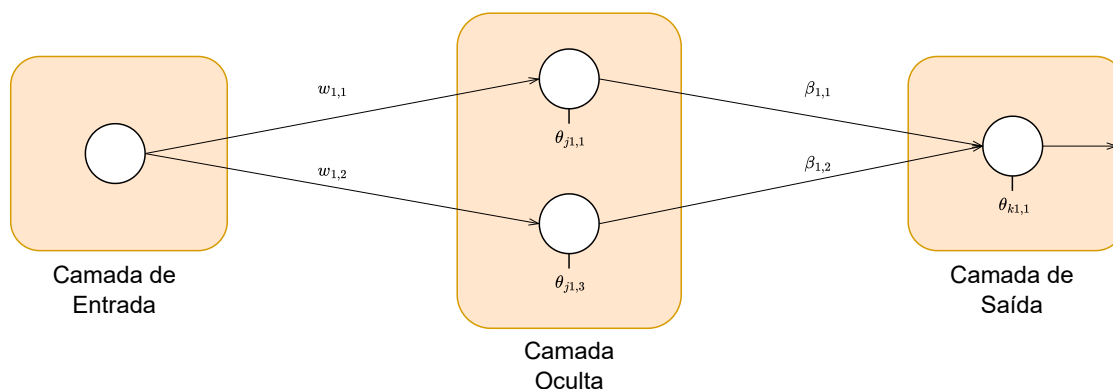


Figura 12 – Estrutura da Rede Neural Perceptron de Múltiplas Camadas (MLP).

Fonte: Autor.

### 3.3.1 Algoritmo *Backpropagation*

Primeiramente, a lógica do algoritmo exige a execução da propagação direta (*forward propagation*), comumente realizada pela MLP na Figura 12, onde os dados de entrada são propagados positivamente pela rede, camada por camada, sendo esse processo denominado de sinal funcional (estímulo). Em seguida, na última camada da rede, é calculada a saída estimada e compara-se com uma saída desejada para se produzir o sinal de erro da saída, que será retropropagado (neurônio por neurônio) pela rede, com a finalidade de refinar as ligações sinápticas que estão entre as camadas de neurônios (Pádua; Leon; Ludermir, 2007). Entretanto, antes de se aplicar esse procedimento, caso os vetores de dados usados para treinar a RNA sejam de uma grandeza muito alta, é recomendável considerar métodos de normalização para se diminuir o estresse computacional e otimizar os cálculos da rede, como o pré-processamento dos

dados (entrada/saída) pelo método Min-Max em (88), para valores normalizados no intervalo de  $[-1,1]$  ou  $[0,1]$ .

$$X' = \frac{(X - X_{\min})(L_{\max} - L_{\min})}{X_{\max} - X_{\min}} + L_{\min} \quad (88)$$

$X$  é o valor original dos dados,  $X_{\min}$  é o menor valor no conjunto de dados,  $X_{\max}$  é o maior valor no conjunto de dados, e  $X'$  é o valor normalizado dos dados dentro dos limites mínimo ( $L_{\min}$ ) e máximo ( $L_{\max}$ ).

Após aplicar a normalização dos dados, é necessário escolher os hiperparâmetros do projeto, como a definição do número de neurônios na camada oculta ( $N_h$ ), que é escolhido com base na premissa:

$$N_h \geq N_i + 1 \quad (89)$$

onde,  $N_i$  é o número de neurônios na camada de entrada.

Os pesos e os bias são inicializados de forma aleatória para evitar um treinamento de rede tendencioso em direção aos mesmos resultados, o que poderia ocasionar uma diminuição significativa do desempenho de classificação (acurácia) da RNA, para possíveis padrões não-lineares que não estão presentes no conjunto de dados de treinamento (Sarangapani, 2018).

Deste modo, os vetores de dados de entrada e saída são embaralhados e normalizados para serem divididos em fases de treinamento, validação e teste. Assim, cada dado é transferido da camada de entrada para a camada oculta e, então, para a camada de saída, ponderado pela função de ativação sigmoide presente ao final das camadas oculta e de saída, conforme os cálculos da propagação direta na MLP em (90) até (93).

$$v_j(n) = \sum_{i=1}^p w_{ij} x_i(n) + \theta_j \quad (90)$$

$$s_j(n) = f(v_j(n)) \quad (91)$$

$$v_k(n) = \sum_{j=1}^m \beta_{jk} s_j(n) + \theta_k \quad (92)$$

$$y_k(n) = g(v_k(n)) \quad (93)$$

A cada iteração  $n$ ,  $v_j$  e  $v_k$  são os somatórios,  $w$  e  $\beta$  são os pesos, e  $\theta_j$  e  $\theta_k$  são os bias, respectivamente, da camada oculta e a de saída que possuem as funções de ativação  $f$  e  $g$  no final das suas estruturas.

A saída prevista  $y_k(n)$  em (93) é comparada com a saída desejada  $d_k(n)$ , relacionada ao vetor de dados de saída, para gerar o erro de previsão  $e_k(n)$ , descrito como:

$$e_k(n) = d_k(n) - y_k(n) \quad (94)$$

onde  $e_k$  é o erro calculado após a camada de saída  $k$  na iteração  $n$ .

A equação (94) é usada para calcular o erro médio quadrático em (95), que compara a saída da rede com o valor desejado a cada iteração  $n$ , e em seguida calcula a função de custo a ser minimizada ( $E_{av}$ ) durante o processo de aprendizado, para todos os  $N$  padrões apresentados à RNA em (96).

$$E(n) = \frac{1}{2} \sum_{k \in C} e_k^2(n) \quad (95)$$

$$E_{av} = \min \frac{1}{N} \sum_{n=1}^N E(n) \quad (96)$$

Todos os neurônios da camada de saída  $k$  pertencem ao conjunto  $C$ , sendo  $E_{av}$  a função custo do erro médio quadrático a ser minimizada e uma função de todos os parâmetros livres da RNA.

Essencialmente, na fase de retropropagação, o algoritmo *Backpropagation* considera os ajustes dos pesos em direção contrária ao gradiente do erro médio quadrático ( $\nabla E(n)$ ) para se reduzir a função custo  $E_{av}$  em (97) e (99), onde os novos pesos sinápticos atualizados são calculados pela soma dos pesos atuais mais o ajuste em (98) e (100). Assim, este processo de ajuste ocorre de forma sequencial da camada de saída para a camada oculta e depois para a camada de entrada.

$$\Delta\beta_{jk}(n) = -\eta \frac{\partial E(n)}{\partial \beta_{jk}(n)} \quad (97)$$

$$\beta_{jk}(n+1) = \beta_{jk}(n) + \Delta\beta_{jk}(n) \quad (98)$$

$$\Delta w_{ij}(n) = -\frac{\partial E(n)}{\partial w_{ij}(n)} \quad (99)$$

$$w_{ij}(n+1) = w_{ij}(n) + \Delta w_{ij}(n) \quad (100)$$

$\beta_{jk}(n+1)$  e  $w_{ij}(n+1)$  são os novos vetores de pesos ajustados e  $\eta$  é a taxa de aprendizado tida como um hiperparâmetro.

É notório destacar que para encontrar os ajustes em (97) e (100) são utilizadas as regras da cadeia pelas derivadas parciais contidas em (101) e (102) para se calcular os gradientes locais  $\delta$  para se gerar os ajustes.

$$\frac{\partial E}{\partial \beta_{jk}} = \frac{\partial E}{\partial \mathbf{e}_k} \frac{\partial \mathbf{e}_k}{\partial y_k} \frac{\partial y_k}{\partial v_k} \frac{\partial v_k}{\partial \beta_{jk}} \quad (101)$$

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial \mathbf{e}_j} \frac{\partial \mathbf{e}_j}{\partial s_j} \frac{\partial s_j}{\partial v_j} \frac{\partial v_j}{\partial w_{ij}} \quad (102)$$

Ao expandir e calcular as derivadas parciais de (101) e (102), como apresentadas por Pádua, Leon e Ludermir (2007), é possível encontrar os gradientes locais ( $\delta$ ) para se calcular os ajustes dos pesos de (97) e (99), descritos como:

$$\delta_k = -\mathbf{e}_k g'(v_k) \quad (103)$$

$$\delta_j = -f'(v_j) \sum_k \delta_k w_{kj} \quad (104)$$

Enfim, ao substituir as equações (103) e (104) nas equações de ajustes dos vetores de pesos, em (97) e (99), tem-se as equações finais para os ajustes dos pesos a seguir:

$$\Delta \beta_{jk}(n) = -\eta \delta_k s_j \quad (105)$$

$$\Delta w_{ij}(n) = -\eta \delta_j x_i \quad (106)$$

Portanto, o mesmo equacionamento produzido para chegar aos ajustes dos vetores de pesos em (107) e (109), foi usado para os ajustes dos bias com o auxílio das derivadas parciais de (101) e (102) para  $\theta_k$  e  $\theta_j$ , onde é possível obter os ajustes dos vetores de bias dos neurônios das camadas de saída e oculta a seguir:

$$\Delta \theta_k(n) = -\eta \delta_k \quad (107)$$

$$\theta_k(n+1) = \theta_k(n) + \Delta \theta_k(n) \quad (108)$$

$$\Delta \theta_j(n) = -\eta \delta_j \quad (109)$$

$$\theta_j(n+1) = \theta_j(n) + \Delta \theta_j(n) \quad (110)$$

### 3.4 PROJETO DE SÍNTESE DO CONTROLADOR CLÁSSICO NEURAL (HIBRIDIZAÇÃO)

O estudo de caso centrado nesta dissertação é feito com base no princípio dos processos dinâmicos reais, como o H2-DOF apresentado no Capítulo 2, de que estes apresentam graus de incerteza quanto à sua dinâmica, onde essa incerteza pode ser

manifestada nos próprios parâmetros físicos do processo ou por perturbações externas. Dessa forma, a depender da intensidade desses problemas em processos controlados por controladores clássicos, cuja síntese não considera incertezas, esses sistemas tornam-se suscetíveis à perda de desempenho em diversas situações ou até mesmo à instabilidade, quando feitas as mudanças nos pontos de operação.

Nesse sentido, visando amenizar os efeitos de degradação produzidos nas trajetórias dos ângulos de arfagem e guinada no sistema H2-DOF, que possui características não lineares devido a variantes como atritos mecânicos, correntes de vento e incertezas de modelo, propõe-se uma metodologia de algoritmo de controles clássicos PI autoajustáveis pela rede neural. O método consiste em explorar a capacidade das redes neurais, que têm a propriedade de generalizar e aprender a aproximar características não lineares em padrões conhecidos pela rede, por meio das funções de ativação, de acordo com os parâmetros de operação do sistema em malha fechada para ajustar os ganhos obtidos nos projetos dos controles locais e tentar reduzir a degradação de desempenho nesse processo.

Desta maneira, para fazer o treinamento *off-line* da RNA pelo algoritmo *Back-propagation*, como descrito na seção 3.3.1, é necessário selecionar as entradas e saídas desejadas da rede, a partir da aquisição de dados obtidos nos experimentos produzidos com os controladores locais da Figura 11. Assim, após obter os dados experimentais, as variáveis são selecionadas de forma intuitiva a critério do projetista, em que essa seleção pode ser feita da seguinte forma:

$$D_{log(in)} = x_j = \begin{bmatrix} y_{r_\psi} & y_{r_\theta} & y_\psi & y_\theta & e_\psi & e_\theta \end{bmatrix} \quad (111)$$

$$D_{log(out)} = d_k = \begin{bmatrix} t_{0_\psi} & t_{0_\theta} & s_{0_\psi} & s_{0_\theta} & s_{1_\psi} & s_{1_\theta} \end{bmatrix} \quad (112)$$

$$D_{log} = \begin{bmatrix} x_j & d_k \end{bmatrix} = \begin{bmatrix} D_{log(in)} & D_{log(out)} \end{bmatrix} \quad (113)$$

onde,  $D_{log(in)}$  é a matriz de vetores dos dados paramétricos usados nos neurônios da camada de entrada, e  $D_{log(out)}$  é a matriz de vetores dos dados paramétricos das saídas desejadas para os neurônios da camada de saída.

Como observado pelas equações (111) e (112), as variáveis utilizadas para estimular os neurônios de entrada são as referências, as saídas e os erros, respectivamente, das malhas de guinada ( $\psi$ ) e arfagem ( $\theta$ ). Enquanto isso, as variáveis adotadas como saídas desejadas pela rede são os termos do projeto RST para sintonizar a máscara dos controladores PI, que fazem o autoajuste de acordo com a mudança das variáveis nas malhas de controle. Então, para cada variável de entrada ou de saída, é adotado usar uma quantidade de neurônios equivalente, para que possam transmitir a informação dos vetores de dados pela rede sem causar problemas no autoajuste

dos ganhos na saída da rede para os controladores clássicos, como apresentado na estrutura da RNA na Figura 13.

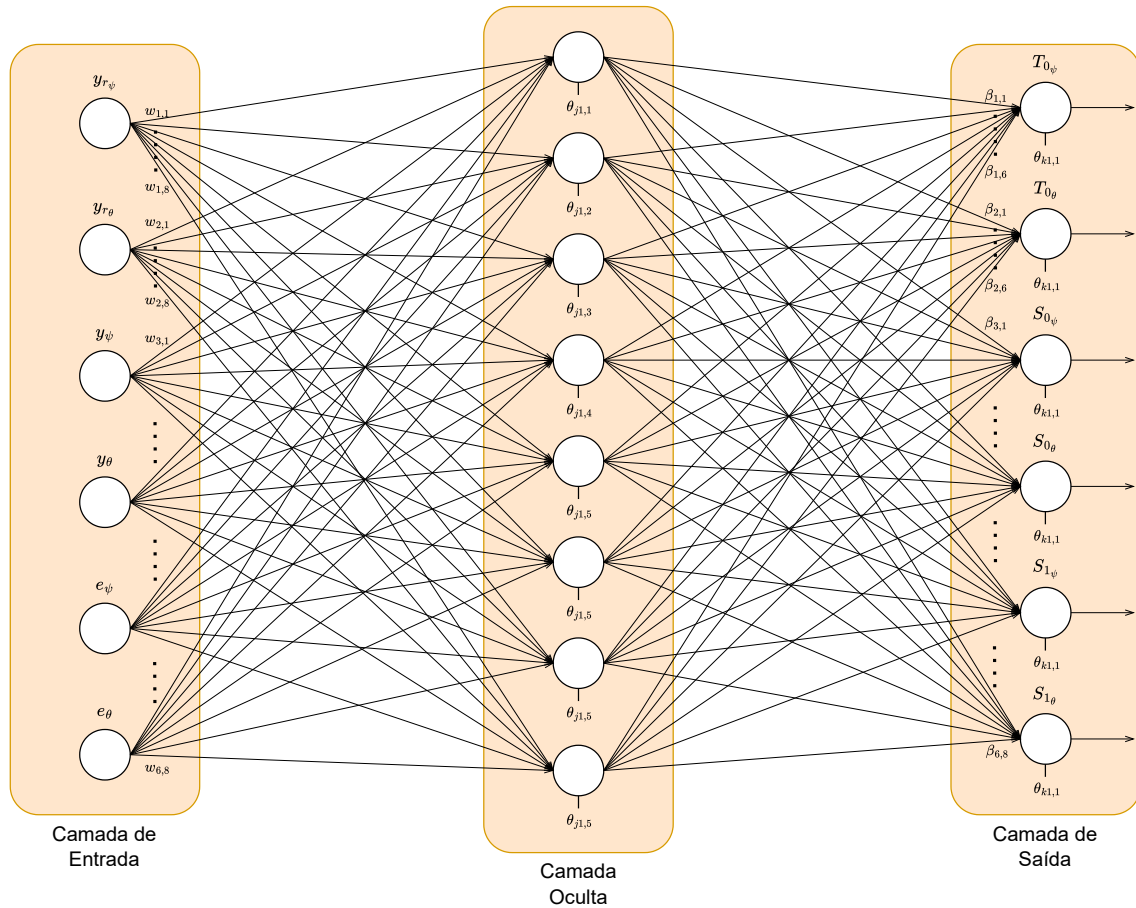


Figura 13 – Estrutura de Projeto da Rede Neural (6-8-6).

Fonte: Autor.

Para a inicialização do treinamento da RNA, é considerada uma taxa de aprendizagem  $\eta = 0,01$ , com os seguintes critérios de parada do algoritmo:

- O número máximo de épocas para fase de treinamento é 1000.
- Se a meta de desempenho da função custo alcançar uma grandeza próxima de 0 ( $E_{av} = 1 \times 10^{-12}$ ).
- Caso o gradiente mínimo chegue próximo da meta mínima da função custo ( $\nabla E = 1 \times 10^{-12}$ ).
- Se verificado que  $E_{av}$  da etapa de validação é mínimo, depois de feitas 100 épocas de tentativas na etapa de treinamento para se gerar um novo valor mínimo na fase de validação.

Os vetores das bases de dados em (111) e (112) são divididos para as fases de treinamento, validação e teste. Desta forma, para a fase de treinamento são utilizados

60% do total das bases de dados, enquanto para a etapa de validação e teste são usados 25% e 15% , respectivamente, para cada fase em (114).

$$D_{log} \begin{cases} F_{trei} \rightarrow 0,60D_{log} \\ F_{val} \rightarrow 0,25D_{log} \\ F_{test} \rightarrow 0,15D_{log} \end{cases} \quad (114)$$

Portanto, o controlador neural proposto consiste em duas partes: a estrutura dos projetos dos controladores clássicos, que controlam diretamente o processo em malha fechada com os parâmetros; e a outra, onde a rede neural treinada é responsável pela adaptação dos parâmetros do controlador de acordo com as condições de operação do sistema, a fim de alcançar uma determinada otimização no índice de desempenho, como mostrado no diagrama de blocos da Figura 14.

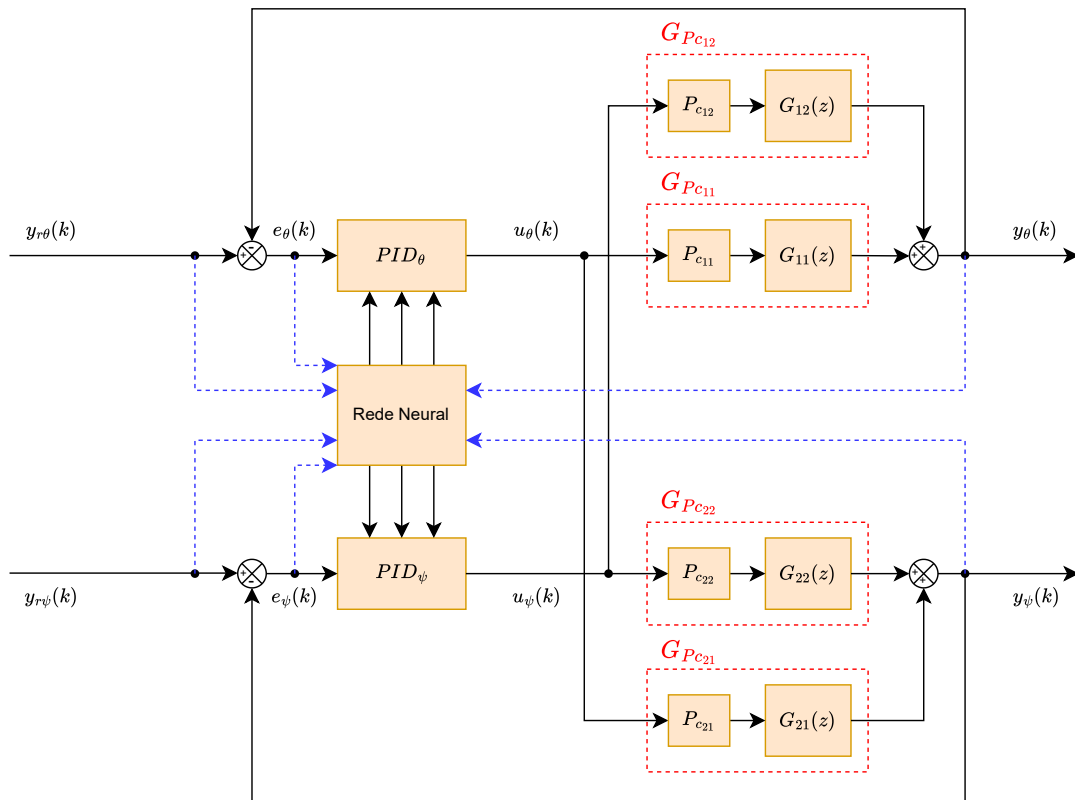


Figura 14 – Estrutura da Síntese do Controlador Clássico Neural.

Fonte: Autor.

Nesta dissertação, a estrutura dos controladores clássicos-neurais da Figura 14 é projetada para identificar, em tempo real, mudanças de região de operação nos eixos de guinada e arfagem do H2-DOF, adaptando os ganhos dos controladores locais conforme o comportamento dinâmico da planta, onde os dados obtidos em (111) e

(112) são usados para treinar a rede a selecionar dois projetos de controladores com dinâmicas de malha fechada diferentes (para cada malha).

Sendo assim, a trajetória projetada nos experimentos H2-DOF consiste em referências do tipo degrau para emular os movimentos de ascensão e descensão, onde são compostos pelos movimentos gerados pelo seguimento dos ângulos de guinada e de arfagem, sendo estimativas angulares que são convertidas a partir das saídas do processo em Volts para graus, como verificado na Figura 15.

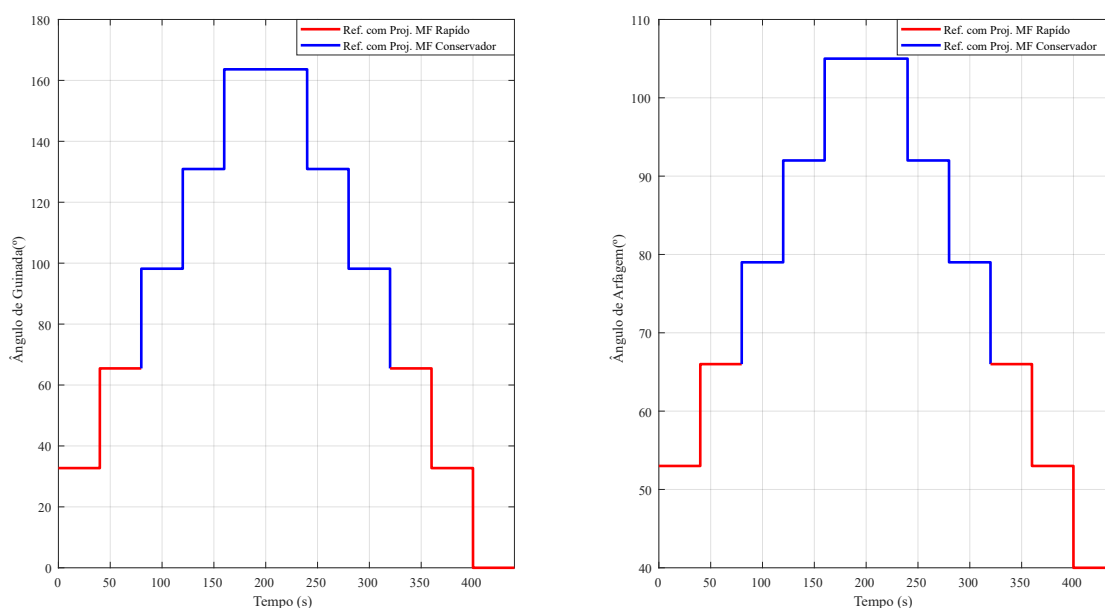


Figura 15 – Trajetória de Referência para os Experimentos no H2-DOF.

Fonte: Autor.

É possível observar na Figura 15, que a síntese da metodologia do controlador neural é realizada considerando, para as regiões de referências baixas (linha em vermelho), um projeto de sintonia do controlador com uma dinâmica de malha fechada mais rápida, pois nessas zonas o processo possui menos incertezas envolvidas, relacionadas a ventos externos e atritos mecânicos, que podem impactar de forma negativa e gerar a instabilidade do processo. Enquanto isso, para as referências mais altas (linha em azul), é escolhido um projeto de sintonia do controlador com uma dinâmica de malha fechada mais conservadora, devido à sua característica de maior robustez a perturbações de carga e ruídos de alta frequência, que são amplificados com o aumento das potências de incertezas, onde são observadas pelos resíduos do estimador MQR sobre tentativas de estimar e isolar um modelo melhor com o processo já controlado pelos controladores clássicos a parâmetros fixos, em altitudes maiores.

Por fim, uma ideia de estimativa média da trajetória das referências da Figura 15 pode ser verificada na Figura 16, como forma de elucidar a trajetória tridimensional

resultante do objetivo das referências, onde aplica-se uma propulsão ao rotor principal no instante 0 s, o ângulo de arfagem começa a subir, mas gerando um efeito parasitário no ângulo de guinada que também se eleva até próximo de 225 segundos. Após isso, verifica-se que ao retirar a propulsão do ângulo de arfagem, o ângulo de guinada também diminui, com ambos retornando a zero aos 450 segundos. Dessa forma, é possível verificar que a combinação média das referências do tipo degrau dos ângulos de arfagem e guinada tenta produzir uma trajetória helicoidal de ascensão e descensão (de forma espelhada), sendo a posição inicial dos ângulos (em 0°) relacionada à localização do nariz do H2-DOF até o seu centro de massa, que está próximo da haste que fixa o corpo do processo.

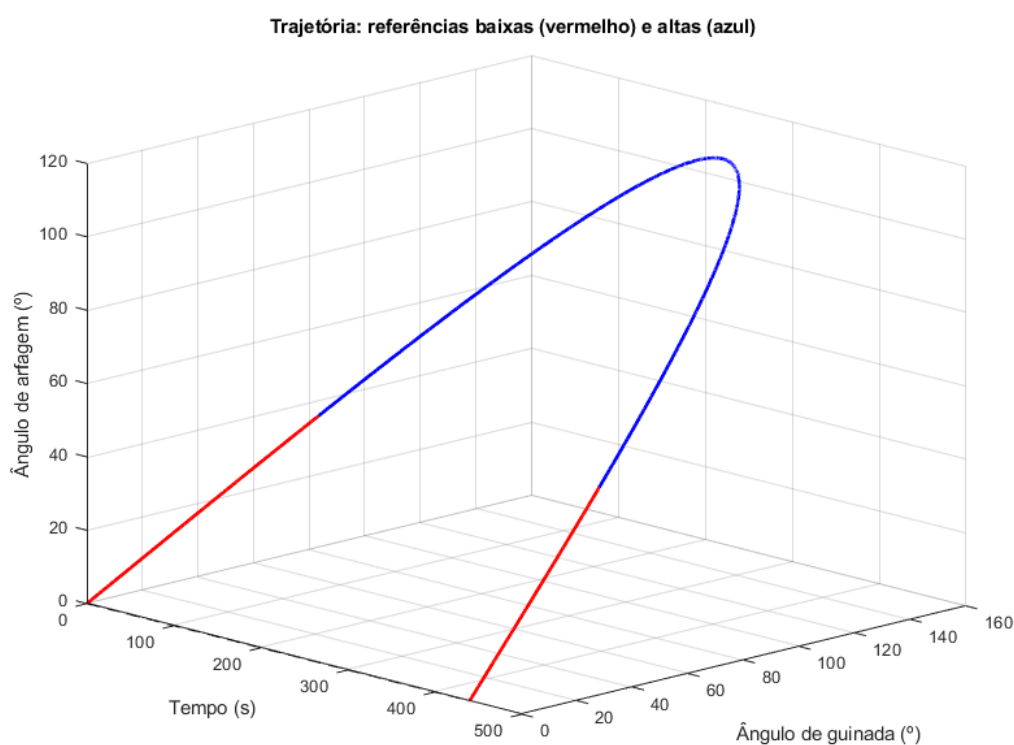


Figura 16 – Aproximação da Trajetória pela Combinação das Referências Angulares.

Fonte: Autor.

### 3.5 ANÁLISE DE DESEMPENHO E ROBUSTEZ

#### 3.5.1 Índices de Desempenho

Em projetos de controle, mensurar o desempenho do sistema por meio de critérios quantitativos é essencial para garantir que o comportamento obtido esteja em conformidade com os objetivos do projeto (Åström; Wittenmark, 2013). Nesse sentido,

índices específicos permitem avaliar a qualidade do seguimento da referência e o nível de esforço exigido pelo atuador, tornando-se ferramentas indispensáveis no processo de ajuste e validação do controlador (Araújo et al., 2017).

Dentre indicadores usados nesta pesquisa, a Integral do Erro Médio Quadrático (*ISE*), expressa acúmulo do erro ao longo do tempo dando uma noção da eficácia do controlador no seguimento das referências, enquanto a Integral do Sinal de Controle Médio Quadrático (*ISU*) representa uma expectativa média da energia necessária para que o sinal de controle  $u$  mantenha o seguimento das referências, onde ambos são definidos pelas seguintes expressões:

$$ISE = \frac{1}{N} \sum_{k=1}^N e(k)^2 \quad (115)$$

$$ISU = \frac{1}{N} \sum_{k=1}^N u(k)^2 \quad (116)$$

onde,  $N$  é a quantidade total de amostras no vetor de dados.

Além disso, métricas estatísticas como a variância dos sinais de erro ( $\sigma_e^2$ ) e controle ( $\sigma_u^2$ ) podem revelar o grau de variabilidade nas respostas do sistema, servindo como um complemento às análises tradicionais (Silveira et al., 2024).

$$\sigma_e^2 = \frac{1}{N} \sum_{k=1}^N [e(k) - \bar{e}]^2 \quad (117)$$

$$\sigma_u^2 = \frac{1}{N} \sum_{k=1}^N [u(k) - \bar{u}]^2 \quad (118)$$

em que  $\bar{e}$  e  $\bar{u}$  correspondem às médias dos sinais de erro e controle, respectivamente.

### 3.5.2 Índices de Robustez

A análise de robustez constitui uma etapa essencial na validação de projeto dos controladores destinados a operar em ambientes sujeitos a incertezas e perturbações externas, como ruídos, variações paramétricas e perturbações de carga. Nesse contexto, os indicadores comumente utilizados para produzir esta análise são a Margem de Ganho (*GM*) e a Margem de Fase (*PM*), que estão associadas à estabilidade relativa e à resistência do sistema frente a perturbações não modeladas. Em geral, valores mais elevados desses índices indicam uma maior robustez, embora possam também estar associados a respostas mais lentas (Silva et al., 2021; Silveira, 2024).

Desta maneira, as margens de ganho expressam a variação admissível no ganho de malha aberta antes que o sistema se torne instável, enquanto as margens de fase quantificam o atraso de fase máximo que o sistema suporta antes da perda

de estabilidade (Coelho; Jeronymo; Araujo, 2019). Ambas podem ser estimadas a partir da análise de duas funções de sensibilidade em malha fechada: a função de sensibilidade  $S_{sen}$  e a função de sensibilidade complementar  $T_{com}$ , definidas pelas expressões:

$$S_{sen}(z) = \frac{1}{1 + G_c(z)G_p(z)} \quad (119)$$

$$T_{com}(z) = \frac{G_c(z)G_p(z)}{1 + G_c(z)G_p(z)} \quad (120)$$

onde  $G_c(z)$  e  $G_p(z)$  representam, respectivamente, os modelos matemáticos do controlador e da planta no domínio discreto ( $z$ ).

Ademais, a curva  $S_{sen}$  descreve como o sistema reage a distúrbios aplicados à saída, enquanto  $T_{com}$  revela como o sistema responde a variações no sinal de referência (Stevens; Lewis; Johnson, 2016). A partir dessas curvas, é verificado o valor máximo dos picos de ressonância nas magnitudes dessas funções, pelas faixas de frequências apresentadas, descritos como:

$$M_S \triangleq \max_{\omega} \left| S_{sen}(e^{j\omega T_s}) \right| \quad (121)$$

$$M_T \triangleq \max_{\omega} \left| T_{com}(e^{j\omega T_s}) \right| \quad (122)$$

Quando se utiliza a estrutura RST no projeto do controlador, o índice  $M_S$  pode ser obtido também por meio de uma relação envolvendo os polinômios característicos do sistema, conforme expresso em (Araújo et al., 2017):

$$M_S \triangleq \max_{\omega} \left| \frac{A(e^{j\omega T_s})R(e^{j\omega T_s})}{A(e^{j\omega T_s})R(e^{j\omega T_s}) + B(e^{j\omega T_s})S(e^{j\omega T_s})} \right| \quad (123)$$

Adicionalmente, segundo Postlethwaite (1996), é possível inferir limites inferiores para  $GM$  e  $PM$  a partir dos valores de  $M_S$  e  $M_T$ , da seguinte forma:

$$GM \geq \min \left\{ 20 \log_{10} \left( \frac{M_S}{M_S - 1} \right), 20 \log_{10} \left( \frac{M_T + 1}{M_T} \right) \right\} \quad (124)$$

$$PM \geq \left( \frac{180}{\pi} \right) \min \left\{ 2 \sin^{-1} \left( \frac{1}{2M_S} \right), 2 \sin^{-1} \left( \frac{1}{2M_T} \right) \right\} \quad (125)$$

Essas expressões fornecem uma avaliação quantitativa e consistente da robustez dos controladores projetados. Para o caso estudado nesta pesquisa, após inúmeros testes e tentativas de controlar o processo, é adotado que as faixas das margens entre  $6dB \leq GM \leq 10dB$  e  $50^\circ \leq PM \leq 60^\circ$  são mais razoáveis e robustas, para que o controle do processo seja satisfatório e manejável nos seguimentos das referências, tendo dinâmicas conservadoras para tentar atenuar os ruídos ocasionados pela sensibilidade dos sensores das saídas.

## 4 SIMULAÇÕES E RESULTADOS

Neste capítulo são implementadas as técnicas de modelagem e controle apresentadas nos capítulos 2 e 3 no processo H2-DOF, de forma que primeiramente, é apresentado o experimento produzido em malha aberta para identificação e seleção do modelo linearizado, que é aproximado para uma dinâmica de primeira ordem, com o intuito de produzir simulações e projetar os controladores para as malhas de guinada ( $\psi$ ) e arfagem ( $\theta$ ) do processo real.

Na etapa seguinte, são estabelecidas as análises sobre as dinâmicas de malha fechada escolhidas para os projetos dos controladores clássicos (PI e PID) a parâmetros fixos e seus ganhos obtidos. Adicionalmente, é apresentada uma comparação breve das simulações em malha fechada, onde são realizados testes envolvendo a variação de referência e perturbação na saída, em comparação com os resultados nos experimentos realizados no processo real com a sintonia obtida pelas simulações.

Na próxima seção é mostrada a realização do controlador PI neural, com os dados obtidos nos experimentos reais dos projetos dos controladores clássicos PI a parâmetros fixos, avaliando as métricas de desempenho obtidas no treinamento da rede neural e os testes realizados no modelo simulado para depois aplicar no experimento real. E por fim, apresenta-se uma análise comparativa dos índices de desempenho para os controladores locais PI e PID, em relação ao controlador PI-Neural, com uma proposição de avaliação das margens de robustez do controlador PI neural devido ao autoajuste dos ganhos produzidos a cada iteração para as malhas de controle, que recai na situação de provas de estabilidade do sistema.

É válido destacar que todos os projetos são concebidos com as entradas e saídas consideradas como sinais de tensão, gerados e obtidos pelo microcontrolador ESP-32, que possui uma faixa de operação de escrita e leitura de 0V até 3,3V e que é convertida, ao fim de cada experimento nos controladores, para uma estimativa angular descrita pelos limites operacionais presentes no processo H2-DOF da seção 2.1.

### 4.1 IDENTIFICAÇÃO

O modelo completo do sistema é identificado a partir de dados produzidos no processo real apresentado na Figura 17, onde é considerado um tempo de amostragem ( $T_s$ ) de 0,05 segundos, escolhido por tentativa e erro, para um teste em malha aberta de 60 segundos, com diferentes sinais de entrada, sendo aplicado um sinal triangular na entrada  $u_I$  (relacionado ao rotor da cauda) e um sinal senoidal para a entrada  $u_V$  (relacionado ao rotor principal).

Observa-se que o sistema tem um certo atraso para começar a produzir variações na saída de  $\psi$ , haja visto que o rotor principal responsável pela ação da saída de  $\theta$  produz uma força que rotaciona o corpo no sentido anti-horário em relação ao

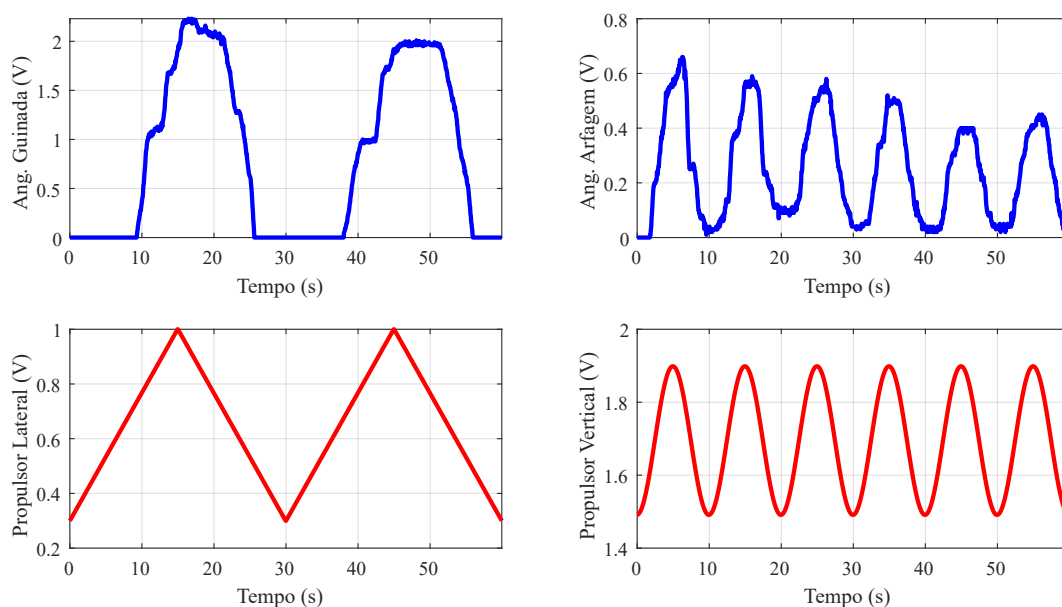


Figura 17 – Ensaio em Malha Aberta.

Fonte: Autor.

eixo Z. Assim, como é verificado em aproximadamente 10 segundos do experimento, quando houve um aumento contínuo da tensão aplicada no propulsor da cauda  $u_l$  e a diminuição contínua em  $u_v$ , que gerou uma força suficiente no sentido horário para produzir  $\psi$ .

Além disso, é possível visualizar de uma forma mais nítida o efeito de *cross-coupling* e as não-linearidades do sistema, entre 40 a 55 segundos, onde há uma maior contribuição das influências geradas entre os estados, quando o valor de  $u_l$  está alto (entre 0,8 e 1 V) e  $u_v$  começa a aumentar a tensão do ponto de operação mínimo até o máximo (de 1,5 até 1,9 V).

Portanto, esse ensaio é utilizado para a aplicação da estimação, com o objetivo de produzir um modelo médio linearizado, que possa conter as dinâmicas do processo real e auxiliar no projeto dos controladores.

#### 4.1.1 Análise de Modelagem

Como o experimento da Figura 17 possui não linearidades causadas pelos acoplamentos dinâmicos, ruídos de leitura e a diferença nas faixas de operação máximas entre as malhas de  $\psi$  e  $\theta$ , são usados sinais periódicos na entrada para tentar captar relações dinâmicas entre os estados do processo em baixas frequências. Entretanto, por conta dessas condições adversas, os sinais devem ser tratados para se evitar tendências que possam polarizar o estimador MQR, sendo necessário aplicar um tratamento para diminuir essas tendências nos dados. Assim, o intuito do tratamento, nos sinais

de entrada e de saída, é remover as tendências dos dados na tentativa de melhorar a estimação dos estados, de forma que, ao aplicar-se o algoritmo de identificação, os novos dados tratados tenham uma média zero para evitar uma convergência desigual na estimação dos estados, ao aplicar a equação (126) e gerar os dados tratados na Figura 18 (Aguirre, 2007).

$$\begin{cases} x_n(k) = x_n(k-1) - \bar{x}_n \\ u_l(k) = u_l(k-1) - \bar{u}_l \end{cases} \quad (126)$$

onde,  $\bar{x}_n$  e  $\bar{u}_l$  são as médias, respectivamente, dos valores do vetor de estado e da entrada.

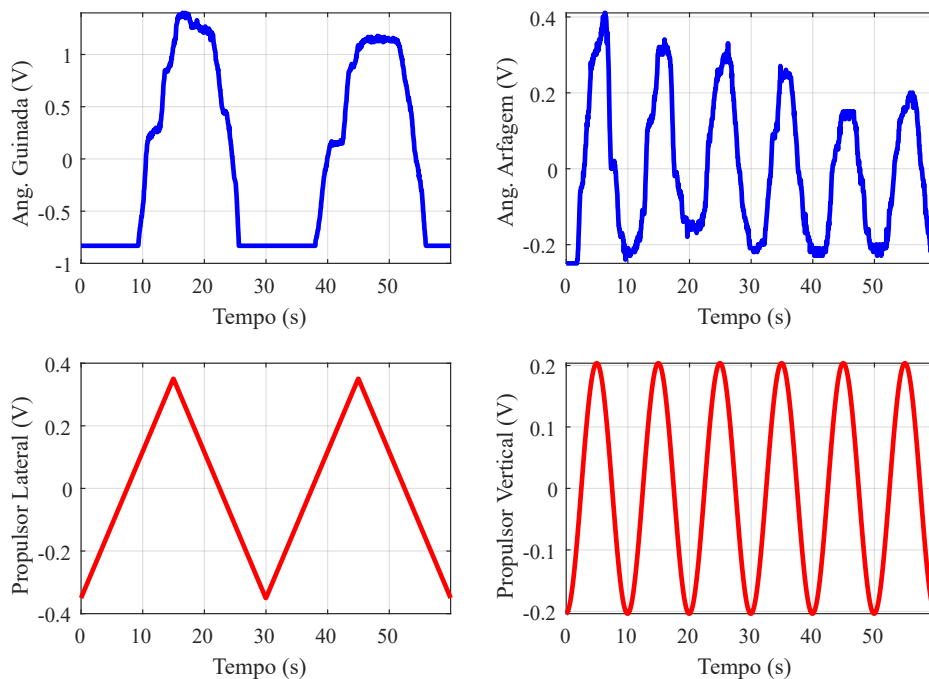


Figura 18 – Tratamento dos Dados do Ensaio em Malha Aberta.

Fonte: Autor.

Desta maneira, o sistema completo em (127) e (128) é modelado pela aproximação em espaço de estados utilizando a teoria de estimação do MQR apresentada na subseção 2.2. O modelo é produzido ao utilizar todo o conjunto de dados presente na Figura 18, em que a seleção do melhor modelo estimado é feita por uma abordagem de verificação em blocos do vetor das matrizes  $A_{est}$ ,  $B_{est}$  e  $\Gamma_{est}$  produzidas a partir do vetor de estimação  $\hat{\theta}$  gerado a cada iteração  $k$ , como descrito na subseção 2.3.

$$\begin{aligned}
\underbrace{\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix}}_{\mathbf{x}(k)} &= \underbrace{\begin{bmatrix} 0,98693 & 0,00680 & 0,01817 & 0,01255 \\ 0,00251 & 0,92555 & 0,00574 & -0,00428 \\ -0,16821 & 0,41685 & 0,43939 & 0,34065 \\ 0,05298 & -1,15616 & 0,12515 & -0,01020 \end{bmatrix}}_{\hat{A}} \underbrace{\begin{bmatrix} x_1(k-1) \\ x_2(k-1) \\ x_3(k-1) \\ x_4(k-1) \end{bmatrix}}_{\mathbf{x}(k-1)} \\
&+ \underbrace{\begin{bmatrix} 0,05586 & -0,03637 \\ -0,00944 & 0,11051 \\ 0,80150 & -1,06704 \\ -0,20499 & 1,77447 \end{bmatrix}}_{\hat{B}} \underbrace{\begin{bmatrix} u_1(k-1) \\ u_2(k-1) \end{bmatrix}}_{\mathbf{u}(k-1)} \\
&+ \underbrace{\begin{bmatrix} 0,00374 & 0,00599 & -0,01101 & -0,01207 \\ -0,01709 & 0,08964 & -0,00304 & -0,00583 \\ -0,09341 & 0,35006 & -0,26538 & -0,34307 \\ -0,41855 & 2,28729 & -0,07562 & -0,20413 \end{bmatrix}}_{\hat{F}} \underbrace{\begin{bmatrix} \hat{w}_1(k-1) \\ \hat{w}_2(k-1) \\ \hat{w}_3(k-1) \\ \hat{w}_4(k-1) \end{bmatrix}}_{\hat{\mathbf{w}}(k-1)}
\end{aligned} \tag{127}$$

$$\underbrace{\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix}}_{\mathbf{y}(k)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}}_{\hat{C}} \underbrace{\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix}}_{\mathbf{x}(k)} + \underbrace{\begin{bmatrix} \hat{v}_1(k) \\ \hat{v}_2(k) \end{bmatrix}}_{\hat{\mathbf{v}}(k)} \tag{128}$$

$$\sigma_{\hat{\mathbf{w}}}^2 = \begin{bmatrix} 0,00146 \\ 0,00027 \\ 0,14741 \\ 0,06085 \end{bmatrix}; \sigma_{\hat{\mathbf{v}}}^2 = \begin{bmatrix} 0,00146 \\ 0,00027 \end{bmatrix} \tag{129}$$

Os estados  $x_1$ ,  $x_2$ ,  $x_3$  e  $x_4$  no processo H2-DOF são, respectivamente, o ângulo de guinada ( $\psi$ ), o ângulo de arfagem ( $\theta$ ), a derivada discreta do ângulo de guinada ( $\psi_{spd}$ ) e a derivada discreta do ângulo de arfagem ( $\theta_{spd}$ ).

Portanto, o algoritmo busca os melhores parâmetros pelo valor mínimo do erro médio quadrático na função custo  $J_{MSE}$  em (21), onde essa operação é feita conferindo os testes realizados nos modelos estimados com as entradas do ensaio tratado da Figura 18, para selecionar o melhor modelo que se aproxima da dinâmica do processo H2-DOF, ilustrado na Figura 19 com os índices de identificação na Tabela 1.

É importante destacar na Tabela 1, que a **média do sistema** de  $J_{MSE}$  é a soma das médias das malhas do processo, sendo o parâmetro global usado na seleção do melhor modelo estimado dentre todos os possíveis modelos gerados por  $\hat{\theta}$  em (20),

enquanto para o caso de  $R^2$  é de fato uma média nos valores obtidos nas malhas locais.

Tabela 1 – Índices de Identificação do  $MQR_{SS}$  por malha do processo.

Malha do Processo	$MQR_{SS}$	
	$R^2$	$J_{MSE}$
$\psi$	0,89432	46,55199
$\theta$	0,89520	2,18334
<b>Média do Sistema</b>	<b>0,89476</b>	<b>48,73533</b>

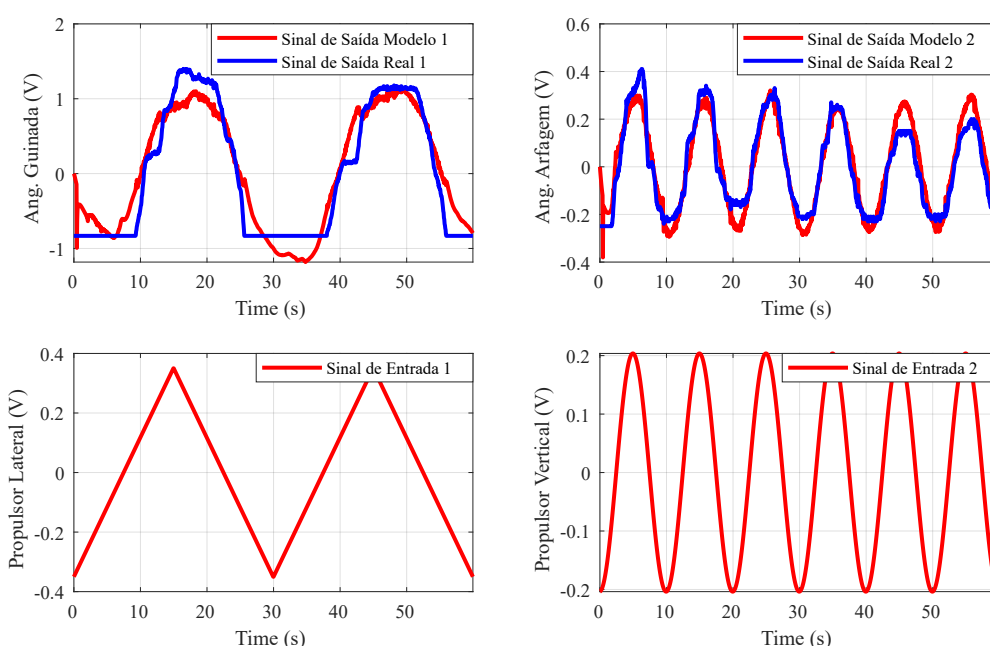


Figura 19 – Resultado Comparativo da Saída Simulada pelo Melhor Modelo Estimado.

Fonte: Autor.

Ademais, para a etapa de validação, é utilizado um novo conjunto de dados obtidos em um experimento de 30 segundos no processo real, com as mesmas formas de entradas usadas no conjunto de estimação da Figura 18, para averiguar se o modelo estimado possui uma relação dinâmica próxima aos estados do ângulo de arfagem e do ângulo de guinada, conforme a Figura 20 e a Tabela 2.

Portanto, para a estrutura de projeto descentralizado com controladores locais da Figura 11, são consideradas apenas as matrizes estimadas  $\hat{A}$ ,  $\hat{B}$  e  $\hat{v}$ . A matriz dos ruídos estimados do processo ( $\hat{r}$ ), que está expressa na equação (127), é mantida na etapa de identificação como uma contribuição complementar para o uso em trabalhos futuros envolvendo o sistema H2-DOF, para que possam usufruir de um modelo completo que é **observável e controlável** (como descrito em 3.3.1.1), em simulações e

Tabela 2 – Índices de Validação do  $MQR_{SS}$  por malha do processo.

Malha do Processo	$MQR_{SS}$	
	$R^2$	$J_{MSE}$
$\psi$	0,79665	39,77452
$\theta$	0,93750	0,58330
<b>Média do Sistema</b>	<b>0,86708</b>	<b>40,35782</b>

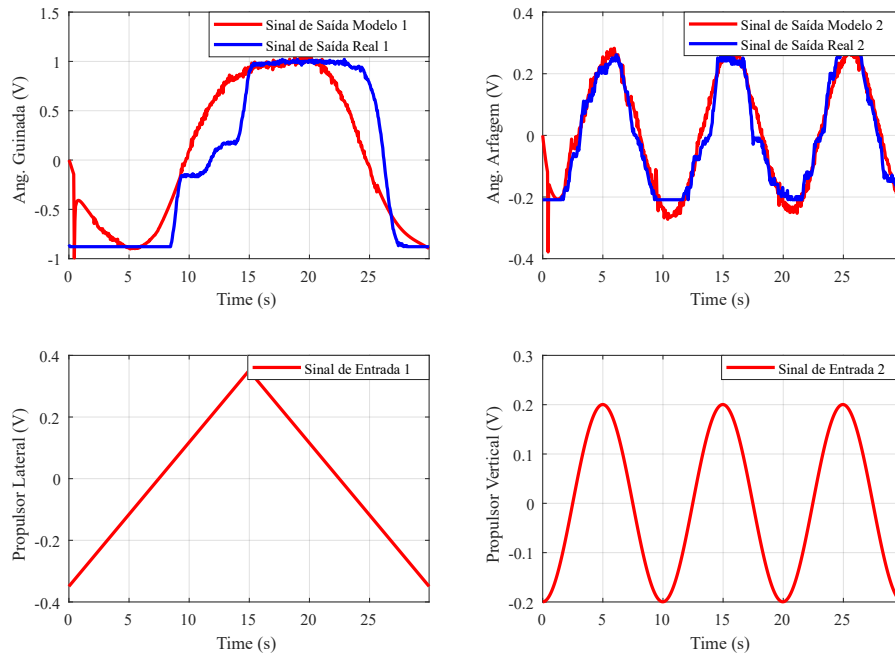


Figura 20 – Validação do Melhor Modelo Estimado com Novos Dados do H2-DOF.

Fonte: Autor.

projetos com estratégias de controle mais sofisticadas e centralizadas, como controladores lineares quadráticos para caso regulatório (LQR) ou com o uso do filtro Kalman para ruídos Gaussianos (LQG).

#### 4.1.2 Modelo na Forma Polinomial

Para projetar os controladores, são aplicadas as considerações da subseção 3.2.1.2 no sistema modelado, onde são extraídas quatro funções de transferência em (130) a (133), que compõem a matriz de funções de transferências  $\hat{G}_p(z)$  do modelo estimado na forma polinomial em (134), ao aplicar-se a equação (47). Desta forma, o método RGA em (49), gera a matriz resultante das relações que as influências das entradas têm nas saídas em (135).

$$\hat{G}_{p_{11}}(z) = \frac{0,00139z^{-1} + 0,00946z^{-2} - 0,06351z^{-3} + 0,05560z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (130)$$

$$\hat{G}_{p_{21}}(z) = \frac{0,00074z^{-1} - 0,01038z^{-2} + 0,01927z^{-3} - 0,00960z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (131)$$

$$\hat{G}_{p_{12}}(z) = \frac{-0,00106z^{-1} - 0,01708z^{-2} + 0,05283z^{-3} - 0,03620z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (132)$$

$$\hat{G}_{p_{22}}(z) = \frac{-0,00216z^{-1} + 0,06311z^{-2} - 0,17070z^{-3} + 0,11070z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (133)$$

$$\hat{G}_p(z) = \begin{bmatrix} \hat{G}_{p_{11}}(z) & \hat{G}_{p_{12}}(z) \\ \hat{G}_{p_{21}}(z) & \hat{G}_{p_{22}}(z) \end{bmatrix} \quad (134)$$

$$f_{RGA}(\hat{G}_p) = \Lambda(\hat{G}_p) = \begin{bmatrix} 0,9868 & 0,0132 \\ 0,0132 & 0,9868 \end{bmatrix} \quad (135)$$

É possível verificar, pela diagonal secundária em (135), que o sistema multi-variável apresenta uma baixa influência cruzada da entrada  $u_1$  sobre a saída  $y_2$ , e vice-versa. No entanto, caso essas influências não sejam consideradas previamente no projeto dos controladores, elas podem aumentar a complexidade do controle do processo e, dependendo da magnitude das entradas e incertezas, até mesmo comprometer a estabilidade do sistema em malha fechada.

Assim, para reduzir esses efeitos e corrigir a resposta estática do sistema, adotou-se as equações de pré-compensação em (51) e (52) para se aplicar no sistema modelado, em que é verificado o ganho DC do sistema estimado pela matriz escalar em (136) para calcular a pré-compensação ( $P_c$ ) em (137).

$$K_s = \begin{bmatrix} 4,47200 & -2,29090 \\ 0,03620 & 1,38920 \end{bmatrix} \quad (136)$$

$$P_c = \begin{bmatrix} 0,22070 & 0,36390 \\ -0,00580 & 0,71030 \end{bmatrix} \quad (137)$$

Em seguida, é aplicada a matriz de pré-compensação da equação (137) na entrada do sistema estimado em (127), onde é feita a correção dos pesos da matriz de entrada  $\hat{B}$  para se obter o sistema estimado pré-compensado convertido para forma polinomial em (142), pelas equações (53) a (55), descrito pelas funções de transferência mostradas a seguir:

$$\hat{G}_{pc_{11}}(z) = \frac{0,00031z^{-1} + 0,00219z^{-2} - 0,01432z^{-3} + 0,01248z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (138)$$

$$\hat{G}_{pc_{21}}(z) = \frac{0,00018z^{-1} - 0,00265z^{-2} + 0,00523z^{-3} - 0,00276z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (139)$$

$$\hat{G}_{pc_{12}}(z) = \frac{-0,00025z^{-1} - 0,00869z^{-2} + 0,01442z^{-3} - 0,00548z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (140)$$

$$\hat{G}_{pc_{22}}(z) = \frac{-0,00127z^{-1} + 0,04105z^{-2} - 0,1143z^{-3} + 0,07514z^{-4}}{1 - 2,34200z^{-1} + 1,68300z^{-2} - 0,29200z^{-3} - 0,04799z^{-4}} \quad (141)$$

$$\hat{G}_{pc}(z) = \begin{bmatrix} \hat{G}_{pc_{11}}(z) & \hat{G}_{pc_{12}}(z) \\ \hat{G}_{pc_{21}}(z) & \hat{G}_{pc_{22}}(z) \end{bmatrix} \quad (142)$$

$$f_{RGA}(\hat{G}_p) = \Lambda(\hat{G}_p) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (143)$$

A Figura 21 apresenta uma breve comparação da resposta ao degrau no sistema multivariável em malha aberta, antes e depois da aplicação da correção do ganho estático do sistema com  $P_c$ , onde as curvas em **vermelho** representam o comportamento do *sistema estimado original*  $\hat{G}_p(z)$  em (134), enquanto as curvas em **azul** correspondem ao *sistema pré-compensado*  $\hat{G}_{pc}(z)$  em (142).

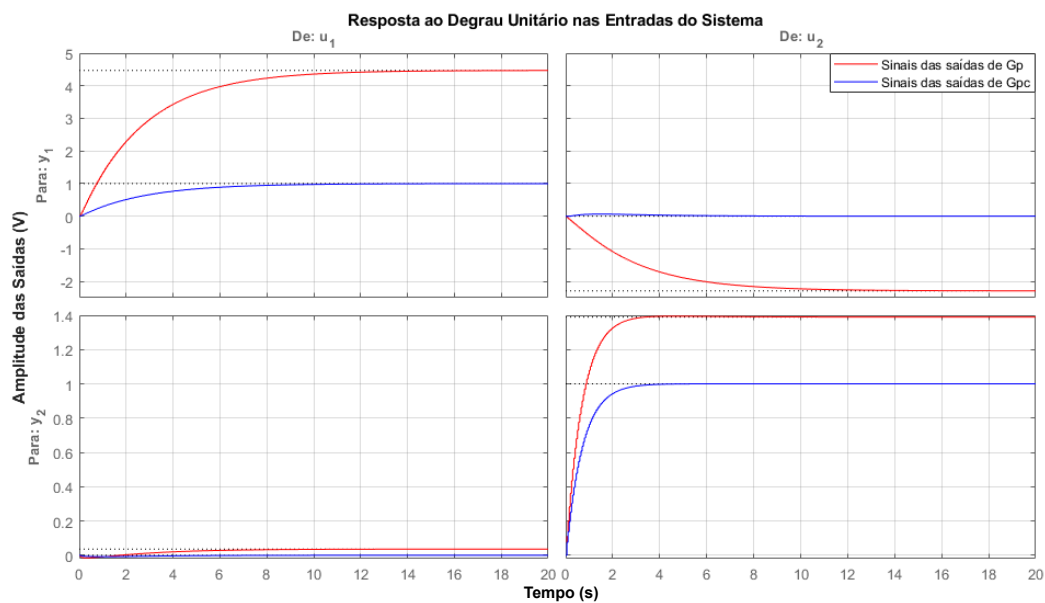


Figura 21 – Comparação da resposta ao degrau no sistema MIMO.

Fonte: Autor.

Observa-se que as saídas associadas à **diagonal principal** da matriz de transferência do sistema pré-compensado ( $G_{pc_{11}}(z)$  e  $G_{pc_{22}}(z)$ ) passam a responder proporcionalmente às entradas do tipo *degrau unitário*, enquanto as respostas cruzadas,

associadas à **diagonal secundária** ( $G_{pc_{12}}(z)$  e  $G_{pc_{21}}(z)$ ) são praticamente anuladas. Isso evidencia que a ação da pré-compensação torna o sistema aproximadamente desacoplado, facilitando a posterior síntese dos controladores para cada malha individualmente.

Desta maneira,  $G_{pc_{11}}(z)$  e  $G_{pc_{22}}(z)$  são usadas para conceber a síntese dos controladores locais nessa pesquisa, de forma que aplicam-se as aproximações de primeira ordem, pelas equações descritas em (56) e (57), para gerar as dinâmicas discretas aproximadas  $\psi$  e  $\theta$  para as malhas descentralizadas, na estrutura ARX em (144) e (144).

$$G_{i_1}(z) = \frac{0,01798z^{-1}}{1 - 0,98200z^{-1}} \quad (144)$$

$$G_{i_2}(z) = \frac{0,06887z^{-1}}{1 - 0,93110z^{-1}} \quad (145)$$

Além disso, as aproximações das dinâmicas feitas em (144) e (145) são adotadas para o projeto, pois é realizada uma verificação da dinâmica, em malha aberta, pelo diagrama de Bode da Figura 22, onde é possível observar que as reduções de ordem aplicadas nas funções de transferência, em (138) e (141), não apresentam mudanças dinâmicas drásticas nas faixas de frequência em que os controladores são projetados.

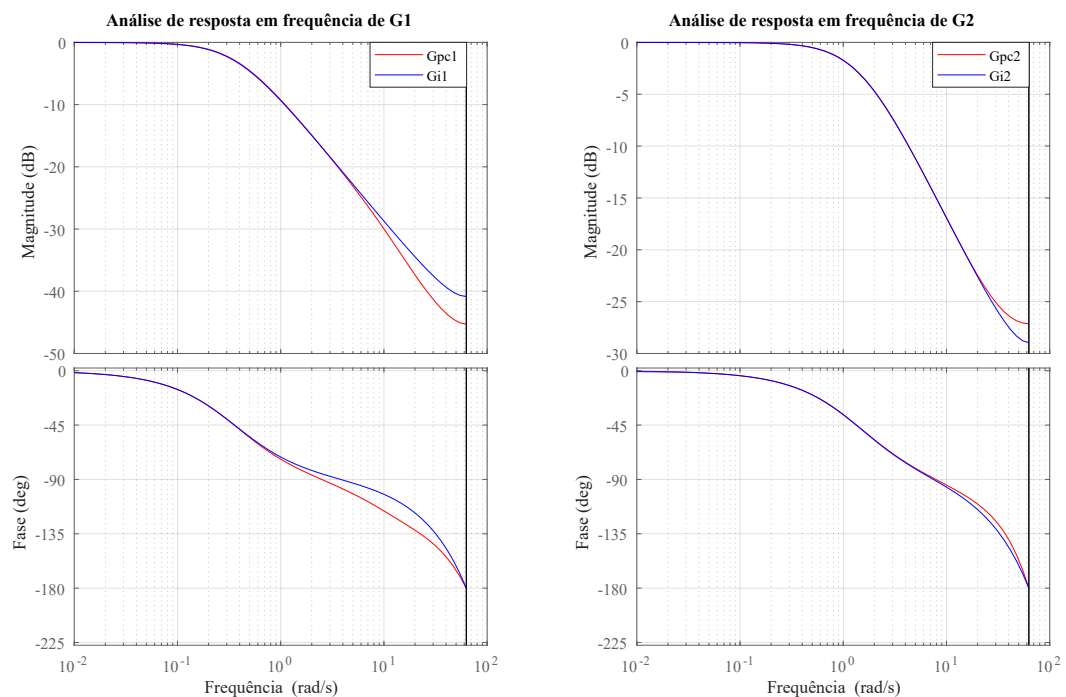


Figura 22 – Análise de resposta em frequência.

Fonte: Autor.

## 4.2 CONTROLE

Os resultados de controle são apresentados com testes simulados e reais do processo H2-DOF, onde os testes reais são implementados com auxílio do software *Daquino* para comunicação serial entre os softwares Arduino e Matlab, com taxa de transmissão de 9600 bits/s e o ESP-32 como controlador. Enquanto isso, para os testes simulados, são produzidas análises sobre a dinâmica do sistema aproximado de primeira ordem e a escolha dos parâmetros desejados de malha fechada, para sintonizar os ganhos dos controladores PI e PID, como descritos nas considerações dos projetos nas subseções 3.3.1 a 3.3.3.

Para condições de testes simulados e reais, são considerados 6 sinais de referência do tipo degrau, com duração de 40 segundos cada, para as malhas de  $\psi$  e  $\theta$  para emular uma trajetória de ascensão e descensão, como descrito ao final da seção 3.4. Entretanto, esses sinais de referência têm amplitudes diferentes para cada malha, devido aos limites físicos mencionados na seção 2.1, sendo a amplitude para  $\psi$  entre 0 V a 3 V (em passos de 0,6 V) e para  $\theta$  entre 0 V a 1 V (em passos de 0,2 V). Então, em termos práticos, os ângulos de referência da Figura 15 são gerados a partir dos limites dos sinais de tensão, da seguinte maneira:

$$\psi[^{\circ}] = \frac{180^{\circ} - 0^{\circ}}{3,3\text{ V} - 0\text{ V}} \cdot \psi[\text{V}] = 54,545 \frac{[^{\circ}]}{[\text{V}]} \cdot \psi[\text{V}] \quad (146)$$

$$\theta[^{\circ}] = \frac{105^{\circ} - 40^{\circ}}{1\text{ V} - 0\text{ V}} \cdot \theta[\text{V}] + 40^{\circ} = 65 \frac{[^{\circ}]}{[\text{V}]} \cdot \theta[\text{V}] + 40^{\circ} \quad (147)$$

onde,  $\psi[\text{V}]$  e  $\theta[\text{V}]$  são os sinais de tensão obtidos pela posição dos potenciômetros no H2-DOF.

É notório destacar em (146) e (147), que  $\psi[\text{V}]$  possui a faixa de leitura igual a do microcontrolador (de 0 V a 3,3 V), apesar do processo conseguir gerar ângulos maiores de  $\psi$  que são saturados pelo microcontrolador, enquanto para  $\theta[\text{V}]$  é considerada uma faixa menor (de 0 V a 1 V), devido a limitações práticas relacionadas a dinâmica do corpo do sistema, onde a Figura 23 apresenta a estimativa de conversão dos sinais da saída.

Além disso, são usadas perturbações pelo ruído branco Gaussiano na saída do sensor nas simulações, baseadas na variância de  $\hat{v}$  em (129), obtida no modelo estimado em (128), para averiguar a robustez do projeto em mitigar esses efeitos, enquanto seguem as referências. Desta maneira, o intuito é verificar se o projeto que é usado no processo real irá instabilizar facilmente, devido a incertezas de modelo, não-linearidades e dinâmicas não modeladas. Logo, após essa etapa, com os resultados simulados, aplicam-se os ganhos do projeto no H2-DOF para testar se é possível controlar com base no projeto das simulações.

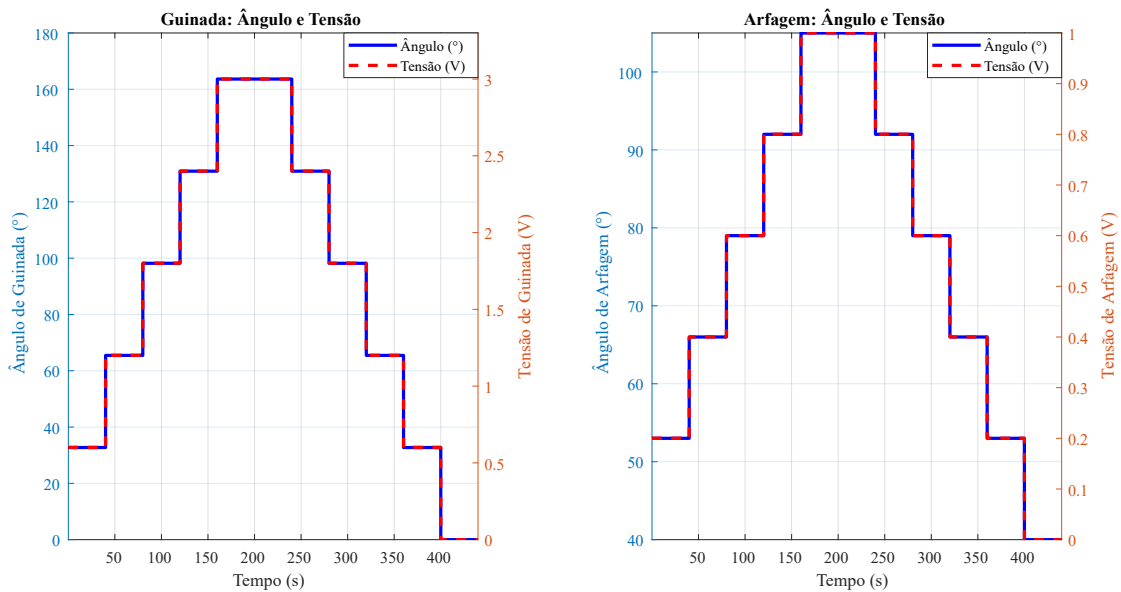


Figura 23 – Estimativa de Conversão dos Sinais de Saída de Tensão para Ângulo.

Fonte: Autor.

Posteriormente, o treinamento da rede neural é realizado com o uso dos dados do sistema real com os controladores PI, como descrito na seção 3.4, com o intuito de aplicar em teste em ambiente simulado e real para se produzir análises comparativas dos resultados obtidos entre os controladores projetados nesta pesquisa.

Todos os testes mencionados são produzidos com o uso de um computador, que possui um processador Intel i5 12500H (4,5 GHz), uma placa de vídeo RTX 3050 (4GB de VRAM) e 16GB de memória RAM.

#### 4.2.1 Análise de Projeto dos Controladores PID

Como critério de sintonia, conforme descrito na subseção 3.2.2, são adotadas as especificações de projeto para (66), como o tempo de subida ( $T_r$ ) e a tolerância de sobressinal ( $M_p$ ), para gerar as dinâmicas de malha fechada desejadas em (67) e (68), para síntese dos controladores PID que controlam as malhas de guinada ( $\psi$ ) e arfagem ( $\theta$ ). Na Tabela 3 são mostrados os valores utilizados.

Tabela 3 – Parâmetros Desejados para Sintonia dos Controladores PID.

Funções de Trânsferências	$T_r(s)$	$M_p$	$\zeta$	$\omega_n(rad/s)$	$\alpha$
$\hat{G}_{mf_1}(\psi)$	9,07500	1%	0,82608	0,49723	$9,81982 \times 10^5$
$\hat{G}_{mf_2}(\theta)$	3,87500	1%	0,82608	1,16448	$9,31062 \times 10^4$

As escolhas feitas das especificações de projeto ( $T_r$  e  $M_p$ ) na Tabela 3, para gerar as dinâmicas desejadas, são baseadas nos valores dos parâmetros observados na

resposta ao degrau unitário de  $G_{i1}$  e  $G_{i2}$  em malha aberta na Figura 24, onde também é possível verificar a dinâmica desejada de segunda ordem discretizada produzida pelas especificações de projeto.

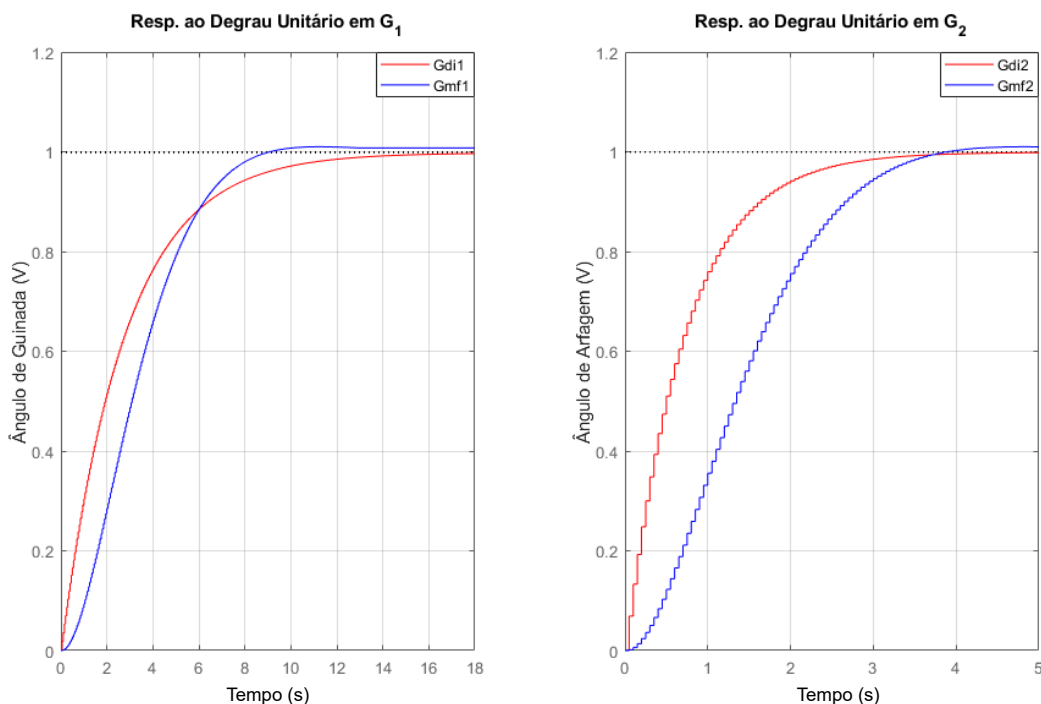


Figura 24 – Resposta ao Degrau em Malha Aberta da Planta e da Dinâmica das Especificações Desejada.

Fonte: Autor.

É importante destacar na Figura 24 que as dinâmicas desejadas possuem características mais conservadoras que as dinâmicas das plantas, devido a várias tentativas de projeto de controladores com dinâmicas mais rápidas gerarem instabilidade quando aplicadas no experimento real, principalmente, para a malha de  $\theta$  que é mais suscetível a instabilizar devido às não-linearidades presentes no sistema H2-DOF e que não foram captadas na etapa de identificação.

A estratégia de controle PID apresentou os resultados de simulação e no experimento real, com rastreamento de referência e sinal de controle, ilustrados nas Figuras 25 e 26. Observa-se que as saídas acompanharam adequadamente as referências nas simulações, mesmo diante da presença de ruídos Gaussianos de variância de  $\sigma_v^2$  em (129), enquanto no experimento real verificou-se um desafio maior para se controlar as malhas, onde houve fortes variações na malha que produz os ângulos de arfagem ( $\theta$ ). A Figura 27 exhibe as curvas das funções de sensibilidade do projeto dos controladores PID com o modelo estimado do H2-DOF. Já na Tabela 4 são apresentados os índices de desempenho e robustez ao longo das iterações dos experimentos simulado e real.

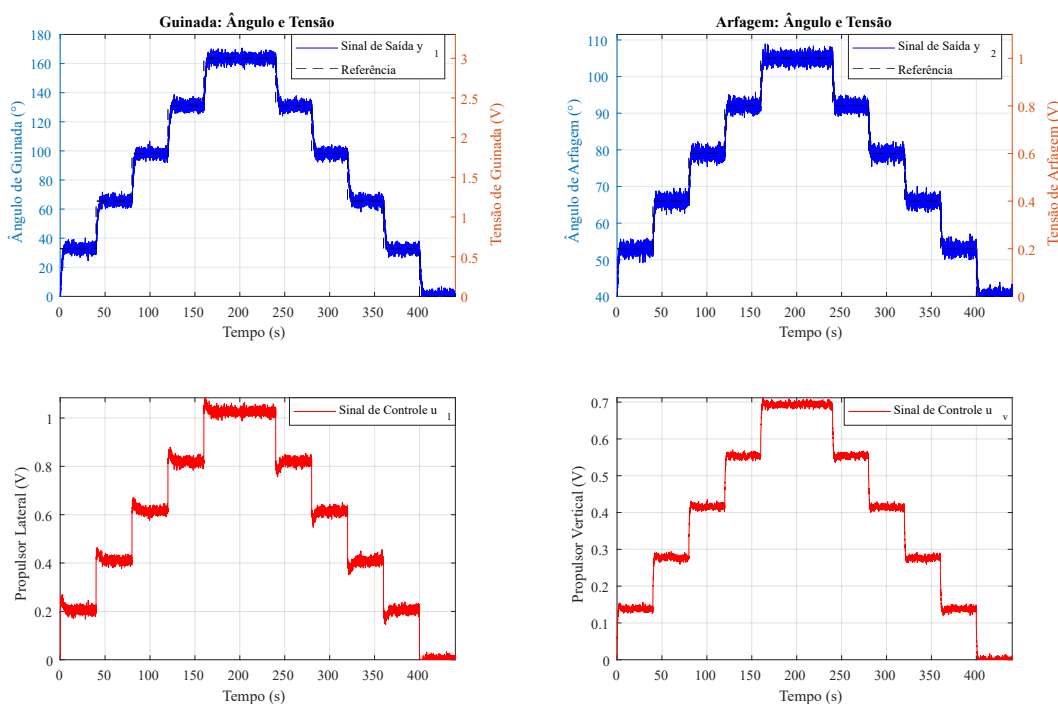


Figura 25 – Resultados Obtidos via Simulação com a Sintonia dos Controladores PID no Modelo Estimado.

Fonte: Autor.

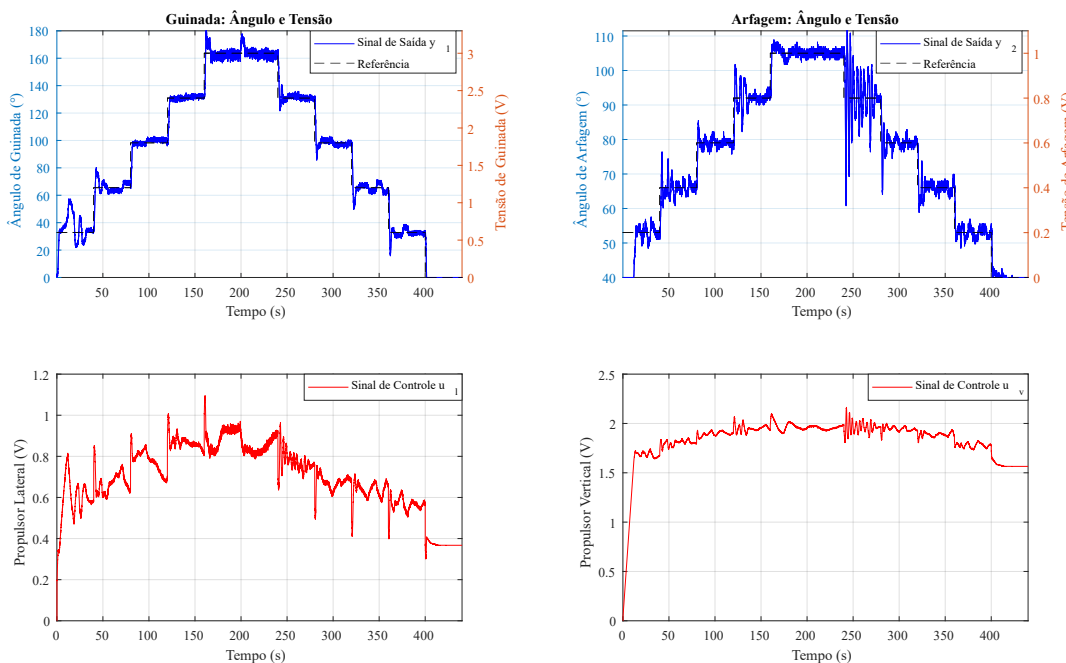


Figura 26 – Resultados Obtidos via Experimento Real com a Sintonia dos Controladores PID no Protótipo H2-DOF.

Fonte: Autor.

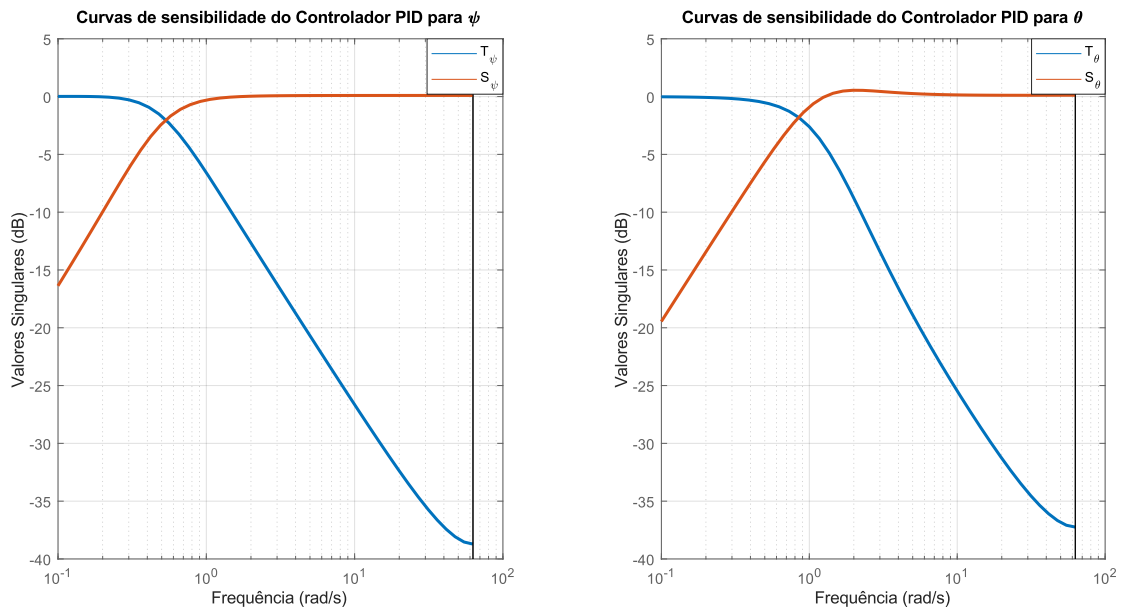


Figura 27 – Curvas de Sensibilidade dos Projetos dos Controladores PID.

Fonte: Autor.

Tabela 4 – Índices dos Controladores Obtidos nos Experimentos Simulado e Real

Índices dos controladores PID	Experimento Simulado		Experimento Real	
	$PID_{\psi}$	$PID_{\theta}$	$PID_{\psi}$	$PID_{\theta}$
$K_p$	1,23644	0,33190	1,23644	0,33190
$K_i$	0,67368	0,93917	0,67368	0,93917
$K_d$	0,00026	0,00061	0,00026	0,00061
ISE	0,00934	0,00092	0,01296	0,00409
ISU	0,42184	0,19211	0,50984	3,40629
$\sigma_{\theta}^2$	0,00934	0,00092	0,01292	0,00406
$\sigma_u^2$	0,10886	0,04924	0,02550	0,04950
GM(dB)	6,01248	6,02069	6,01248	6,02069
PM(°)	59,24300	56,00155	59,24300	56,00155

Na Figura 26 é possível verificar o momento em que mais ocorrem as oscilações, relacionadas às duas primeiras referências para a etapa de descensão, onde estas regiões são as mais desafiadoras de serem controladas pela estratégia PID. Destarte, é realizado mais um projeto PID com dinâmicas mais lentas que as descritas na Tabela 3 e na Figura 24, para verificar se uma nova sintonia poderia reduzir as oscilações de forma global no experimento real e melhorar o controle nessas regiões mais desafiadoras.

A Tabela 5 mostra a escolha dos novos valores dos parâmetros desejados, onde é possível averiguar o comportamento da dinâmica desejada de segunda ordem na

Figura 28, seguindo a mesma lógica usada no projeto anterior.

Tabela 5 – Parâmetros Desejados Reajustados para nova sintonia dos Controladores PID

Funções de Trânsferências	$T_r(s)$	$M_p$	$\zeta$	$\omega_n(rad/s)$	$\alpha$
$G_{mf_1}(\psi)$	12,01000	1,5%	0,80075	0,34484	$9,81982 \times 10^5$
$G_{mf_2}(\theta)$	4,64500	1,5%	0,80075	0,89732	$9,31062 \times 10^4$

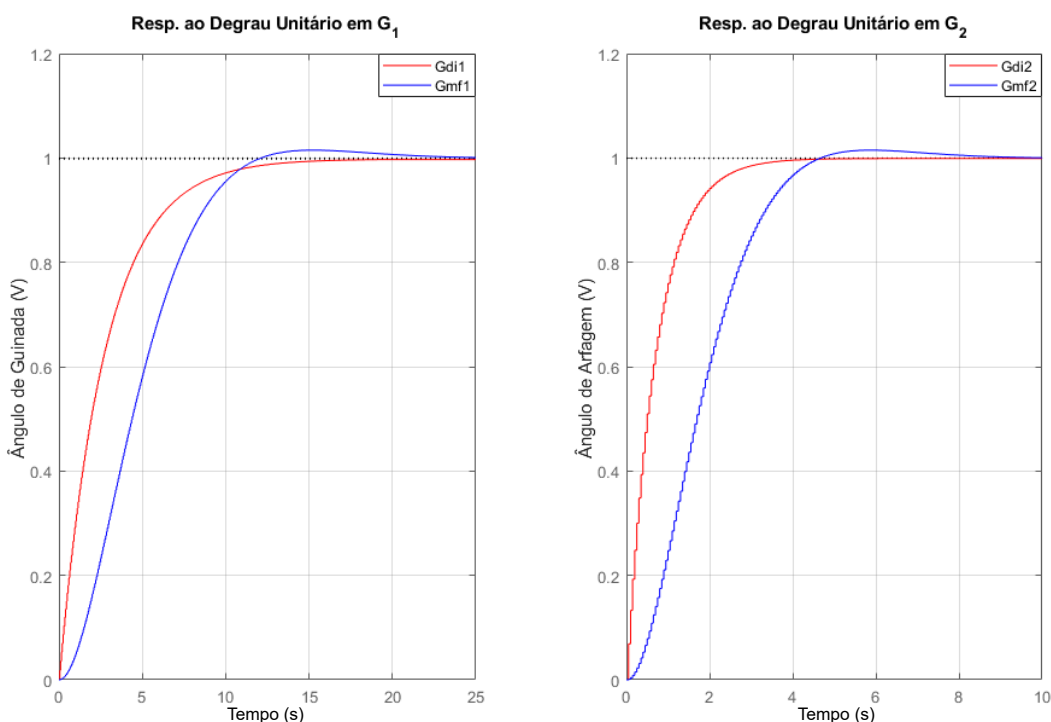


Figura 28 – Resposta ao degrau da planta e da dinâmica desejada reajustada.

Fonte: Autor.

Desta maneira, os resultados de reprojeto dos novos controles PID dos experimentos simulado e real estão expressos nas Figuras 29 e 30. A Figura 31 apresenta as novas curvas de sensibilidades que são geradas pelo ajuste dos parâmetros da Tabela 5, enquanto a Tabela 6 apresenta os índices resultantes da nova sintonia adotada.

Os resultados da Tabela 6 indicam que a nova sintonia teve uma piora de desempenho para controlar a malha de arfagem, em relação ao aspecto oscilatório de forma global, em que a primeira referência da etapa de descensão teve uma degradação maior, e que pode ser explorada em trabalhos futuros com a mudança da estrutura PID paralelo para estrutura I+PD, que poderia amenizar problemas ocasionados pela parte derivativa, que comumente é prejudicada na presença de ruídos do processo amplificarem impulsos desnecessários na ação de controle dessa topologia.

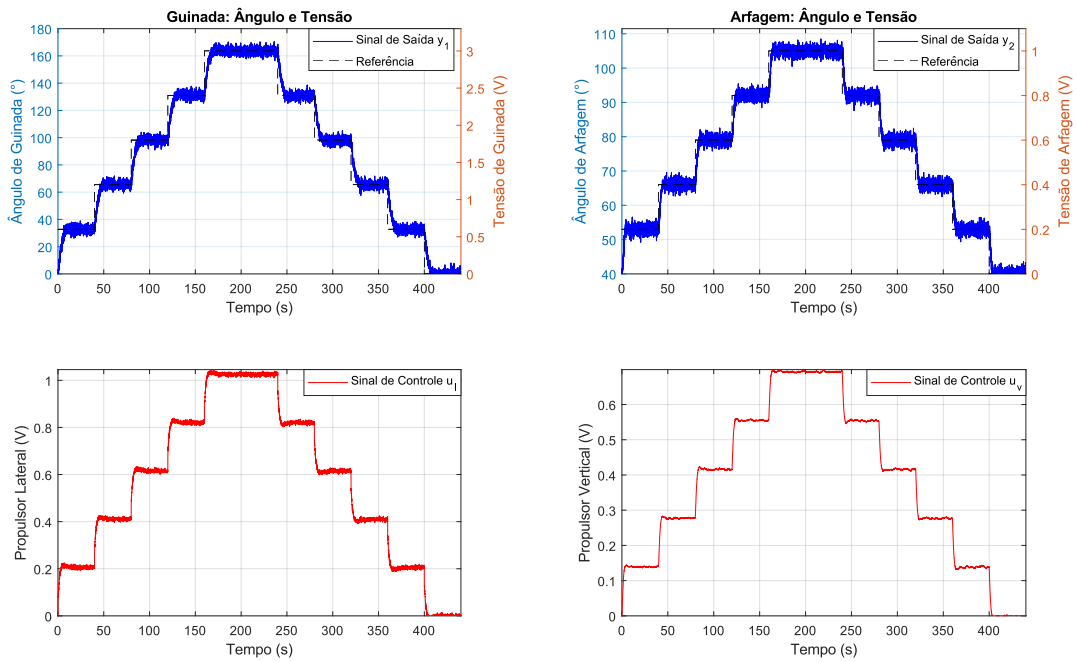


Figura 29 – Resultados Obtidos via Simulação com a Nova Sintonia para os Controladores PID no Modelo Estimado.

Fonte: Autor.

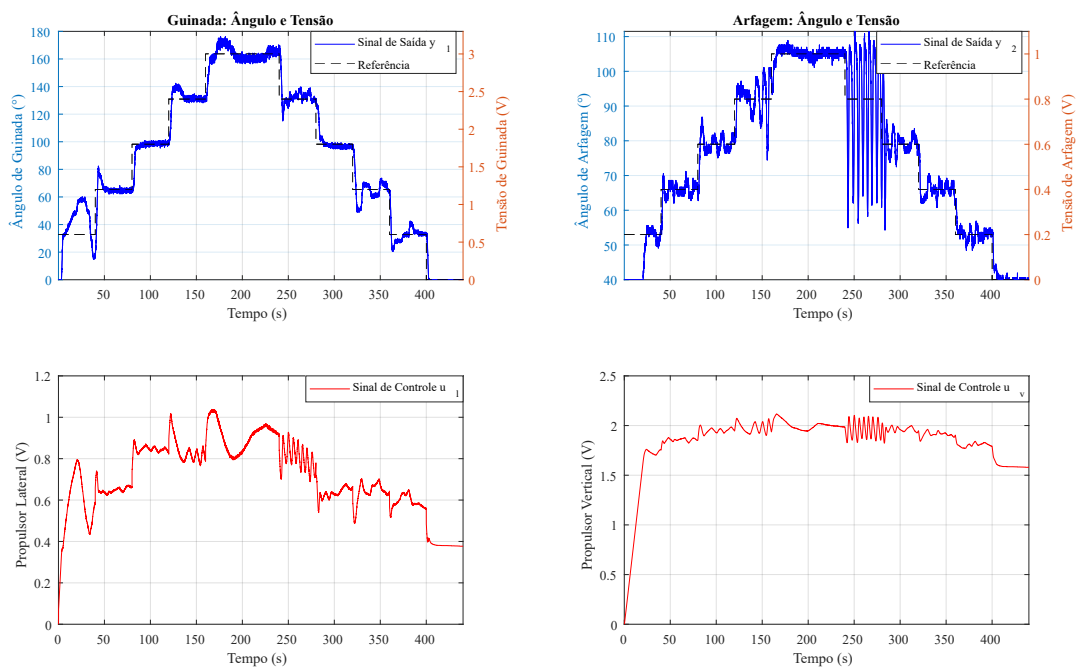


Figura 30 – Resultados Obtidos via Experimento Real com a Nova Sintonia para os Controladores PID no Protótipo H2-DOF.

Fonte: Autor.

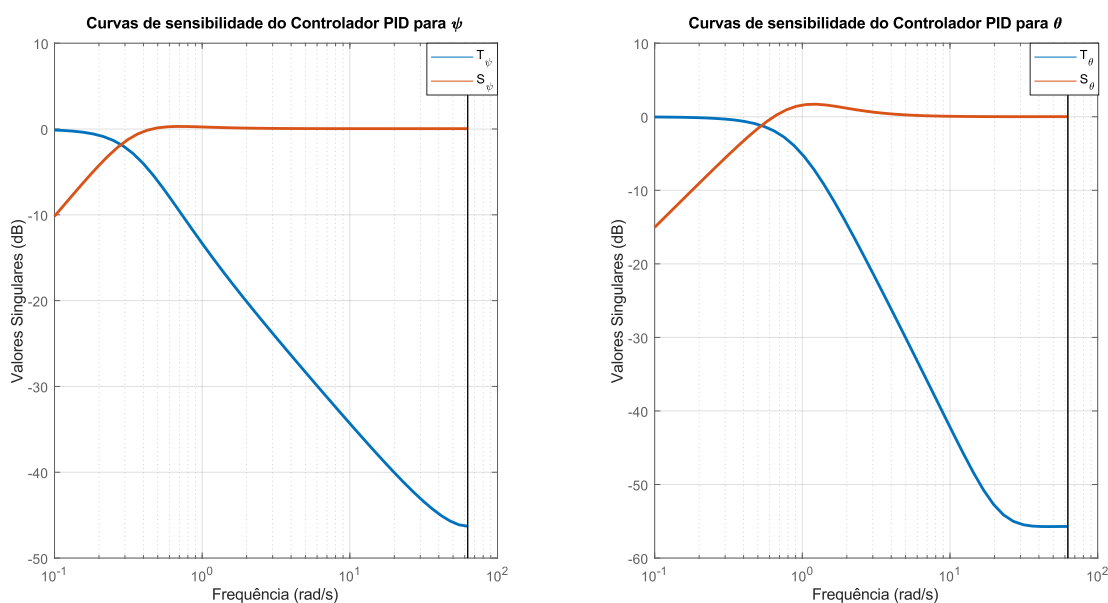


Figura 31 – Curvas de Sensibilidade do Nova Sintonia dos Controladores PID.

Fonte: Autor.

Tabela 6 – Índices dos Novos Controladores Obtidos nos Experimentos Simulado e Real.

Índices dos controladores PID	Experimento Simulado		Experimento Real	
	$PID_{\psi}$	$PID_{\theta}$	$PID_{\psi}$	$PID_{\theta}$
$K_p$	0,51283	0,00664	0,51283	0,00664
$K_i$	0,32620	0,56446	0,32620	0,56446
$K_d$	0,00027	0,00063	0,00027	0,00063
$ISE$	0,01703	0,00143	0,02928	0,01039
$ISU$	0,42063	0,19199	0,52964	3,47802
$\sigma_e^2$	0,01703	0,00143	0,02911	0,01032
$\sigma_u^2$	0,10759	0,04910	0,03014	0,07799
$GM(dB)$	6,02108	6,02075	6,02108	6,02075
$PM(^{\circ})$	57,77120	48,48647	57,77120	48,48647

#### 4.2.2 Análise de Projeto dos Controladores PI

Os controladores PI são concebidos pelas considerações de projeto descritas na subseção 3.2.3, onde é realizado dois projetos com dinâmicas desejadas de malha fechada diferentes, para os controladores PI, de acordo com os parâmetros na Tabela 7 e apresentadas pela respostas ao degrau unitário em malha aberta nas Figuras 32 e 33. Embora esses projetos apresentem dinâmicas desejadas próximas, os resultados simulados mostrados nas figuras 34 e 35 apontam que dinâmicas mais rápidas produzem maior entrada de ruídos de alta frequência no sinal de controle, o que reforça a necessidade das dinâmicas desejadas serem mais conservadoras.

Tabela 7 – Parâmetros Desejados para Sintonia dos Controladores PI do Projeto 1 e 2.

Malha do controlador PI	Projeto 1 - Dinâmica rápida		Projeto 2 - Dinâmica lenta	
	$\tau_{mf}(s)$	$\sigma$	$\tau_{mf}(s)$	$\sigma$
$G_{mf_1}(\psi)$	2,25000	1,23750	2,45000	1,40875
$G_{mf_2}(\theta)$	0,45000	2,34000	0,50000	2,75000

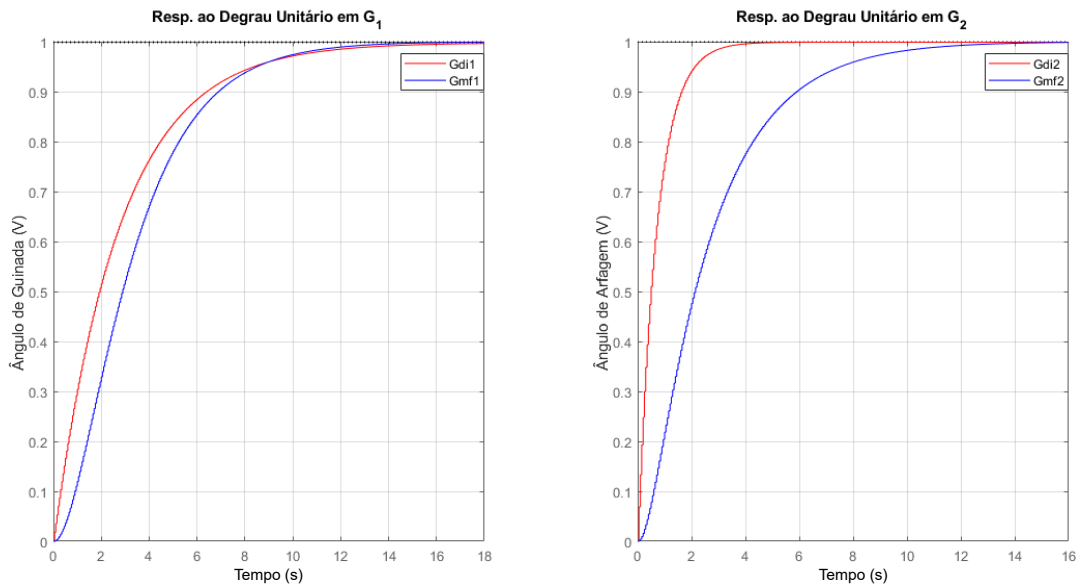


Figura 32 – Resposta ao Degrau da Planta e da Dinâmica Desejada (Projeto 1).

Fonte: Autor.

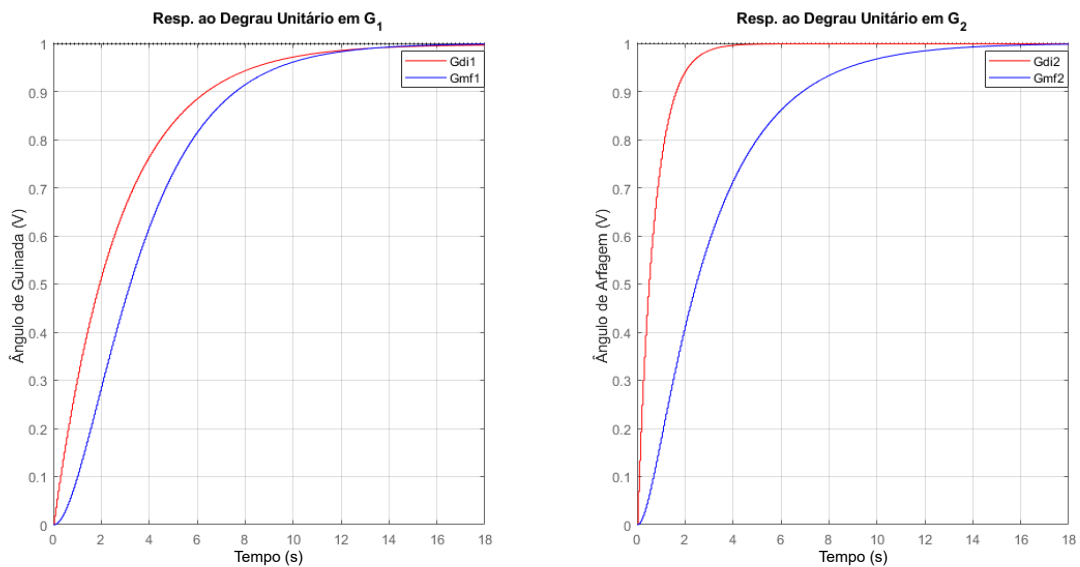


Figura 33 – Resposta ao Degrau da Planta e da Dinâmica Desejada (Projeto 2).

Fonte: Autor.

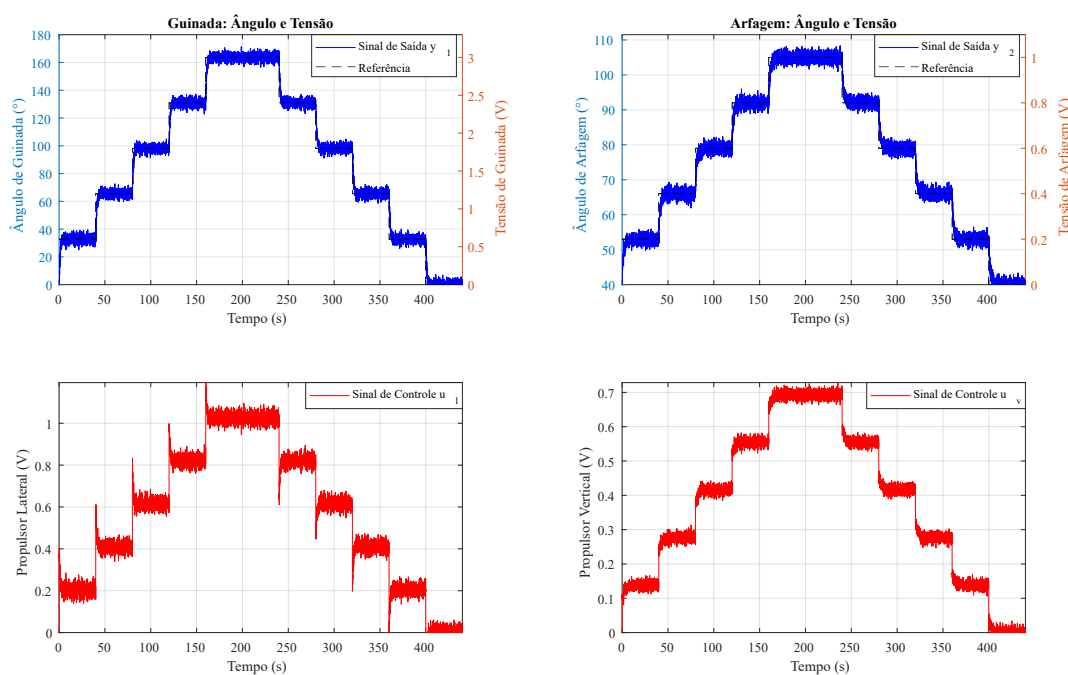


Figura 34 – Resultados Obtidos via Simulação com os Controladores PI do Projeto 1 no Modelo Estimado.

Fonte: Autor.

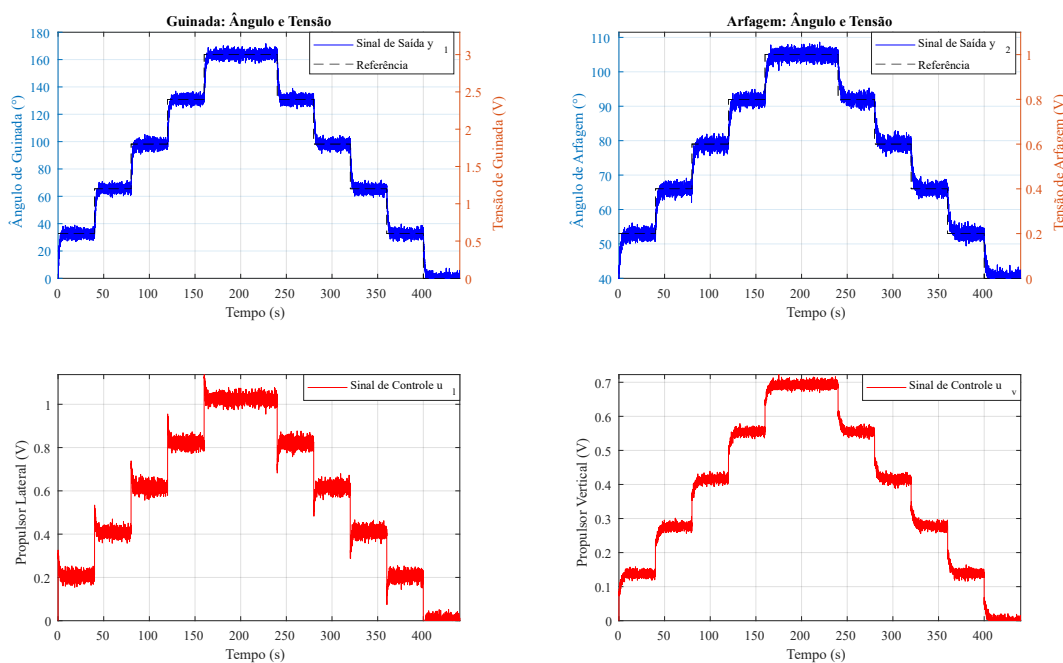


Figura 35 – Resultados Obtidos via Simulação com os Controladores PI do Projeto 2 no Modelo Estimado.

Fonte: Autor.

As curvas de sensibilidades dos projetos dos controladores PI, ilustradas nas Figuras 36 e 37, apresentam robustez razoável para o seguimento das referências, mitigação de perturbações de carga e de ruídos. Já em relação às curvas de sensibilidade dos projetos PID, das Figuras 27 e 31, o cruzamento das curvas dos projetos PI são menos acentuados e não possuem uma região de frequência sensível nas faixas de frequências entre 1 e 10 rad/s.

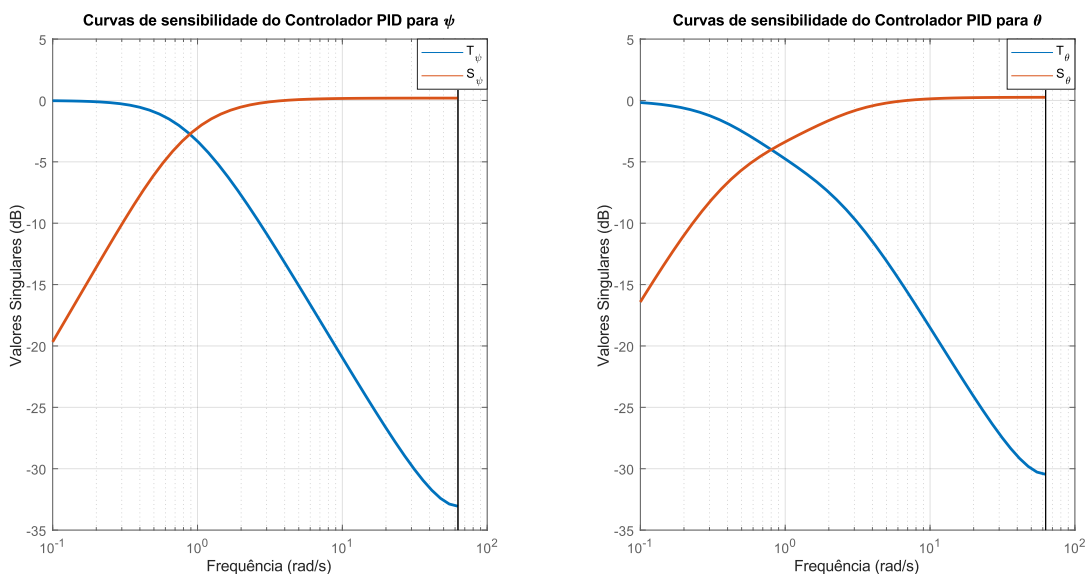


Figura 36 – Curvas de Sensibilidade dos Controladores PI do Projeto 1.

Fonte: Autor.

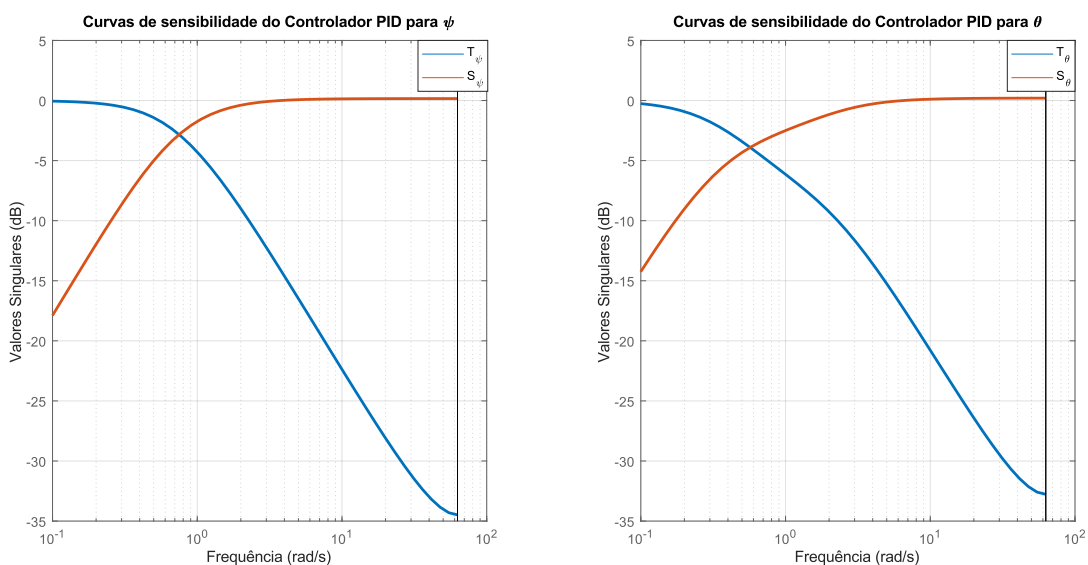


Figura 37 – Curvas de Sensibilidade dos Controladores PI do Projeto 2.

Fonte: Autor.

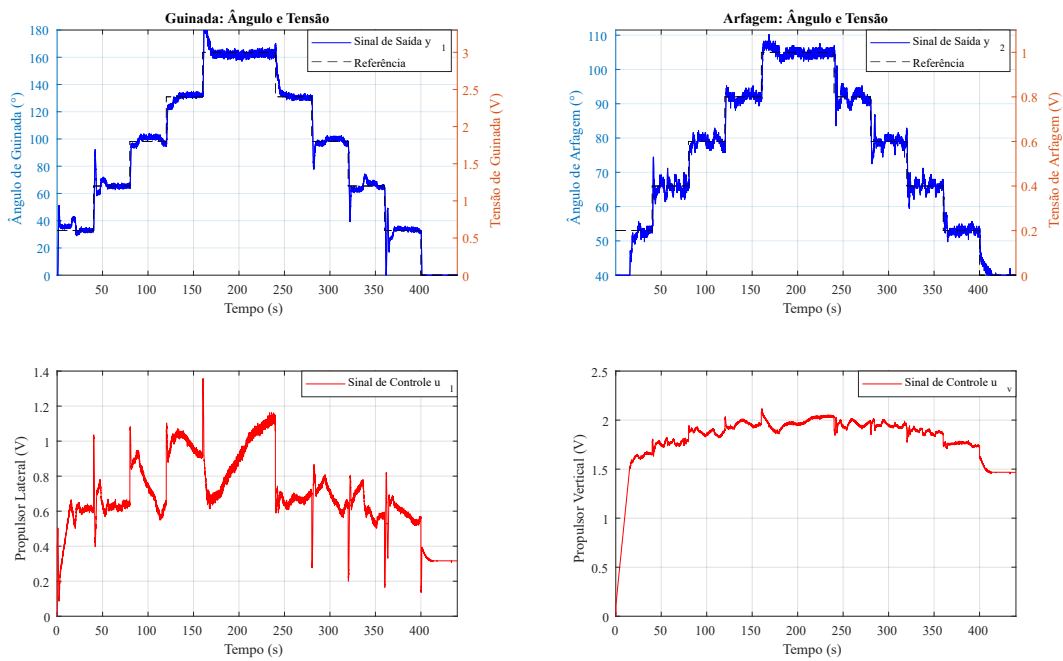


Figura 38 – Resultados Obtidos via Experimento Real com os Controladores PI do Projeto 1 no Protótipo H2-DOF.

Fonte: Autor.

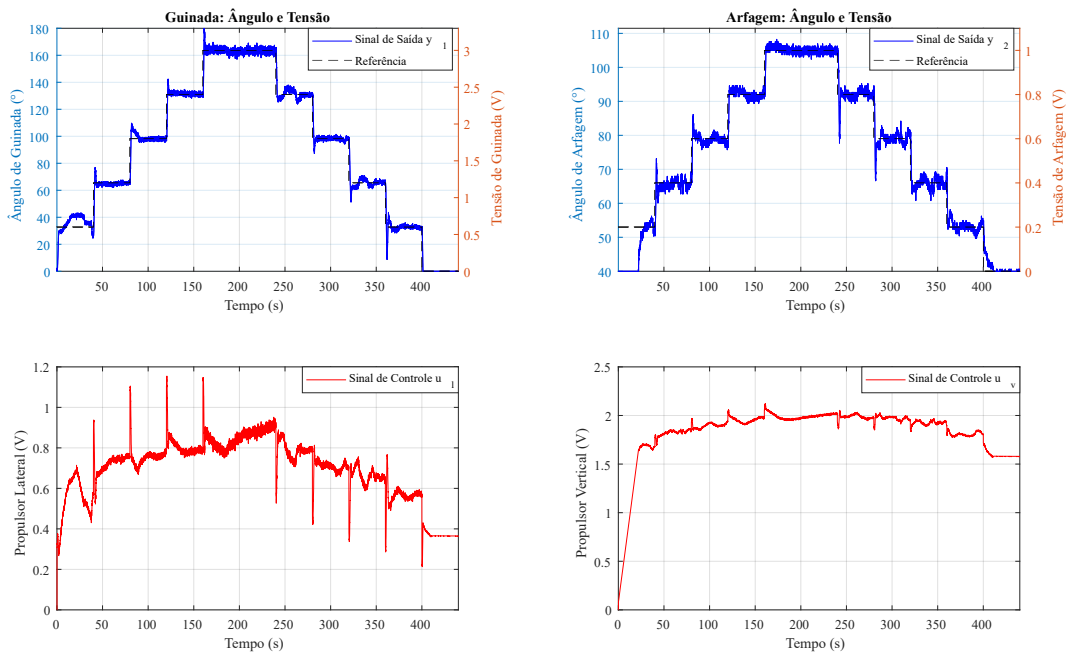


Figura 39 – Resultados Obtidos via Experimento Real com os Controladores PI do Projeto 2 no Protótipo H2-DOF.

Fonte: Autor.

Observa-se que os experimentos reais apresentados nas Figuras 38 e 39 reali-

zaram o seguimento das referências com menores oscilações em relação aos projetos dos controladores da estratégia PID. Além disso, o controlador PI do Projeto 1, apesar de ser mais oscilatório que o controlador PI do Projeto 2, desempenhou melhor para corrigir as trajetórias com referências menores, enquanto o controlador PI do Projeto 2, com dinâmica mais conservadora, conseguiu resultados melhores para as referências mais altas. Nas Tabelas 8 e 9 são apresentados os índices resultantes de cada projeto, onde é verificado que os índices tiveram uma melhora considerável em relação aos projetos dos controladores PID, principalmente, no início das etapas de descensão.

Tabela 8 – Índices dos Controladores PI Obtidos nos Experimentos Simulado e Real com os Parâmetros do Projeto 1.

Índices dos controladores PI	Experimento Simulado		Experimento Real	
	$PI_{\psi}$	$PI_{\theta}$	$PI_{\psi}$	$PI_{\theta}$
$K_p$	2,37436	0.80061	2,37436	0.80061
$K_i$	0,96811	0,64558	0,96811	0,64558
$ISE$	0,00616	0,00088	0,01218	0,00265
$ISU$	0,42262	0,19197	0,52290	3,33785
$\sigma_e^2$	0,00616	0,00088	0,01216	0,00260
$\sigma_u^2$	0,10958	0,04911	0,04494	0,06566
$GM(dB)$	6,02070	6,02148	6,02070	6,02148
$PM(^{\circ})$	58,56568	58,08192	58,56568	58,08192

Tabela 9 – Índices dos Controladores PI Obtidos nos Experimentos Simulado e Real com os Parâmetros do Projeto 2.

Índices dos controladores PI	Experimento Simulado		Experimento Real	
	$PI_{\psi}$	$PI_{\theta}$	$PI_{\psi}$	$PI_{\theta}$
$K_p$	2,02180	0,61748	2,02180	0,61748
$K_i$	0,78362	0,49790	0,78362	0,49790
$ISE$	0,00714	0,00108	0,00914	0,00340
$ISU$	0,42217	0,19184	0,50326	3,43781
$\sigma_e^2$	0,00714	0,00108	0,00911	0,00329
$\sigma_u^2$	0,10881	0,04893	0,02405	0,07578
$GM(dB)$	6,02086	6,02195	6,02086	6,02195
$PM(^{\circ})$	58,77861	58,51906	58,77861	58,51906

#### 4.2.3 Análise do Projeto de Síntese do Controlador Neural

O controlador neural parte da síntese de treinamento da rede com as aquisições de dados obtidas nos experimentos reais dos projetos (1 e 2) da estratégia PI das Figuras 38 e 39, como descrito na seção 3.4, onde as entradas da rede são consideradas como sendo os valores gerados pelas referências, saídas do processo e os

erros de cada malha a cada iteração em (111). Desta forma, é gerada uma combinação de dados dos projetos, ilustrada na Figura 40, para a rede treinar os padrões das variáveis de entrada da rede ( $X_j$ ) e aprender a correlacionar aos valores de ganhos dos controladores PI (nos termos da estrutura RST) na saída da rede, que produz o comportamento dado à região do projeto usado.

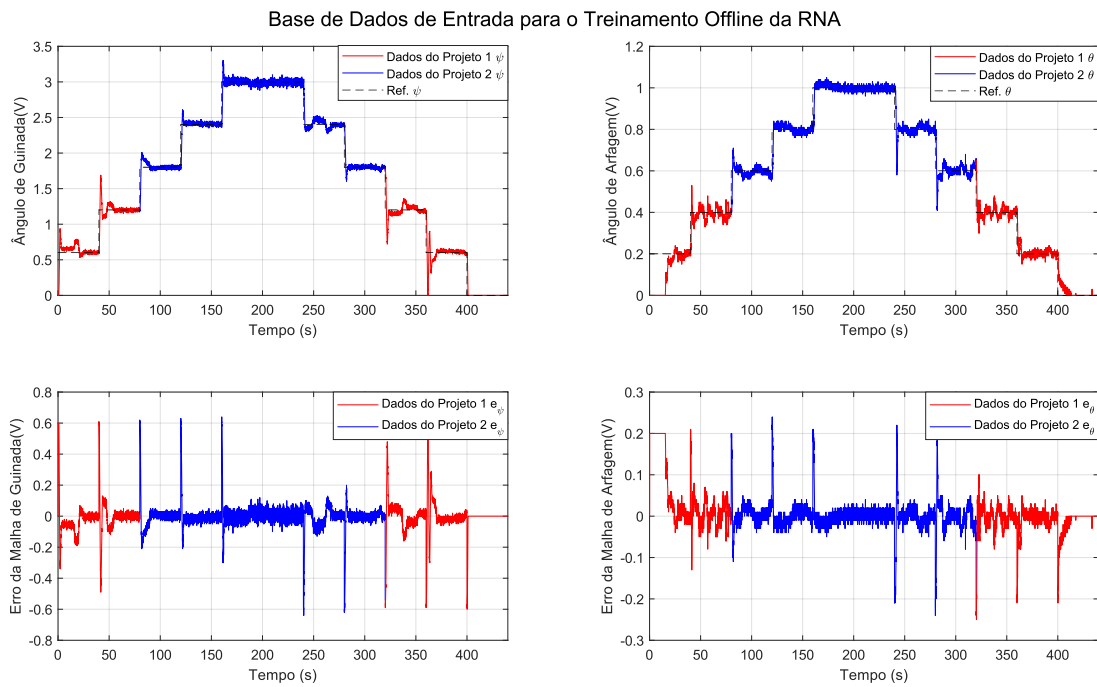


Figura 40 – Base de dados de Entrada ( $x_j$ ).

Fonte: Autor.

Posteriormente, realizado o treinamento de maneira *offline*, verifica-se o desempenho da rede pela quantidade de épocas de treinamento, o valor do gradiente do erro médio quadrático ( $\nabla E$ ) e o mínimo da função custo  $E_{av}$  em (96), dados apresentados na Tabela 10. Desta forma, é observado que, após 100 épocas de verificações na etapa de treinamento do algoritmo *Backpropagation*, não houve redução do erro médio quadrático na etapa de validação após a época 9, como apresentado na Figura 41, e os valores dos pesos e bias refinados são extraídos nas equações (148), (149) e (150).

Tabela 10 – Índices de Desempenho Obtidos no Treinamento da Rede com a Base de Dados dos Projetos dos Controladores PI.

Índices de Desempenho	Valor Inicial	Valor Final
Épocas	0	109
$E_{av}$	0,01670	$9,24 \times 10^{-6}$
$\nabla E$	0,00248	$4,18 \times 10^{-10}$

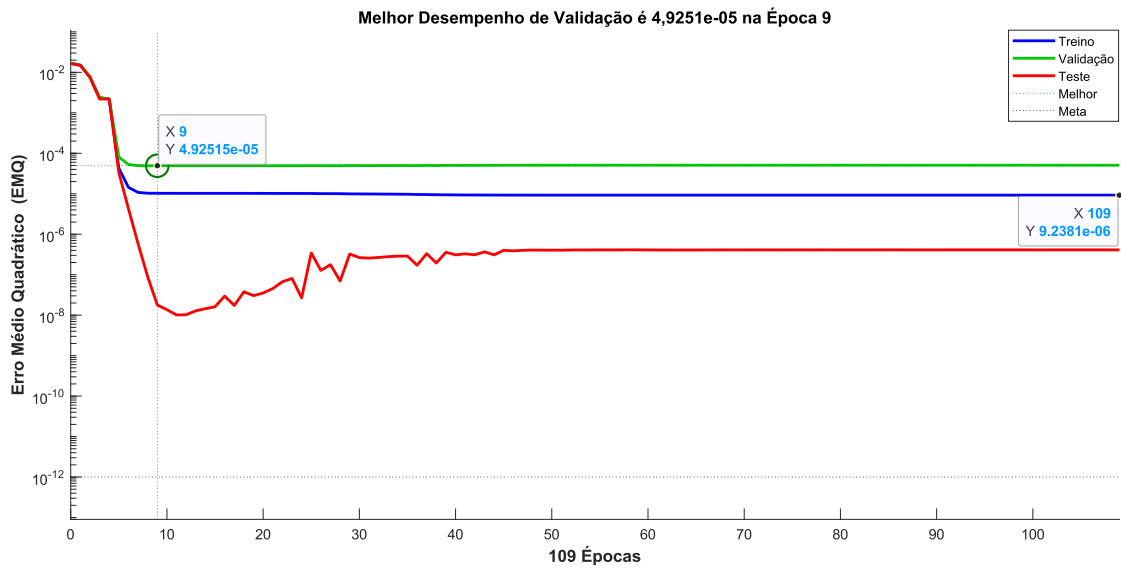


Figura 41 – Desempenho da Rede Neural nas Etapas de Treinamento, Validação e Teste.

Fonte: Autor.

$$W_{ij} = \begin{bmatrix} 0,62540 & -0,67977 & -0,64418 & -0,22440 & -0,67630 & -0,69925 \\ -0,98006 & 0,44015 & 0,37099 & 0,56315 & 0,19600 & -0,59086 \\ -1,14320 & -0,73326 & 0,30892 & -0,87294 & -0,32855 & -0,22283 \\ 0,07358 & -0,80742 & 0,76836 & 0,26860 & 0,13818 & -0,39550 \\ 0,20550 & 0,65287 & 0,01057 & -0,50372 & -0,98020 & 0,80324 \\ -3,38328 & -2,47535 & -0,93493 & -2,13193 & -0,73650 & -0,71201 \\ 0,96417 & 0,78896 & 0,09704 & -0,70235 & -0,58335 & 0,64652 \\ -1,36656 & 0,41817 & -0,87546 & 0,46585 & 1,88320 & 0,33396 \end{bmatrix} \quad (148)$$

$$\beta_{jk} = \begin{bmatrix} 0,17642 & 0,02290 & 0,83422 & -0,36352 & -1,28268 & 2,56579 \\ 0,57287 & 0,26118 & 1,35628 & -1,08095 & -0,82643 & 2,00499 \\ -0,70695 & -0,17647 & 1,59660 & 0,22525 & 1,28942 & 3,33001 \\ -0,04511 & 0,31790 & -0,59116 & 0,52580 & 0,18949 & 4,17783 \\ 0,89588 & -0,48370 & -1,29760 & 0,61741 & -0,98236 & -3,20345 \\ 0,34646 & 0,19972 & -1,66248 & 0,37693 & 0,16817 & -2,97422 \\ 0,57138 & -0,11854 & & & & \\ 0,84211 & 0,06224 & & & & \\ 0,13737 & 0,70220 & & & & \\ 0,30067 & 0,11586 & & & & \\ 0,50567 & -0,50009 & & & & \\ -1,03389 & -0,35172 & & & & \end{bmatrix} \quad (149)$$

$$\theta_j = \begin{bmatrix} -2,77359 \\ 1,94124 \\ 0,00863 \\ -0,01683 \\ -0,64133 \\ -0,63817 \\ 1,85531 \\ -1,16625 \end{bmatrix}, \quad \theta_k = \begin{bmatrix} -1,31745 \\ -0,98169 \\ 0,54908 \\ -0,37096 \\ -0,13744 \\ 1,00017 \end{bmatrix} \quad (150)$$

Desta maneira, com os pesos e os bias refinados, a rede neural treinada é acoplada na estrutura descentralizada de controladores PI, como na Figura 14, para se produzir os experimentos simulado e real que são apresentados nas Figuras 42 e 43. As previsões feitas pela rede neural sobre os termos da estrutura RST são apresentadas sobrepostas na Figura 44, onde são usadas para calcular a adaptação dos ganhos dos controladores PI, a cada iteração de  $k$ , na Figura 45, haja vista que a rede neural adotou uma predição aproximada para os dois casos. Por fim, a Tabela 11 mostra os resultados dos índices dos controladores PI neurais dos experimentos.

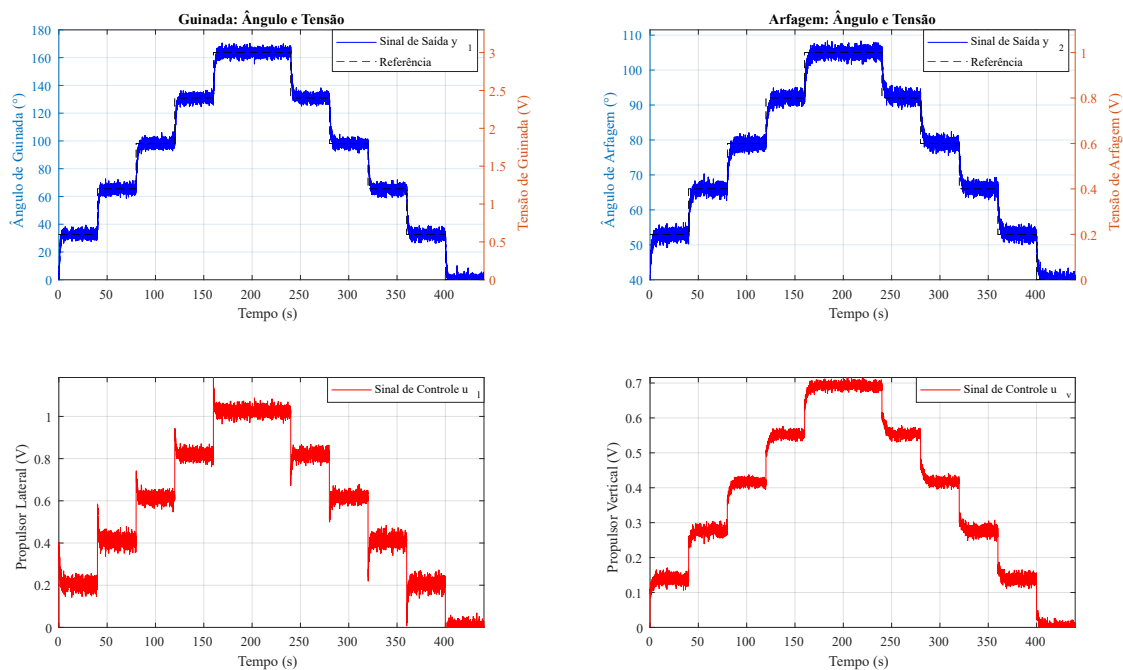


Figura 42 – Resultados Obtidos via Simulação com os Controladores PI Neurais no Modelo Estimado.

Fonte: Autor.

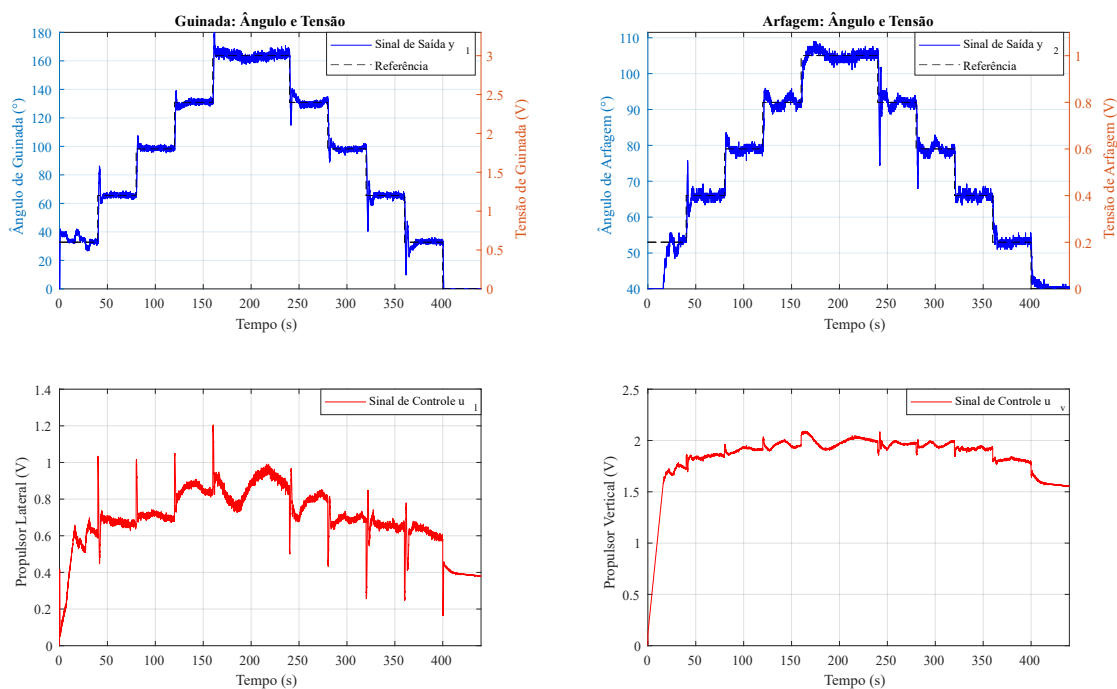


Figura 43 – Resultados Obtidos via Experimento Real com os Controladores PI Neurais no Protótipo H2-DOF.

Fonte: Autor.

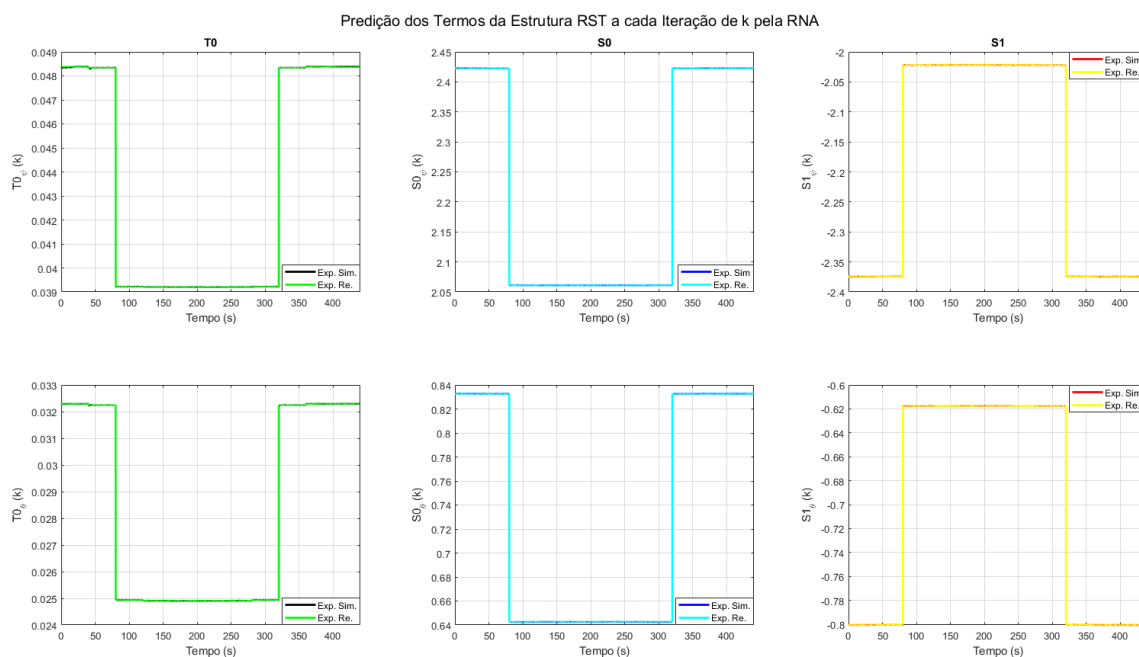


Figura 44 – Predição dos Termos RST pela RNA nos Experimentos Simulado e Real.

Fonte: Autor.

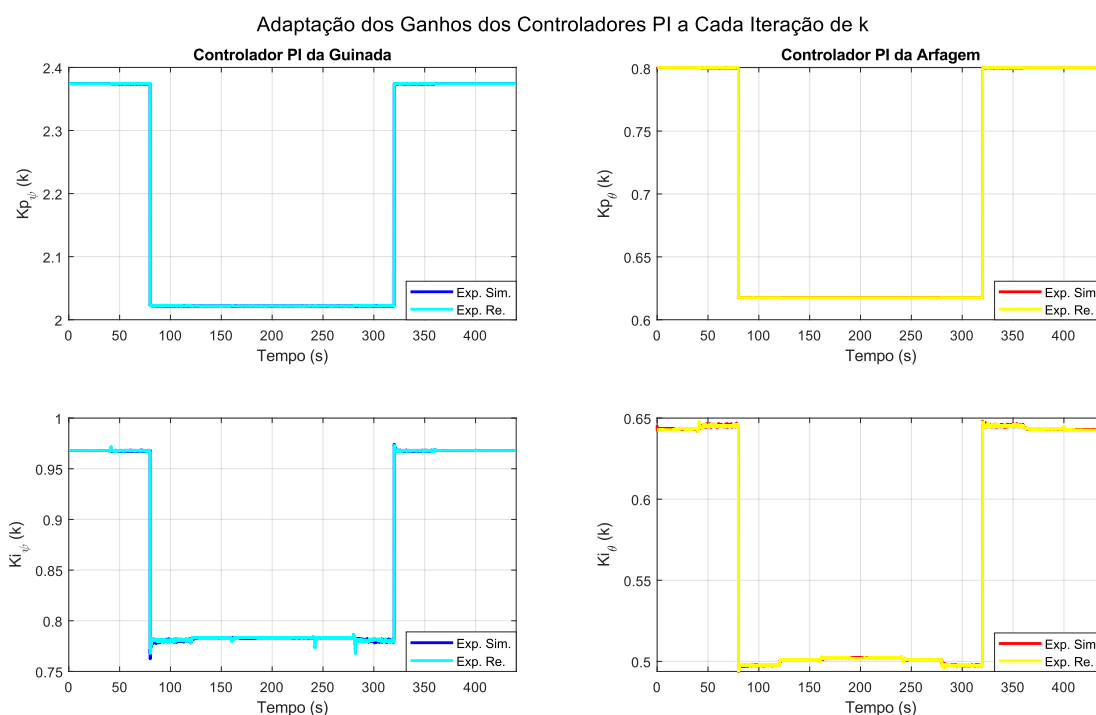


Figura 45 – Adaptação Ganhos dos Controladores PI nos Experimento Simulado e Real.

Fonte: Autor.

Tabela 11 – Índices dos Controladores Neurais PI Obtidos nos Experimentos Simulado e Real.

Índices dos controladores PI	Experimento Simulado		Experimento Real	
	$PI_{\psi}$	$PI_{\theta}$	$PI_{\psi}$	$PI_{\theta}$
$ISE$	0,00674	0,00098	0,00777	0,00272
$ISU$	0,42207	0,19171	0,50586	3,46150
$\sigma_e^2$	0,00674	0,00098	0,00775	0,00266
$\sigma_u^2$	0,10915	0,04889	0,02806	0,05947

#### 4.2.4 Análise Comparativa

Diante disso, é possível averiguar pelos gráficos sobrepostos dos projetos dos controladores PI a parâmetros fixos e do controlador PI Neural, apresentados na Figura 46, que a rede neural treinada conseguiu absorver os padrões de posição de trajetória, pelos parâmetros apresentados pela Figura 40, e correlacionar as regiões em que os ganhos dos projetos dos controladores fixos tiveram melhor desempenho. Com os vetores dos ângulos da Figura 46, foram produzidas as trajetórias tridimensionais de ascensão e descensão (de forma espelhada), geradas pelo uso dos controladores PI, sendo estas trajetórias mostradas na Figura 49.

Além disso, a rede neural torna parte das não linearidades do processo como

um padrão conhecido e aproximado, dando possíveis soluções para melhorar o desempenho do controlador por meio de microajustes, mesmo quando o processo estava aproximadamente acomodado nas referências, onde é mais evidente verificar este fator na parcela de ganhos integradores na Figura 45. As Tabelas 12 e 13 mostram os comparativos dos índices de desempenho e de robustez dos experimentos reais. No entanto, para o caso da adaptação dos ganhos gerados pela decisão da rede neural, foi feito um espectro de *GM* e *PM* produzidos pela rede neural, como uma perspectiva de avaliar a estabilidade do sistema em relação aos projetos com ganhos fixos, pelas Figuras 47 e 48.

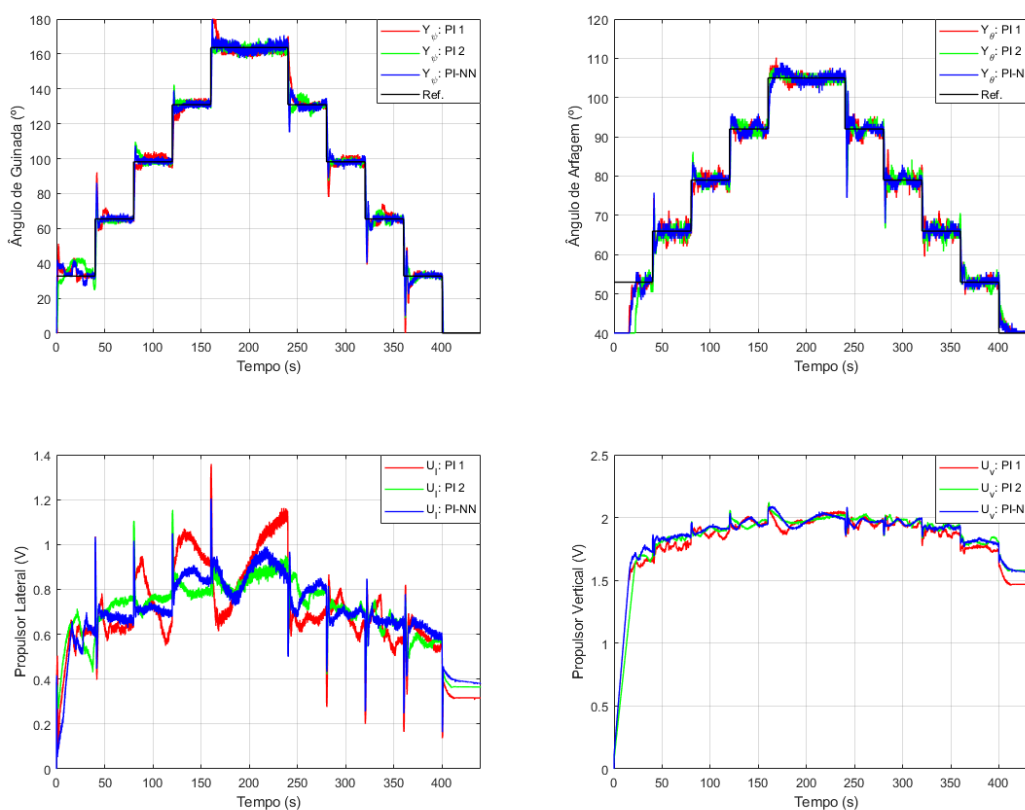


Figura 46 – Comparação Entre os Controladores PI dos Experimentos Reais.

Fonte: Autor.

Observa-se nas Tabelas 12 e 13, que as comparações dos resultados dos índices de desempenho e robustez dos controladores PI foram feitas pela análise por malha de processo, em que os valores marcados em **vermelho** representam os resultados menos favoráveis e os marcados em **verde** os mais favoráveis, para mostrar as vantagens e desvantagens de cada controlador na manutenção do seguimento de referências no processo.

Assim, os controladores responsáveis pelo ângulo de guinada ( $\psi$ ) foram comparados, onde foi possível verificar que o controlador PI a parâmetros fixos do Projeto 1

Tabela 12 – Comparação dos índices de desempenho dos controladores PI (Experimento Real).

Índice	PI - Projeto 1		PI - Projeto 2		PI-NN	
	$PI_{\psi}$	$PI_{\theta}$	$PI_{\psi}$	$PI_{\theta}$	$PI_{\psi}$	$PI_{\theta}$
ISE	0,01218	0,00265	0,00914	0,00340	0,00777	0,00272
ISU	0,52290	3,33785	0,50326	3,43781	0,50586	3,46150
$\sigma_e^2$	0,01216	0,00260	0,00911	0,00329	0,00775	0,00266
$\sigma_u^2$	0,04494	0,06566	0,02405	0,07578	0,02806	0,05947
$GM(dB)$	6,02070	6,02148	6,02086	6,02195	–	–
$PM(^{\circ})$	58,56568	58,08192	58,77861	58,51906	–	–

Tabela 13 – Comparação percentual dos índices de desempenho dos controles PI Fixos em relação ao PI-NN (referência).

Índice	PI - Projeto 1		PI - Projeto 2		PI-NN (Ref.)	
	$PI_{\psi}$	$PI_{\theta}$	$PI_{\psi}$	$PI_{\theta}$	$PI_{\psi}$	$PI_{\theta}$
ISE	+56,7%	-2,6%	+17,6%	+25,0%	0%	0%
ISU	+3,37%	-3,57%	-0,52%	-0,68%	0%	0%
$\sigma_e^2$	+56,8%	-2,26%	+17,5%	+23,7%	0%	0%
$\sigma_u^2$	+60,1%	+10,4%	-14,3%	+27,4%	0%	0%

apresentou uma relação de controle menos favorável para a manutenção dos ângulos de guinada e, devido à dinâmica do controle ser mais rápida que os demais, produziu um aumento da entrada de perturbações para o sinal de controle que se refletiu no aumento significativo da variância do sinal de controle ( $\sigma_u^2$ ) e na variância do sinal de erro ( $\sigma_e^2$ ). Entretanto, apesar dessas desvantagens, esse controlador apresentou uma relação de manutenção dos ângulos de arfagem ( $\theta$ ) interessante, principalmente, para as duas primeiras referências na fase de descensão, onde não houve uma queda abrupta das trajetórias em relação aos demais controladores da Figura 49.

Ademais, o controlador PI a parâmetros fixos do projeto 2 mostra uma redução favorável, em termos energéticos, do sinal de controle pelos valores dos termos de ISU e  $\sigma_u^2$ , que foram usados para a manutenção do seguimento das referências de  $\psi$  e com reduções razoáveis dos termos ISE e  $\sigma_e^2$ , quando comparadas com o controlador PI do projeto 1, apesar de ter apresentado uma piora na relação do seguimento de referência na malha de arfagem. Dessa maneira, devido à dinâmica conservadora do controlador produzir uma reação de correção mais lenta das referências, principalmente, em baixas altitudes e nas duas primeiras referências fase de descensão, com as quedas abruptas observadas nas trajetórias na Figura 49, o controlador PI do projeto 2 apresentou uma relação de seguimento de referência melhor para  $\psi$  do que para  $\theta$ , em que teve um aumento de ISU,  $\sigma_u^2$  e mesmo apresentando as melhores margens de robustez entre os controladores PI para ambas às malhas do processo.

Para o caso do controlador PI Neural, que foi produzido com base às experiências de comportamentos dinâmicos anteriores dos controladores PI a parâmetros fixos (do projeto 1 e 2), foi observado uma factibilidade promissora no uso das redes neurais aliadas à teoria de controle clássica na Tabela 13, como uma ferramenta de otimização, para se produzir uma generalização do comportamento dinâmico do processo, a fim de se promover uma hibridização entre os projetos de controle clássico e melhorar a relação de controle para ambas as malhas no processo H2-DOF. Dessa forma, o controlador PI Neural não apresentou apenas resultados interessantes pelos termos do ISE e  $\sigma_e^2$  para o malha de arfagem da Tabela 12, como também obteve resultados de desempenho muito próximos aos controladores que foram usados para o treinamento da rede neural.

No entanto, como essa metodologia adaptativa produz diferentes soluções para os ganhos para os controladores ao decorrer do tempo, não garantindo uma estabilidade plena como as margens de robustez dos controladores PI a parâmetros fixos, foi proposto analisar as margens de robustez pela aplicação do controlador PI Neural no modelo estimado com 1000 Ruídos Gaussianos de Variância  $\sigma_v^2$ , pelo método de Monte Carlo, para se verificar se as margens de robustez do controlador PI Neural sofreriam degradações severas, apresentado pelos índices na Figura 47.

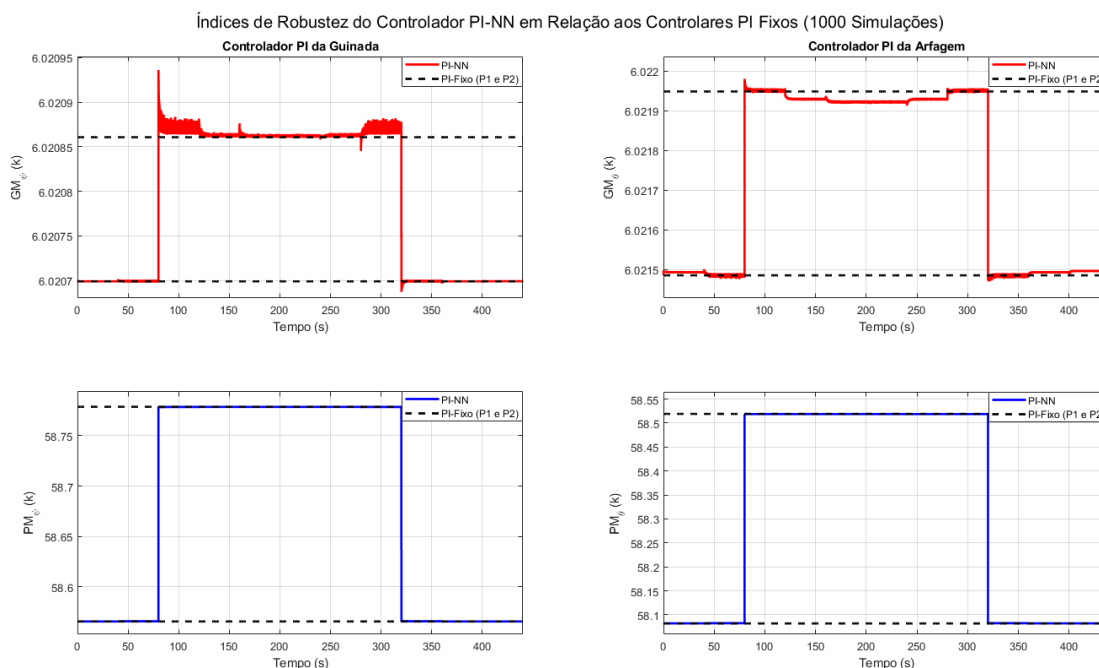


Figura 47 – Variação dos Índices de Robustez Produzidos pela Adaptação dos Ganhos no Controlador PI-NN nos Experimentos Simulados Usando o Método Monte Carlo com 1000 Ruídos Gaussianos de Variância  $\sigma_v^2$ .

Fonte: Autor.

É importante observar na Figura 47, que as soluções adaptativas produzidas,

em 1000 simulações com controlador PI Neural, geraram margens de robustez factíveis e próximas às faixas das margens de robustez dos controladores PI a parâmetros fixos do projeto 1 e 2 (linhas tracejadas), usados para treinar a rede neural, e entre certas faixas de tempo, como as de 75 s e 125 s, o controlador PI neural aplicou possíveis soluções de ganhos do controlador que produziram valores de  $GM$  modestamente maiores. Além disso, na Figura 48, foi analisado que as margens de robustez produzidas pela adaptação dos ganhos do controlador PI Neural no experimento real com o processo H2-DOF, e que complementa a ideia das soluções se manterem nas faixas de possibilidades das margens de robustez estabelecidas nas 1000 simulações pelo método de Monte Carlo, com uma singela diferença próxima de 275 s no controlador PI Neural responsável pela malha de  $\psi$ .

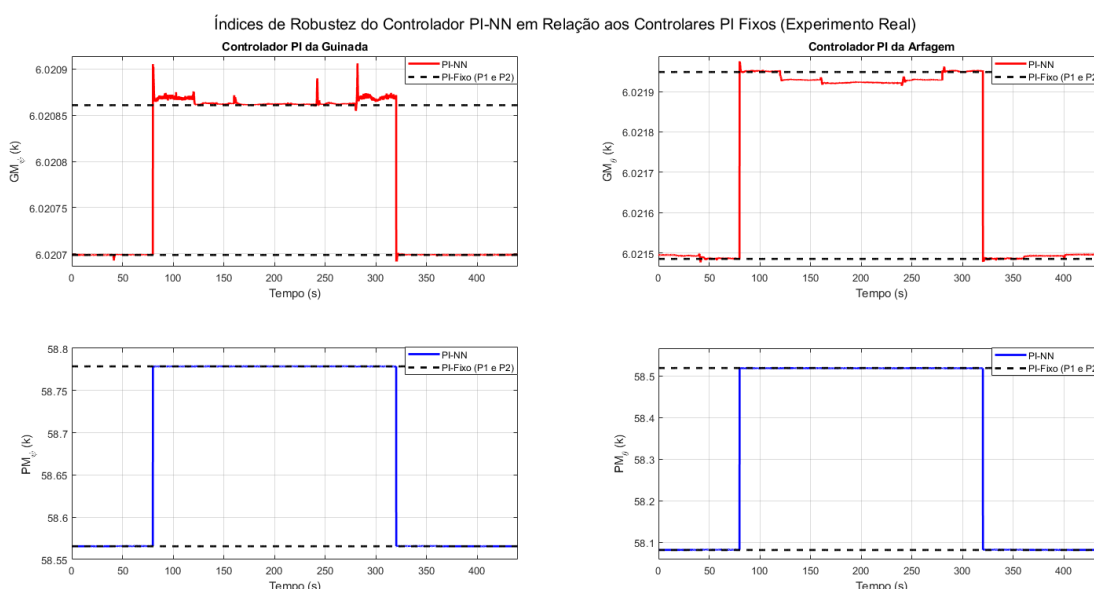


Figura 48 – Espectro dos Índices de Robustez Produzidos pela Adaptação dos Ganhos no Controlador PI-NN no Experimento Real.

Fonte: Autor.

Desta maneira, todos os controladores PI conseguiram fazer o seguimento das referências de forma satisfatória, sem instabilizar de maneira abrupta durante toda a trajetória nos experimentos reais com o processo H2-DOF, mesmo apresentando degradações de desempenho devido às incertezas já descritas nesta dissertação, conforme apresentado pelas trajetórias na Figura 49.

Portanto, a proposição feita sobre o estudo de caso da metodologia aplicada, para a síntese do controlador PI Neural nessa pesquisa, demonstrou um compromisso de robustez herdado pelo treinamento da rede neural com base em dados de comportamentos dinâmicos das variáveis escolhidas, obtidos em projetos de controladores PI a parâmetros fixos que já são robustos. Logo, o foco dessa produção não está em

demonstrar um controlador “vencedor”, pelos resultados positivos e promissores, mas sim demonstrar uma contribuição positiva dessa proposição em um experimento real, que possibilite a abertura de propostas para trabalhos futuros, na área de controle de processos, a expandir o uso potencial das redes neurais em técnicas de controle mais sofisticadas, onde envolvam a supervisão de comportamentos dinâmicos em processos mais complexos que talvez necessitem gerar soluções mais adequadas e não lineares na saída da rede.

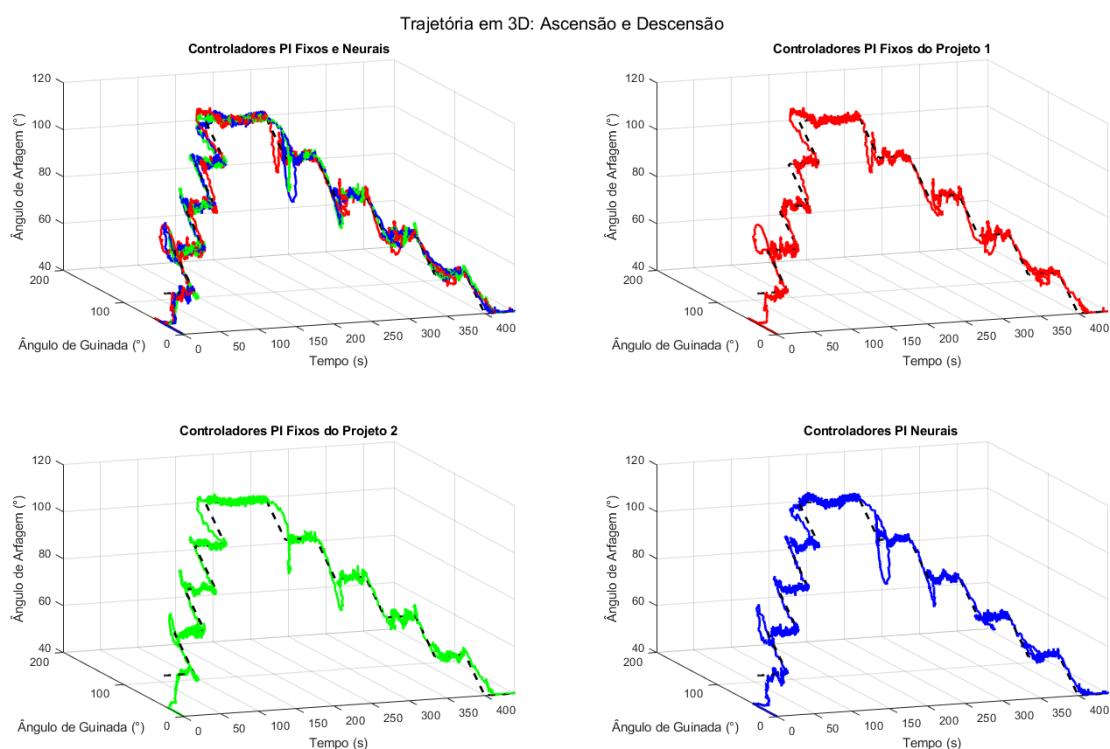


Figura 49 – Trajetória Tridimensional dos Ângulos Produzidos pelos Controladores PI.

Fonte: Autor.

## 5 CONCLUSÃO

Nesta dissertação, foi desenvolvida e validada uma metodologia de controle híbrido que integra controladores clássicos do tipo PI, sintonizados por alocação de polos, com redes neurais artificiais do tipo Perceptron Multicamadas (MLP), aplicados ao controle de posição de um sistema MIMO não linear com dois graus de liberdade (H2-DOF). O sistema, representado por um helicóptero didático, apresenta forte acoplamento entre as malhas de guinada e arfagem e uma dinâmica não linear, o que impõe desafios significativos à modelagem e ao projeto de controladores clássicos com ganhos fixos.

O primeiro passo da metodologia consistiu na identificação do sistema real por meio do algoritmo de Mínimos Quadrados Recursivos (MQR), permitindo a obtenção do modelo linearizado em espaço de estados. A partir desse modelo, foram projetados dois conjuntos de controladores PI e PID (Projetos 1 e 2), com sintonias distintas, para avaliação comparativa do desempenho em experimentos reais. Em seguida, os dados de entrada, saída e erro coletados desses experimentos foram utilizados para treinar uma rede neural MLP. O objetivo da rede foi identificar a região de operação da planta e, com base nisso, ajustar dinamicamente os ganhos para o caso do controlador PI, criando assim o controlador PI-Neural (PI-NN) proposto.

A análise quantitativa dos resultados, evidenciada nas Tabelas 12 e 13, comprova resultados positivos do controlador PI-Neural em relação aos projetos PI locais. A Tabela 12 apresenta os índices de desempenho obtidos em experimentos reais, revelando que o PI-Neural atingiu valores menores ou próximos, em termos de ISE (Integral do Erro Quadrático) e ISU (Integral do Sinal de Controle), em ambas as malhas. Esses resultados indicam, respectivamente, uma maior precisão no seguimento de referência e um menor esforço de controle. Notadamente, a suavidade do sinal de controle gerado pelo PI-Neural também se destaca, demonstrando em grande parte menores variâncias em relação aos controladores de base fixa, o que é um indicativo de menor desgaste físico dos atuadores.

A Tabela 13 complementa essa análise e resume a comparação percentual dos índices de desempenho entre os projetos PI locais e o PI-Neural. Desta maneira, o controlador PI-NN apresentou um desempenho satisfatório em relação a ambos os controladores locais em praticamente todos os critérios. O Projeto 1 expressou um desempenho pior para malha de guinada ( $PI_{\psi}$ ) de **+56,7% no ISE** e **+60,1% na variância do controle** ( $\sigma_u^2$ ), indicando maior erro e maior esforço de atuação em relação ao PI-NN, enquanto para a malha de arfagem ( $PI_{\theta}$ ) teve desempenho levemente superior no ISE, ISU e  $\sigma_e^2$ , mas ainda assim com aumento expressivo na variância do controle. Já para o Projeto 2, os ganhos em termos de ISU são praticamente irrisórios, enquanto os demais termos apontaram resultados razoáveis, se comparados com o

controlador do Projeto 1, onde destaca-se que a variância do sinal controle na malha de  $\psi$  apresentou resultados positivos, em relação aos demais controladores, devido à dinâmica do projeto ser mais conservadora.

Esses resultados refletem a capacidade da rede neural de capturar as não linearidades da planta e de se adaptar dinamicamente às mudanças de regime operacional, atuando sobre os ganhos do controlador PI em tempo real. Ao contrário dos controladores com ganhos fixos, que são otimizados para um ponto específico de operação, o PI-Neural conseguiu generalizar sua ação para diferentes condições, mesmo diante de perturbações, variações de carga e ruído no sistema real.

Além do desempenho satisfatório, destaca-se o fato de a rede neural ter sido treinada de maneira offline, sem a necessidade de aprendizado em tempo de execução, o que contribuiu para maior confiabilidade e menor complexidade computacional na fase de operação. Destarte, essa abordagem contribui para o potencial do uso de redes neurais como ferramentas de adaptação em tempo real para sistemas de controle baseados em estruturas clássicas, preservando a simplicidade e a interpretabilidade dos controladores PI, enquanto adiciona uma solução flexível.

Diante disso, a metodologia proposta, além de viável, pode ser promissora para aplicações em sistemas não lineares, multivariáveis e com fortes acoplamentos dinâmicos, caso os conceitos sejam mais lapidados em futuros trabalhos. Logo, o uso de uma rede MLP para adaptação dos ganhos do controlador revelou-se eficaz tanto em termos de desempenho quanto de esforço computacional, sendo compatível com sistemas embarcados de baixo custo.

Para trabalhos futuros, sugere-se a exploração de redes neurais mais sofisticadas, como redes recorrentes (RNN), redes com memória de longo prazo (LSTM) ou arquiteturas híbridas com aprendizado por reforço (*Deep Reinforcement Learning*), que podem ampliar a capacidade de adaptação com estimação de modelos não lineares (NARX) e a resposta da estrutura de controle (Madebo et al., 2024). Além disso, é recomendável estender a aplicação da metodologia para sistemas MIMO de maior ordem, e investigar a integração com estratégias de controle ótimo (LQR e LQG) e preditivos (GPC), de forma a antever comportamentos futuros da planta e atuar preventivamente na adaptação dos ganhos.

Portanto, a hibridização entre controladores clássicos e redes neurais representa uma alternativa interessante, flexível e eficaz para o controle de sistemas não lineares e dinâmicos, sendo uma linha de pesquisa relevante e com potencial de impacto prático na indústria moderna, que segundo Samad et al. (2020) se mostra promissora, e em um futuro próximo, o uso de controle inteligente terá uma expectativa de representatividade maior nas malhas de processos industriais, passando de 24% para 59%.

## REFERÊNCIAS

- AGUIRRE, Luis Antonio. **Introdução à Identificação de Sistemas: Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais**. 3a. MG, Brasil: Editora da UFMG, 2007.
- ARAS, Ayse Cisel; KAYNAK, Okyay. Trajectory tracking of a 2-DOF helicopter system using neuro-fuzzy system with parameterized conjunctors. In: IEEE. 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics. [S.l.: s.n.], 2014. p. 322–326.
- ARAUJO, Maryson.; DUTRA, Bruno G; PINHEIRO, Tarcisio; CUNHA, Leiliane; SILVEIRA, Antonio. Projeto de controladores digitais PID e I-PD via RST: Uma análise de desempenho. In: SIMPÓSIO Brasileiro de Automação inteligente (SBAI). [S.l.: s.n.], 2017.
- ARAÚJO, Rejane B. et al. Controladores preditivos filtrados utilizando otimização multiobjetivo para garantir offset-free e robustez, 2017.
- ASTRÖM, Karl J.; WITTENMARK, Björn. **Adaptive Control**. 2nd. Mineola, NY, USA: Dover Publications, 2008.
- ÅSTRÖM, Karl J.; WITTENMARK, Björn. **Computer-controlled systems: theory and design**. [S.l.]: Courier Corporation, 2013.
- BISHOP, Christopher M; NASRABADI, Nasser M. **Pattern recognition and machine learning**. [S.l.]: Springer, 2006. v. 4.
- BREGANON, Ricardo. **Controle de arfagem e guinada de um sistema de hélices paralelas**. 2009. Dissertação de Mestrado – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos. Acesso em: 2025-05-19.
- BRISTOL, Edgar. On a new measure of interaction for multivariable process control. **IEEE transactions on automatic control**, IEEE, v. 11, n. 1, p. 133–134, 1966.
- COELHO; COELHO. **Identificação de Sistemas Dinâmicos Lineares**. [S.l.]: Editora da UFSC, 2004.
- COELHO; JERONYMO; ARAUJO. Sistemas Dinâmicos Controle Classico e Preditivo Discreto. **Editora da UFSC**, 2019.
- COSTA, Alex Muniz da. **Aplicações de IA na sintonia de controladores PI e PID**. 2024. Diss. (Mestrado) – Universidade Federal do Rio Grande do Norte.
- CRUZ, Jahyrahã Leal dos Santos et al. Augmentação estocástica com horizonte de predição estendido baseada no PID para um sistema multivariável. Universidade Federal do Pará, 2019.

- CYBENKO, George. Approximation by superpositions of a sigmoidal function. **Mathematics of control, signals and systems**, Springer, v. 2, n. 4, p. 303–314, 1989.
- DUTRA, Bruno; SILVEIRA, Antonio; PEREIRA, Antonio. Grasping force estimation using state-space model and Kalman Filter. **Biomedical Signal Processing and Control**, Elsevier, v. 70, p. 103036, 2021.
- FRANKLIN, Gene F.; POWELL, J. David; EMAMI-NAEINI, Abbas. **Feedback Control of Dynamic Systems**. 8. ed. Boston: Pearson, 2019. ISBN 9780134685717.
- FRANKLIN, Gene F.; POWELL, J. David; WORKMAN, Michael L. **Digital Control of Dynamic Systems**. 3rd. [S.l.]: Addison-Wesley, 1997.
- GONZÁLEZ, Edilberto Carlos Vivas; RIVERA, Diego Mauricio; GÓMEZ, Edwar Jacinto. Model and observer-based controller design for a quanser helicopter with two dof. In: IEEE. 2012 IEEE ninth electronics, robotics and automotive mechanics conference. [S.l.: s.n.], 2012. p. 267–271.
- GÜNTHER, Johannes; REICHENSDÖRFER, Elias; PILARSKI, Patrick M; DIEPOLD, Klaus. Interpretable PID parameter tuning for control engineering using general dynamic neural networks: An extensive comparison. **Plos One**, Public Library of Science San Francisco, CA USA, v. 15, n. 12, e0243320, 2020.
- GUO, Hongpeng; GUO, Cheng; XU, Beichun; XIA, Yujie; SUN, Fanhui. MLP neural network-based regional logistics demand prediction. **Neural Comput. Appl.**, Springer-Verlag, Berlin, Heidelberg, v. 33, n. 9, p. 3939–3952, maio 2021. ISSN 0941-0643.
- HAYKIN, Simon. **Neural networks and learning machines, 3/E**. [S.l.]: Pearson Education India, 2009.
- HERNÁNDEZ, Rodrigo; GARCÍA, Luis Govinda; SALGADO, Tomás; GÓMEZ, Alfonso; FONSECA, Fernando. Neural network-based self-tuning PID control for underwater vehicles. **MDPI**, v. 16, n. 9, p. 1429, 2016.
- HORNIK, Kurt. Approximation capabilities of multilayer feedforward networks. **Neural Networks**, Elsevier, v. 4, n. 2, p. 251–257, 1991.
- KHALID, Muhammad Umair; SALEEM, Faisal; SHAIKH, Inam UI Hasan; ALI, Ahsan. Decentralized 2 degree of freedom loop shaping H controller for twin rotor aerodynamic system. In: IEEE. 2017 13th International Conference on Emerging Technologies (ICET). [S.l.: s.n.], 2017. p. 1–6.
- KINOSHITA, Kento; WAKITANI, Shin; OHNO, Shuichi. Design of neural network PID controller based on E-FRIT. **Electrical Engineering in Japan**, Wiley Online Library, v. 205, n. 2, p. 33–42, 2018.

LANDAU, Ioan Doré; LOZANO, Rogelio; M'SAAD, Mohammed; KARIMI, Alireza. **Adaptive control: algorithms, analysis and applications**. [S.l.]: Springer Science & Business Media, 2011.

LANDAU, Ioan Doré; ZITO, Gianluca. **Digital control systems: design, identification and implementation**. [S.l.]: Springer, 2006. v. 130.

MADEBO, Muluken Menebo; ABDISSA, Chala Merga; LEMMA, Lebsework Negash; NEGASH, Dereje Shiferaw. Robust tracking control for quadrotor UAV with external disturbances and uncertainties using neural network based MRAC. **IEEE Access**, IEEE, 2024.

MCCULLOCH, Warren S; PITTS, Walter. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, Springer, v. 5, n. 4, p. 115–133, 1943.

MINSKY, Marvin; PAPER, Seymour. **Perceptrons: An Introduction to Computational Geometry**. Cambridge, MA: MIT Press, 1969. ISBN 0-262-63022-2.

NASCIMENTO, Guilherme do. Aplicação de Controle Preditivo nas Malhas de Arfagem e Guinada da Planta Didática "Helicóptero 2-DOF. JOINVILLE, SC, 2016.

NOSRATI, Hasan G.; KHALILI, Khalil; GERDOOEI, Mahdi. Theoretical and experimental evaluation of no-slip feeding condition in ECAP-Conform of a square-section metallic rod. **The International Journal of Advanced Manufacturing Technology**, Springer, v. 112, p. 375–385, 2021.

NOSRATI, Hasan G.; KHALILI, Khalil; GERDOOEI, Mahdi. Theoretical and numerical investigation of required torque in ECAP-Conform process. **Metallurgical and Materials Transactions B**, Springer, v. 51, p. 519–528, 2020.

OGATA, Katsuhiko. **Discrete-time control systems**. [S.l.]: Prentice-Hall, Inc., 1995.

OGATA, Katsuhiko; SOUZA, Heloisa Coimbra de; TANNURI, Eduardo Aoun. **Engenharia de controle moderno**. [S.l.]: Pearson Prentice Hall, 2011.

OH, Paul Yu; GREEN, William Edward. CQAR: Closed quarter aerial robot design for reconnaissance, surveillance and target acquisition tasks in urban areas. **Drexel University. Department of Mechanical Engineering and Mechanics. Faculty research and publications.**, 2004.

OPPENHEIM, Alan V. **Discrete-time signal processing**. [S.l.]: Pearson Education India, 1999.

PÁDUA, Antônio B.; LEON, André C. F. P. de; LUDERMIR, Teresa B. **Redes neurais artificiais: teoria e aplicações**. [S.l.]: LTC Editora, 2007.

PATEL, Ravi; DEB, Dipankar; MODI, Himani; SHAH, Sunny. Adaptive backstepping control scheme with integral action for quanser 2-dof helicopter. In: IEEE. 2017

International Conference on Advances in Computing, Communications and Informatics (ICACCI). [S.l.: s.n.], 2017. p. 571–577.

POSTLETHWAITE, Ian. **Multivariable feedback control: analysis and design**. [S.l.]: John Wiley & Sons, Inc., 1996.

QUANSER, Quanser. 2-DOF Helicopter-Laboratory Manual. **Markham, ON, Canada: Quanser Inc**, 2011.

ROSENBLATT, Frank. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, American Psychological Association, v. 65, n. 6, p. 386, 1958.

RUMELHART, David E.; HINTON, Geoffrey E.; WILLIAMS, Ronald J. Learning representations by back-propagating errors. **nature**, Nature Publishing Group UK London, v. 323, n. 6088, p. 533–536, 1986.

SAMAD, Tariq et al. Industry engagement with control research: Perspective and messages. **Annual Reviews in Control**, Elsevier, v. 49, p. 1–14, 2020.

SARANGAPANI, Jagannathan. **Neural network control of nonlinear discrete-time systems**. [S.l.]: CRC press, 2018.

SEBORG, D. E.; EDGAR, T. F.; MELLICHAMP, D. A. **Process Dynamics and Control**. 2. ed. [S.l.]: John Wiley & Sons, 2003. ISBN 9780471000778.

SHARIFI, Iman; ALASTY, Aria. Self-Tuning PID Control via a Hybrid Actor-Critic-Based Neural Structure for Quadcopter Control. **arXiv preprint arXiv:2307.01312**, 2023.

SILVA, Daniel Abreu Macedo da; NASCIMENTO, André Cavalcante do; BARROS ARAÚJO, Rejane de; SOUZA MELO, Rogério José de. Generalized predictive controller applied in a bidirectional dc-dc converter. In: IEEE. 2021 Brazilian Power Electronics Conference (COBEP). [S.l.: s.n.], 2021. p. 1–6.

SILVA, Daniel Abreu Macedo da et al. Analysis of classical and advanced control techniques tuned with reinforcement learning. Universidade Federal do Pará, 2023.

SILVEIRA, Antonio. **Introdução ao controle digital, moderno e preditivo**. [S.l.]: 1a Ed. versão 1.0, LACOS-UFPA. Disponível em: <https://shorturl.at/xJX28>. Acesso em: 17/04/2024, 2024.

SILVEIRA, Antonio; SAGLIANO, Marco; TRENTINI, Rodrigo; SEELBINDER, David; THEIL, Stephan. Generalized Predictive Control: ARIX vs. ARIMAX-based Designs for a Floating Spacecraft Emulator Using a Quadcopter. **IEEE Transactions on Industry Applications**, IEEE, 2024.

SILVEIRA, Antonio; SILVA, Anderson; COELHO, Antonio; REAL, José; SILVA, Orlando. Design and real-time implementation of a wireless autopilot using multivariable

predictive generalized minimum variance control in the state-space. **Aerospace Science and Technology**, v. 105, p. 106053, 2020. ISSN 1270-9638.

SKOGESTAD, Sigurd; POSTLETHWAITE, Ian. **Multivariable Feedback Control: Analysis and Design**. 2. ed. [S.l.]: Wiley-Interscience, 2005.

STEVENS, Brian L.; LEWIS, Frank L.; JOHNSON, Eric N. **Aircraft Control and Simulation: dynamics, controls design, and autonomous systems**. 3rd. [S.l.]: John Wiley & Sons, Inc., 2016.

WERBOS, Paul J. Backpropagation through time: what it does and how to do it. **Proceedings of the IEEE**, IEEE, v. 78, n. 10, p. 1550–1560, 1990.

WINGERTER, Denise Guerra; SANTOS, Emelyne Gabrielly de Oliveira; BARBOSA, Isabelle Ribeiro. Uso de redes neurais artificiais para classificação de municípios quanto à vulnerabilidade social no Estado do Rio Grande do Norte, Brasil. **Cadernos de Saúde Pública**, Escola Nacional de Saúde Pública Sergio Arouca, Fundação Oswaldo Cruz, v. 36, n. 8, e00038319, 2020. ISSN 0102-311X.

WU, Bing; WU, Jiale; ZHANG, Jian; TANG, Guojian; ZHAO, Zhijia. Adaptive neural control of a 2DOF helicopter with input saturation and time-varying output constraint. In: MDPI, 11. **ACTUATORS**. [S.l.: s.n.], 2022. v. 11, p. 336.

ZHOU, J; LIU, Y; YU, J et al. Study on Neural Network Self-Tuning PID Control for Temperature of Active Solar House Heating System. In: 2010 2nd International Workshop on Intelligent Systems and Applications (ISA). [S.l.: s.n.], 2010.

ZHU, Yinfei; ZHAO, Han; SUN, Hao; HUANG, Kang; DONG, Yinghui. Trajectory tracking control of two degrees-of-freedom helicopter based on Udwadia–Kalaba theory. **Advances in Mechanical Engineering**, SAGE Publications Sage UK: London, England, v. 10, n. 11, p. 1687814018808939, 2018.

ZRIBI, Ali; CHTOUROU, Mohamed; DJEMEL, Mohamed. A new PID neural network controller design for nonlinear processes. **Journal of Circuits, Systems and Computers**, World Scientific, v. 27, n. 04, p. 1850065, 2018.