

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA
CAMPUS DE ILHA SOLTEIRA**

MARCIO MAESTA ANDREASSA

**CONSTRUÇÃO, MODELAGEM E CONTROLE DE UM PÊNDULO COM
RODA DE REAÇÃO UTILIZANDO TÉCNICAS MODERNAS**

ILHA SOLTEIRA

2022

MARCIO MAESTA ANDREASSA

**CONSTRUÇÃO, MODELAGEM E CONTROLE DE UM PÊNDULO COM
RODA DE REAÇÃO UTILIZANDO TÉCNICAS MODERNAS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, Faculdade de Ilha Solteira - UNESP como parte dos requisitos necessários para obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Jean Marcos de Souza Ribeiro

ILHA SOLTEIRA

2022

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

A557c Andreassa, Marcio Maesta.
Construção, modelagem e controle de um pêndulo com roda de reação
utilizando técnicas modernas / Marcio Maesta Andreassa. -- Ilha Solteira: [s.n.],
2022
71 f. : il.

Dissertação (mestrado) - Universidade Estadual Paulista. Faculdade de
Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2022

Orientador: Jean Marcos de Souza Ribeiro
Inclui bibliografia

1. Pêndulo invertido por roda de reação. 2. Hardwares de baixo custo. 3.
Controles lineares. 4. Software SCADA. 5. Swing up.


Raiane da Silva Santos


CERTIFICADO DE APROVAÇÃO

TÍTULO DA DISSERTAÇÃO: CONSTRUÇÃO, MODELAGEM E CONTROLE DE UM PÊNDULO COM RODA DE REAÇÃO UTILIZANDO TÉCNICAS MODERNAS

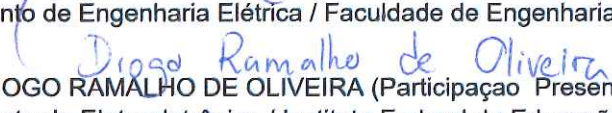
AUTOR: MARCIO MAESTA ANDREASSA

ORIENTADOR: JEAN MARCOS DE SOUZA RIBEIRO

Aprovado como parte das exigências para obtenção do Título de Mestre em ENGENHARIA ELÉTRICA, área: Automação pela Comissão Examinadora:


Prof. Dr. JEAN MARCOS DE SOUZA RIBEIRO (Participação Presencial)
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira - UNESP


Prof. Dr. MARCELO CARVALHO MINHOTO TEIXEIRA (Participação Presencial)
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira - UNESP


Prof. Dr. DIOGO RAMALHO DE OLIVEIRA (Participação Presencial)
Departamento de Eletroeletrônica / Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso do Sul - IFMS

Ilha Solteira, 25 de abril de 2022

Agradecimentos

Primeiramente gostaria de agradecer meus pais e irmã por todo o apoio e por me encorajarem a continuar estudando.

Para minha namorada que sempre apoiou minhas escolhas, e me ajudou nos momentos que eu mais precisava.

Para o Prof. Dr. Jean Marcos de Souza Ribeiro, por toda contribuição com seu conhecimento e orientação que possibilitou realizar essa pesquisa de mestrado.

Para o Dr. João Trentin que compartilhou o seu conhecimento, auxiliando desta forma este trabalho.

Para meus amigos da instituição UNESP que me apoiaram diante das dificuldades, e do mestrado.

Gostaria de agradecer a Deus por todas as oportunidades que surgiram em minha vida que possibilitou chegar onde estou.

E por fim, gostaria também de agradecer pelo apoio financeiro a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

*"Se você quiser descobrir os segredos do Universo, pense
em termos de energia, frequência e vibração."*

Nikola Tesla (1856 - 1943)

Resumo

Este trabalho tem o foco na modelagem e controle do pêndulo invertido por roda de reação. É uma variante relativamente nova do pêndulo invertido com dinâmica não linear, que utiliza como atuador uma roda de reação para troca de momento angular, que tem sido de interesse às áreas de robótica, pesquisa, acadêmica e aeroespacial. Esta dissertação propõe a modelagem e controle para elevar o pêndulo e estabilizá-lo em um ponto de equilíbrio instável. O pêndulo foi construído com estruturas mecânicas de baixo custo, e também é empregado uma simulação pelo *software* Matlab Simulink, para análise de desempenho a partir dos parâmetros calculados e obtidos experimentalmente. Para levar o pêndulo até a vizinhança do ponto de equilíbrio instável, foi aplicado um método inspirado no controle de energia, desenvolvendo um movimento denominado *swing up*. O controlador linear empregado para estabilizar o pêndulo sobre um ponto de um equilíbrio instável consiste de um controle por realimentação de estados. Para a troca de controladores em suas regiões de atuação, foi desenvolvida uma estratégia de chaveamento. A estratégia de chaveamento foi implementada por um microcontrolador, sendo o motor da roda de reação alimentado a partir de uma ponte H. Um *software* de supervisão e aquisição de dados foi implementado no modelo experimental, com qual é possível, em tempo real, ajustar os ganhos dos controladores e analisar as respostas em gráficos.

Palavras-chave: Pêndulo invertido por roda de reação. *Hardwares* de baixo custo. Controladores lineares. *Software* SCADA. *Swing up*

Abstract

This work focuses on the modeling and control of a reaction wheel pendulum. It is a relatively new variant of the inverted pendulum with nonlinear dynamics, that uses a reaction wheel as an actuator to exchange angular momentum, which has been of interest, for instance, in robotics, research and aerospace. This dissertation proposes the modeling and control to raise the pendulum and stabilize it an unstable equilibrium point. The pendulum was built with low-cost mechanical structures, and the numerical simulation was carried out using Simulink software, to verify the performance analysis based on the calculated parameters and those obtained experimentally. To bring the pendulum to the neighborhood of an unstable equilibrium point, a method inspired in energy control is applied, developing a movement called swing up. The linear controller used to stabilize the pendulum over an unstable equilibrium point consists of a state feedback control. For the switch of controllers in their operating regions, a switching strategy is developed. The switching strategy was implemented by a microcontroller, with the reaction wheel being powered with an H-bridge. A supervision and data acquisition software is implemented in the experimental model, in which it is possible, in real time, to adjust the gains of the controllers and analyze the responses in graphs.

Keywords: Reaction wheel pendulum. Low cost hardware. Linear controllers. Software SCADA. Swing up

Lista de Figuras

1	Pêndulo invertido com roda de reação	14
2	Pêndulo invertido 3-D (cubli)	17
3	Ilustração do modelo do PIRR	21
4	Foto lateral PIRR	25
5	Foto frontal PIRR e Roda no Solidworks	26
6	Microcontrolador, driver e encoder rotacional	27
7	Variação do ângulo da haste (σ) para estimar o atrito do pêndulo	30
8	Aproximação gráfica das amostas do ensaio	31
9	Gráficos do controle de estabilização	35
10	Gráfico de velocidade do controle de estabilização	36
11	Gráficos de resultados com variável <i>swing up</i> δ_s	39
12	Área de atuação dos controladores	40
13	Diagrama de blocos	41
14	Sistema elétrico do modelo PIRR	42
15	Resultado dos gráficos de estados	45
16	Resultado do gráfico dos controladores	46
17	Resultado dos gráficos de estados com filtro	47
18	Gráfico dos controladores com filtro	48
A.1.1	Modelo simulink PIRR	57
A.3.1	Modelo roda de reação solidworks	67
A.3.2	Modelo roda de reação solidworks vista frontal	68
A.4.1	Tela inicial Elipse E3	68
A.4.2	Tela de controle Elipse E3	69
A.4.3	Tela de gráficos Elipse E3	69
A.4.4	Gráfico posição Elipse E3	70

Lista de Tabelas

1	Parâmetros físicos do PIRR	28
2	Parâmetros físicos do motor cc de imã permanente	29
3	Índice de desempenho	49
4	Tabela de valores	50

Lista de Símbolos e Acrônimos

A_m	-	Atrito do motor
$C.I$	-	Condição inicial
C_i	-	Constantes de inércia
E_{ct}	-	Energia cinética
E_{pg}	-	Energia potencial
F_c	-	Atrito de coulomb do pêndulo
F_{rwp}	-	Atrito do pêndulo
F_v	-	Atrito viscoso do pêndulo
g	-	Aceleração da gravidade
h_v	-	Tempo de amostragem da velocidade
I_a	-	Corrente de armadura
I_{ab}	-	Corrente de armadura de rotor bloqueado
I_p	-	Momento de inércia do pêndulo
I_r	-	Momento de inércia da roda
K	-	Ganhos do controle de estabilização
K_t	-	Constante de torque
K_v	-	Constante de força eletromotriz
K_{sw}	-	Ganho do controle <i>swing up</i>
L	-	Comprimento da haste
L_{ag}	-	Função de Lagrange
L_{ch1}	-	Distância do pivô ao centro de massa da haste 1
L_{ch2}	-	Distância do pivô ao centro de massa da haste 2
L_{h1}	-	Comprimento da haste da roda em relação ao pivô
L_{h2}	-	Comprimento da haste do contrapeso em relação ao pivô
L_{h3}	-	Distância do centro de massa do conjunto de massas na roda em relação ao pivô
L_{h4}	-	Distância do centro de massa do contrapeso em relação ao pivô
L_y	-	Função de Lyapunov
\bar{M}_1	-	Centro de massas da roda
\bar{M}_2	-	Centro de massas do contrapeso
M_a	-	Massa do acoplamento

M_c	-	Massa do contrapeso
M_{h1}	-	Massa da haste 1
M_{h2}	-	Massa da haste 2
M_m	-	Massa do motor
M_r	-	Massa da roda
M_{tr}	-	Conjunto de massas na roda
n	-	Redução do motor
P_a	-	Pólos malha aberta
P_f	-	Pólos malha fechada
R_a	-	Resistência de armadura
T	-	Constante de tempo de filtragem
u_1	-	Controle de estabilização
u_2	-	Controle <i>swing up</i>
V	-	Tensão de alimentação do motor
y	-	Sinal de saída da velocidade
$Y(s)$	-	Sinal de saída da velocidade no domínio de Laplace
y_f	-	Sinal de saída filtrado da velocidade
$Y_f(s)$	-	Sinal de saída filtrado da velocidade no domínio de Laplace
δ	-	Posição de referência para o controle de estabilização
δ_s	-	Posição de referência para o controle de swing up
σ	-	Posição angular do pêndulo
τ	-	Torque
τ_e	-	Torque elétrico nominal
τ_{eb}	-	Torque de rotor bloqueado
τ_i	-	Torque generalizado do atuador
ϕ	-	Posição angular da roda
cc	-	Corrente contínua
IAE	-	Integrel da magnitude absoluta do erro
IDE	-	Ambiente de desenvolvimento integrado
LQR	-	Regulador quadrático linear
ODE	-	<i>Open dynamics engine</i>
PID	-	Proporcional, integral, derivativo
PIRR	-	Pêndulo invertido por roda de reação
ppr	-	Pulsos por revolução
PWM	-	Modulação por largura de pulso

rpm - Rotações por minuto
SCADA - *Supervisory Control and Data Acquisition*
STA - *Super-twisting algorithm*

Sumário

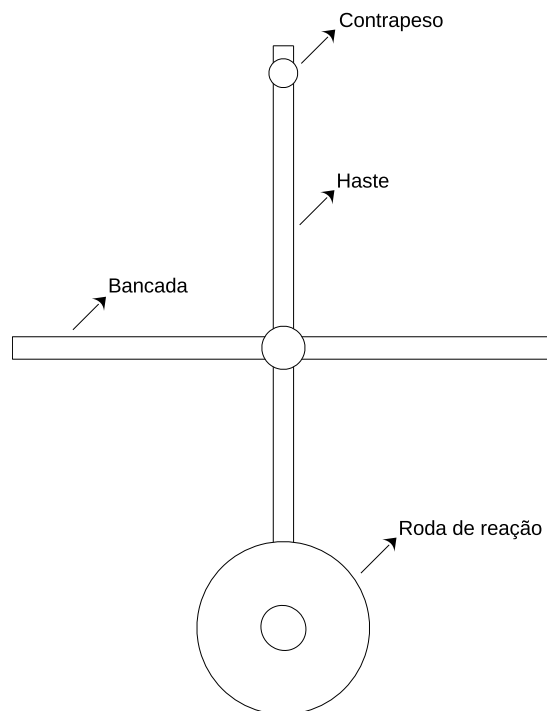
1	INTRODUÇÃO	14
1.1	OBJETIVOS	18
1.2	ORGANIZAÇÃO DO TRABALHO	19
2	PÊNDULO POR RODA DE REAÇÃO	20
2.1	MODELAMENTO DA DINÂMICA PIRR	20
2.1.1	Dinâmicas do motor cc	23
3	MODELO EXPERIMENTAL	25
3.1	AQUISIÇÃO DOS PARÂMETROS DO PÊNDULO	27
3.2	AQUISIÇÃO DOS PARÂMETROS DO MOTOR	28
3.3	ATRITO NO PÊNDULO	29
3.4	FILTRO DE VELOCIDADE	31
4	CONTROLADORES EMPREGADOS NO SISTEMA	33
4.1	CONTROLE POR REALIMENTAÇÃO DE ESTADOS	33
4.2	CONTROLE <i>SWING UP</i>	37
4.3	ESTRATÉGIA DE CONTROLE <i>SWITCH</i>	39
4.4	MALHA DE OPERAÇÃO DO SISTEMA	40
5	RESULTADOS COMPARATIVOS	44
5.1	RESULTADOS COMPARATIVOS COM FILTRO	46
5.2	ÍNDICE DE DESEMPENHO DOS CONTROLADORES	48
5.3	ELIPSE E3	49
5.4	LISTA DE VALORES DOS MATERIAIS UTILIZADOS	50
6	OBSERVAÇÕES FINAIS	52
6.1	CONCLUSÃO	52
6.2	PROPOSTAS DE CONTINUIDADE DO TRABALHO	53
	REFERÊNCIAS	54
A.1	PROGRAMA PARA SIMULAÇÃO SIMULINK	57

A.2	PROGRAMAÇÃO EXPERIMENTAL	58
A.3	PROJETOS SOLIDWORKS	67
A.4	ELIPSE E3	68

1 INTRODUÇÃO

O pêndulo invertido com roda de reação (PIRR), vem sendo amplamente utilizado na área de educação e pesquisa, por ser um sistema dinâmico sub-atuado, não linear e relativamente simples, que o torna desejado para o estudo na área de controle. Introduzido inicialmente por Spong, Corke e Lozano (2001), o pêndulo com roda de reação é uma variante de outros modelos existentes do pêndulo clássico, assim como o pêndulo de Furuta, pendubot, carro-pêndulo e entre outros (BLOCK; ÅSTRÖM; SPONG, 2007). O PIRR consiste basicamente de um pêndulo com uma roda acoplada em um motor na ponta de sua haste, que tem o objetivo de estabilizar o pêndulo na posição invertida, através do momento de inércia gerado pela roda, conforme pode ser visto na Figura 1.

Figura 1 – Pêndulo invertido com roda de reação



Fonte: Autor (2022)

A roda de reação é um dispositivo com a capacidade de armazenar o momento angular

e realizar a transferência de torque de reação através da variação de velocidade angular da roda acoplada a um motor. Além de sua contribuição para efetuar a estabilização e controle em pêndulos, as rodas possuem uma grande importância em sua utilização na área de indústrias aeroespaciais, onde são muito utilizadas em naves espaciais e satélites, devido sua alta precisão em controle de manobras de giro de grandes ângulos também de estabilização de atitude, entre outros (MASTERSON; MILLER; GROGAN, 2002; KIM, 2014). Para solucionar os problemas do controle tolerante de falhas de atitude de naves espaciais, Ran *et al.* (2018) desenvolveram um estudo de um controlador com modo deslizante de segunda ordem adaptativo. O controle de atitude de satélites com apenas duas rodas de reação, ao contrário do uso de pelo menos três rodas de reação, é complexo e possui diferentes pesquisas para contornar essa complexidade; Golzari *et al.* (2020) implementaram um controlador linear variante no tempo preditivo para realizar o controle do satélite com duas rodas de reação. Nesta dissertação é realizado o controle do PIRR utilizando apenas uma única roda de reação.

Para efetuar o controle do PIRR é necessário entender inicialmente os problemas de controlabilidade existentes. O primeiro seria de balançar o pêndulo para levá-lo em sua região invertida, esta ação é denominada *swing up* (SRINIVAS; BEHERA, 2008), e o segundo problema está no controle do pêndulo em seu ponto de equilíbrio nessa mesma região; esse é o ponto de equilíbrio instável. Esses problemas podem ser resolvidos com estratégias de controle que realizam a troca entre controladores, sendo denominado controlador híbrido, ou mesmo controlador que é capaz de efetuar os dois objetivos sem a necessidade de uma estratégia de controle única (HERNANDEZ; SIRA-RAMIREZ, 2003).

Spong, Corke e Lozano (2001) introduziram o pêndulo por roda de reação com dois controladores que necessitam de uma estratégia de controle para se alternarem em condições específicas. Dois métodos são utilizados para o controle de estabilização: linearização por realimentação e linearização aproximada. O *swing up* utiliza um controle de linearização por realimentação parcial em conjunto com função de energia. Uma estratégia de controle híbrida para garantir a estabilidade global do pêndulo por roda de reação foi publicada por Andrievsky (2011), que propõe um projeto de controlador *swing up* que usa o Gradiente de Velocidade Baseado em Energia junto com um controlador por modo deslizantes para a estabilização do pêndulo. Trentin *et al.* (2020) introduziram uma nova variação do pêndulo por roda de reação que utiliza duas rodas de reação. É proposto no trabalho duas formas de controladores não lineares, um controle não linear proporcional-derivativo, e um controle por modo deslizantes.

Sowman, Laila e Longo (2015) desenvolveram um Modelo de Controlador Não linear

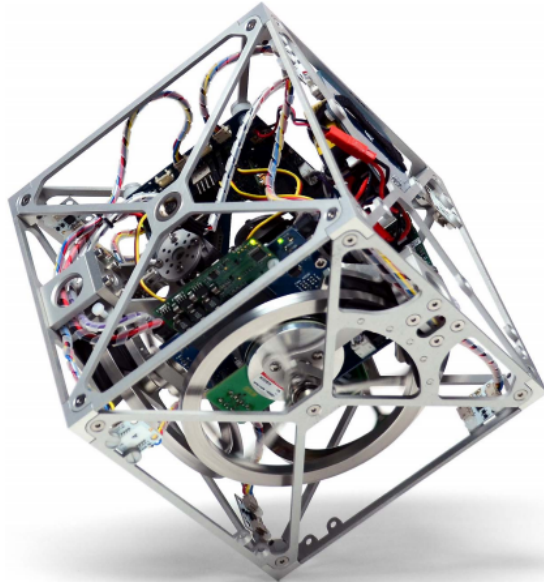
Preditivo para a atuação do *swing up* e do controle do PIRR. O controlador, diferente do que geralmente é utilizado, efetua o *swing up* e a estabilização do pêndulo invertido utilizando somente o Modelo de Controlador Não linear Preditivo, enquanto geralmente é utilizado dois controladores para realizar essas funções. Rizal *et al.* (2018) apresentaram um controlador que utiliza modo deslizantes de segunda ordem para realizar o controle do PIRR com base em um *super-twisting algorithm* (STA). A simulação realizada é baseada em *Open Dynamic Engine* (ODE) utilizada para verificar o controlador proposto, e desenvolvido no *software* Anycode Marilou, que proporciona um ambiente para simulação e modelamento para aplicações robóticas. Lublovary e Insperger (2018) elaboraram um dispositivo experimental do pêndulo com roda de reação para demonstrar os comportamentos de transiente produzidos por dispositivos controlados digitalmente.

Torna-se importante mencionar que o PIRR pode ser linearizado em torno do ponto de equilíbrio na posição invertida, desta maneira é possível utilizar controles lineares para projetar ganhos que permitem sua estabilização. Bapiraju *et al.* (2004) promoveram um estudo de três técnicas clássicas de linearização para o pêndulo com roda de reação ao redor do ponto de equilíbrio, com um controlador de realimentação de estados aplicado ao sistema linearizado, sendo essas técnicas a linearização por série de Taylor de primeira ordem, linearização por realimentação parcial, transformação do modelo com linearização, e também é implementado um controlador de lógica Fuzzy para ser analisado. Olivares e Albertos (2014) introduziram uma estrutura de controle proporcional, integral e derivativo (PID) para o PIRR com um segundo *loop* de controle para contornar o problema de instabilidade interna utilizando um método de controle simples. Um controlador de realimentação de estados é sugerido, sendo um possível candidato para ser considerado no segundo *loop* para evitar o problema de cancelamento de polos causando a instabilidade interna.

Montoya e Gil-González (2020) realizaram a análise não linear e controle baseado na abordagem por Lyapunov. É desenvolvido a partir deste método duas funções diferentes que são comparadas com outras metodologias clássicas de controle do pêndulo com roda de reação na posição invertida e também junto com *swing up*. Os resultados obtidos das simulações utilizando os mesmos ganhos demonstram similaridades em ambos os valores dos controladores lineares e não lineares. Por ser um sistema mecânico sub-atuado possui uma importância significativa por estar presente em um grande número de aplicações; desta maneira tem sido desenvolvido métodos de controles para diferentes sistemas dinâmicos utilizando de maneira análoga o pêndulo por roda de reação. Por exemplo, uma análise não linear e controle de um pêndulo invertido 3-D em formato cúbico

que utiliza três rodas de reação, conforme a Figura 2 (MUEHLEBACH; DíANDREA, 2016).

Figura 2 – Pêndulo invertido 3-D (cubli)



Fonte:(MUEHLEBACH; DíANDREA, 2016)

O pêndulo invertido 3-D possui a característica de saltar de uma posição de descanso para a posição vertical apoiado em seu vértice e se equilibrar. Possui como base, para realizar o salto, duas fases, o modelo de freio com base de impacto para atuar nas rodas de reação para efetuar o salto, e a fase de orientação para guiar o cubo no equilíbrio vertical; também é implementado um algoritmo de aprendizagem para adaptar as velocidades iniciais das rodas de reação. Para o balanceamento assintótico do cubo são utilizadas duas formas de controle, o controlador *backstep* para balanceamento, e o controlador por linearização de realimentação, responsável pelo rastreo de movimentos não equilibrados pré-definidos. Essa analogia também pode ser observada em outras aplicações para estabilização na área de robóticas, que pode ser incluído controles para efetuar o balanceamento de robôs bicicletas, que possuem naturalmente uma característica de instabilidade, assim como mencionado pode ser visto como um pêndulo de reação para realizar a estabilização ao redor da posição invertida, que o mesmo pode ser realizado por um controlador com uma estrutura simples como a de PID (KIM *et al.*, 2013).

Uma contribuição importante para o robôs humanoides é o modelo Pêndulo com Massa de Reação, uma generalização 3D do pêndulo por roda de reação com uma variável de inércia contínua. O principal propósito é de demonstrar como a geometria genérica e a dinâmica de um robô pode ser mapeada no modelo Pêndulo com Massa de

Reação (LEE; GOSWAMI, 2007). Jae-Oh, In-Woo e Jang-Myung (2011) propuseram um método de controle semelhante do pêndulo de reação para um robô monociclo. O robô consiste de um pêndulo de reação móvel, com uma roda responsável pela dinâmica de guinada e a roda de reação pela dinâmica de rolagem; um controlador Fuzzy por Modo Deslizantes é responsável pelo balanceamento dos eixos separadamente. Driessen *et al.* (2019) apresentaram a primeira demonstração prática de uma teoria recente, que tem como objetivo um robô realizar movimentos rápidos e extensos, enquanto se equilibra em um suporte estreito. O robô utilizado no experimento tem um comportamento semelhante a de um pêndulo com roda de reação.

Devido a importância da contribuição do pêndulo por roda de reação e a possibilidade de diversas maneiras de controle, é desenvolvido ao longo deste trabalho um estudo e investigação de estratégias de controles híbridas para elevar o pêndulo para região invertida do pêndulo com um controlador *swing up*, em conjunto com outro controlador de estabilização para efetuar o controle do pêndulo em seu ponto de equilíbrio instável na região invertida.

1.1 OBJETIVOS

O principal objetivo desta pesquisa foi estudar a dinâmica do sistema e investigar estratégias de controle da roda de reação para realização do *swing up* e estabilização da roda no ponto de equilíbrio instável presente na região invertida do pêndulo. É proposto o controle em tempo real do modelo experimental, utilizando ambos controladores junto com o *software* Elipse E3, que se trata de um sistema de supervisão e aquisição de dados (SCADA, do inglês *Supervisory Control and Data Acquisition*).

Os objetivos secundários são apresentados a seguir:

- Adaptar a modelagem da dinâmica do PIRR para compreender seu sistema subatuado não linear, permitindo movimento relativo;
- Construir e modelar um pêndulo com roda de reação, utilizando materiais de baixo custo;
- Escrever a formulação e projetar controladores para elevar e estabilizar o pêndulo em sua região invertida e realizar análise dos dados;
- Comparar o as simulações do modelo teórico com os ensaios realizados do modelo experimental em função de seus parâmetros.

1.2 ORGANIZAÇÃO DO TRABALHO

Esta dissertação está organizada nos seguintes capítulos:

- **Capítulo 1 - INTRODUÇÃO:** Apresenta a motivação e revisão bibliográfica abordando as diversas contribuições para o controle do pêndulo por roda de reação, e também os objetivos.
- **Capítulo 2 - PÊNDULO POR RODA DE REAÇÃO:** É abordado o modelo matemático do PIRR e do motor implementado.
- **Capítulo 3 - MODELO EXPERIMENTAL:** É feita a definição do dispositivo experimental, aquisição dos parâmetros físicos mais o atrito para considerar nas simulações, e também o filtro implementado no sistema.
- **Capítulo 4 - CONTROLADORES EMPREGADOS NO SISTEMA:** São definidos os controladores e estratégia de controle de que serão implementadas no sistema e também é apresentado a malha de operação e ligação elétrica do experimento de bancada.
- **Capítulo 5 - RESULTADOS COMPARATIVOS:** O capítulo aborda os resultados obtidos tanto em bancada como simulação computacional, e é feita a comparação entre ambos modelos. O índice de desempenho dos controladores são comparados, e apresentado o *software* SCADA utilizado para controle em tempo real dos controladores.
- **Capítulo 6 - OBSERVAÇÕES FINAIS:** Realiza a conclusão do trabalho, e propõe os trabalhos futuros.

2 PÊNDULO POR RODA DE REAÇÃO

Na Seção 2.1 serão abordadas as equações de movimento que descrevem o PIRR, assim como o modelo linearizado e o motor de corrente contínua utilizado.

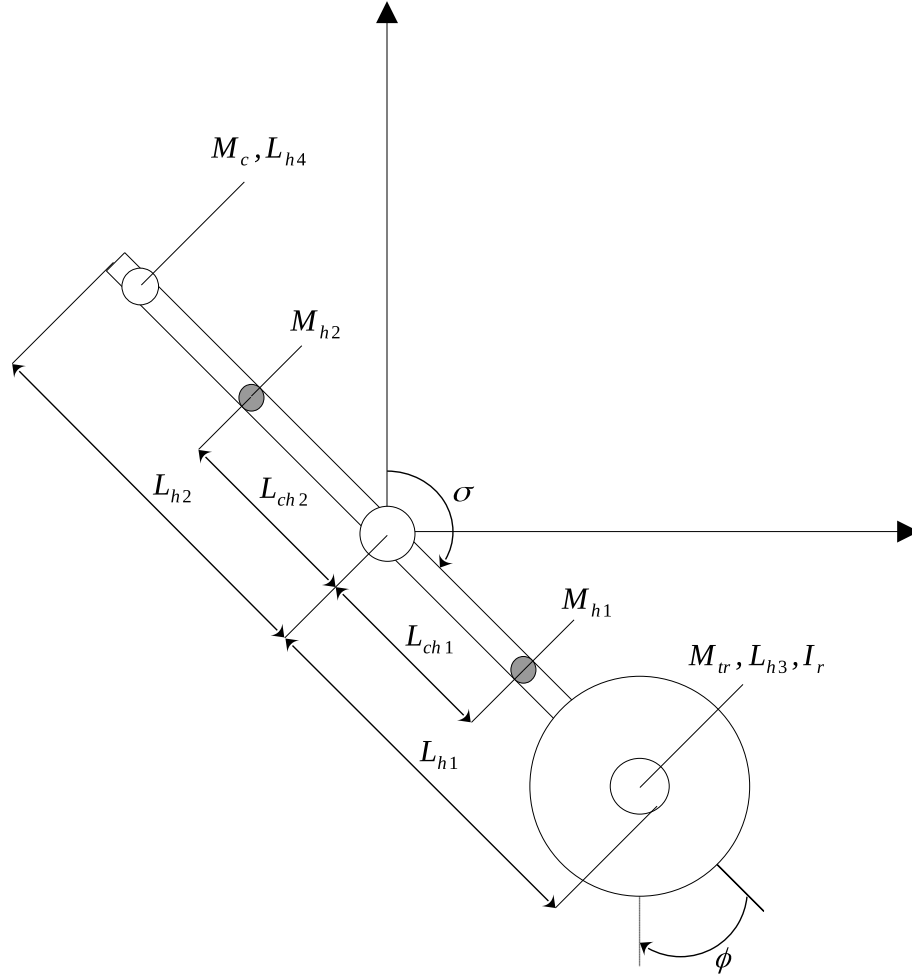
2.1 MODELAMENTO DA DINÂMICA PIRR

A dinâmica que constitui o PIRR é exibida na Figura 3, também é demonstrado o modelamento dinâmico que utiliza como base a função de Lagrange. É levado em consideração o sistema de coordenadas generalizadas, com $\sigma(t)$ sendo a posição do pêndulo e $\phi(t)$ a posição da roda. Além disso, é ilustrado o momento de inércia da roda I_r , sendo o torque $\tau(t)$ aplicado na roda de reação pelo motor. É importante mencionar que o momento de inércia da roda foi obtido através de *software*.

Os parâmetros M_a , M_c , M_h , M_m e M_r representam as massas de cada componente, sendo M_a é a massa do acoplamento da roda com o motor, M_c a massa do contrapeso, $M_{hk}, k = 1, 2$ a massa das hastes no qual $M_{h1} = M_{h2}$, M_m a massa do motor e M_r representa a massa da roda. O comprimento da haste a partir do pivô é dado por $L_{hi}, i = 1, 2, 3, 4$ no qual L_{h1} é o comprimento da haste da roda com relação ao pivô, L_{h2} idem para o contrapeso, L_{h3} é a distância do centro de massa do conjunto de massas na roda em relação ao pivô e L_{h4} idem para o contrapeso, sendo que $\frac{L}{2} = L_{h1} = L_{h2}$.

Por fim é definida a distância do centro de massa das hastes em relação ao pivô, que é dada por $L_{chj}, j = 1, 2$ em que $L_{ch1} = L_{ch2}$. Além dessas variáveis de dinâmica, são introduzidas as seguintes quantidades para simplificar o arranjo de equações de dinâmica do pêndulo.

Figura 3 – Ilustração do modelo do PIRR



Fonte: Adaptado de Fantoni, Lozano e Sinha (2002)

$$\begin{aligned}
 M_{tr} &= M_a + M_m + M_r, \\
 I_p &= \frac{1}{3} M_{h1} L_{h1}^2 + \frac{1}{3} M_{h2} L_{h2}^2 + M_{tr} L_{h3}^2 + M_c L_{h4}^2 + I_r, \\
 \bar{M}_1 &= M_{h1} L_{ch1} + M_{tr} L_{h3}, \\
 \bar{M}_2 &= M_{h2} L_{ch2} + M_c L_{h4}.
 \end{aligned} \tag{1}$$

A leitura do ângulo do pêndulo é medida a partir da posição de equilíbrio instável vertical superior. Os ângulos do PIRR são obtidos pela leitura do encoder incremental implementado no centro da haste do pêndulo, e pelo encoder do motor, responsável pelas leituras da roda de reação, anexado junto com a roda em uma extremidade da haste.

Para realizar o modelamento, é aplicada a equação de movimento de Lagrange, no qual utiliza um conjunto de coordenadas generalizadas do sistema mecânico escolhido. No sistema estão presentes a energia cinética do pêndulo, rotor e motor; a soma dessas

energias representa a energia cinética total que está apresentada em (2).

$$E_{ct} = \frac{1}{2}I_p\dot{\sigma}^2(t) + I_r\dot{\phi}(t)\dot{\sigma}(t) + \frac{1}{2}I_r\dot{\phi}^2(t). \quad (2)$$

A energia potencial obtida, desconsidera a elasticidade presente no eixo do motor ou na haste do pêndulo, sendo apenas a energia potencial gravitacional considerada, como visto em (3).

$$E_{pg} = \bar{M}_1g(\cos(\sigma(t)) - 1) + \bar{M}_2g(1 - \cos(\sigma(t))). \quad (3)$$

Como a haste possui massa aproximadamente semelhante nos dois extremos, causada pelos encaixes na haste, suas diferenças resulta próximo de 0, devido a isso é desconsiderada da equação. Com as equações de energia (2) e (3), aplica-se a função de Euler-Lagrange, que descreve o sistema a partir das coordenadas generalizadas, conforme descrito em (4).

$$L_{ag}(q_i, \dot{q}_i, t) = E_{ct}(q_i, \dot{q}_i, t) - E_{pg}(q_i), \quad (4)$$

$$L_{ag} = \frac{1}{2}I_p\dot{\sigma}^2(t) + I_r\dot{\phi}(t)\dot{\sigma}(t) + \frac{1}{2}I_r\dot{\phi}^2(t) + \bar{M}_1g(1 - \cos\sigma(t)) + \bar{M}_2g(\cos\sigma(t) - 1).$$

Por fim, é realizada a derivada parcial da (4) aplicando-a em (5).

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \tau_i, \quad (5)$$

sendo $\tau_i(t)$ o torque generalizado do atuador. Um fator importante a ser mencionado, é que há um torque $\tau(t)$ sendo exercido no rotor do motor, enquanto isso um torque de sentido oposto $-\tau(t)$ é aplicado ao pêndulo. Desta maneira resulta nas equações de movimento (6) e torque (7), que descrevem o sistema do PIRR.

$$I_p\ddot{\sigma}(t) + I_r\ddot{\phi}(t) - \bar{M}_1g\sin\sigma(t) + \bar{M}_2g\sin\sigma(t) = 0, \quad (6)$$

$$I_r\ddot{\sigma}(t) + I_r\ddot{\phi}(t) = \tau(t). \quad (7)$$

A partir desse sistema de equações, é possível isolar (6) em função de $\ddot{\sigma}$ e (7) de $\ddot{\phi}$, para então, desta maneira, obter o sistema em função da aceleração do pêndulo e da roda de reação, através de substituições nas equações. Assim pode ser calculado os estados do

sistema com (8) a partir do *software* Matlab Simulink apresentado na figura A.1.1.

$$\begin{aligned}\ddot{\sigma} &= \frac{1}{I_p I_r - I_r^2} (I_r \bar{M}_1 g \sin \sigma(t) - I_r \bar{M}_2 g \sin \sigma(t) - I_r \tau(t)), \\ \ddot{\phi} &= \frac{1}{I_p I_r - I_r^2} (I_p \tau(t) + I_r \bar{M}_2 g \sin \sigma(t) - I_r \bar{M}_1 g \sin \sigma(t)).\end{aligned}\tag{8}$$

A equação final pode ser linearizada em torno do ponto de operação, pois para valores de σ próximos de 0, o $\sin(\sigma)$ resulta no próprio σ . Possibilitando, desta maneira, utilizar a equação linearizada para projeto de controle linear. A equação linearizada é descrita por (9).

$$\begin{aligned}\ddot{\sigma} &= \frac{1}{I_p I_r - I_r^2} (I_r \bar{M}_1 g \sigma(t) - I_r \bar{M}_2 g \sigma(t) - I_r \tau(t)), \\ \ddot{\phi} &= \frac{1}{I_p I_r - I_r^2} (I_p \tau(t) + I_r \bar{M}_2 g \sigma(t) - I_r \bar{M}_1 g \sigma(t)).\end{aligned}\tag{9}$$

2.1.1 Dinâmicas do motor cc

É necessário que a dinâmica do motor cc seja implementada na equação de movimento do PIRR a partir de um rearranjo de equações, para que através do sistema em malha fechada possibilite fornecer a tensão elétrica necessária para o sistema com base na entrada de controle. Assim as equações que descrevem a dinâmica do motor corrente contínua (cc), considerando o fluxo constante, podem ser descritas a partir de (10) e (11).

$$\tau(t) = n K_t I_a(t) - A_m \dot{\phi}(t),\tag{10}$$

sendo K_t a constante de torque e A_m o atrito presente nas escovas do motor cc. A corrente de armadura representada por $I_a(t)$ é obtida em (11).

$$V(t) = R_a I_a(t) + n K_v \dot{\phi}(t),\tag{11}$$

em que $V(t)$ é a tensão a ser aplicada ao motor cc, n a redução do motor, R_a a resistência de armadura e K_v a constante de força eletromotriz. A indutância presente no motor é desprezada das equações para o acoplamento das dinâmicas. Isolando (11) em função da corrente de armadura e aplicando-a na (10), resulta em (12).

$$\tau(t) = \frac{n K_t V(t)}{R_a} - \frac{n K_t n K_v \dot{\phi}(t)}{R_a} - A_m \dot{\phi}(t),\tag{12}$$

que representa o torque em função da tensão de alimentação, que foi definida como a entrada de controle do sistema. Por fim, substituindo a equação (12) em (9) para obter (13), que será utilizada para descrever o sistema em espaços de estados.

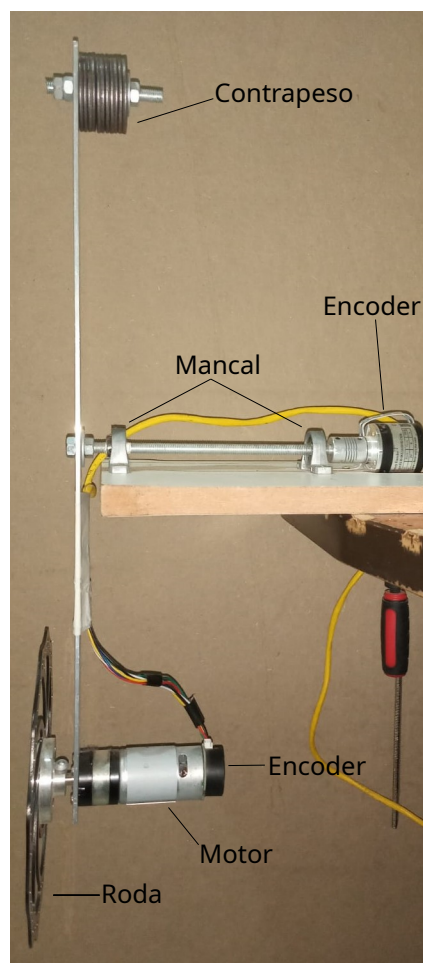
$$\begin{aligned}\ddot{\sigma} &= \frac{I_r \bar{M}_1 g \sigma(t)}{C_i} - \frac{I_r \bar{M}_2 g \sigma(t)}{C_i} - \frac{I_r n K_t V(t)}{C_i R_a} + \frac{I_r n^2 K_t K_v \dot{\phi}(t)}{C_i R_a} + \frac{I_r A_m \dot{\phi}(t)}{C_i}, \\ \ddot{\phi} &= \frac{I_r \bar{M}_2 g \sigma(t)}{C_i} - \frac{I_r \bar{M}_1 g \sigma(t)}{C_i} + \frac{I_p n K_t V(t)}{C_i R_a} - \frac{I_p n^2 K_t K_v \dot{\phi}(t)}{C_i R_a} - \frac{I_p A_m \dot{\phi}(t)}{C_i},\end{aligned}\tag{13}$$

sendo $C_i = I_p I_r - I_r^2$ uma simplificação para as constante de inércia.

3 MODELO EXPERIMENTAL

Para a montagem do aparato experimental foram utilizados materiais e *hardwares* de fácil disponibilidade e de baixo custo, para servirem como base e atuarem como controladores em tempo real. A parte lateral do modelo pode ser observada na Figura 4, que demonstra sua montagem com mancais, um fuso trapezoidal de inox, e equipamentos utilizados para obtenção dos parâmetros e atuação pelo controlador.

Figura 4 – Foto lateral PIRR

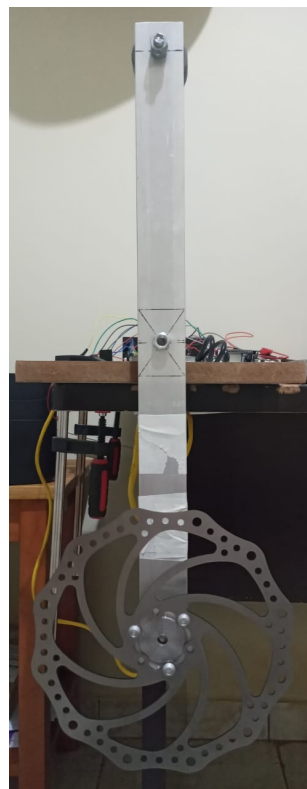


Fonte: Autor (2022)

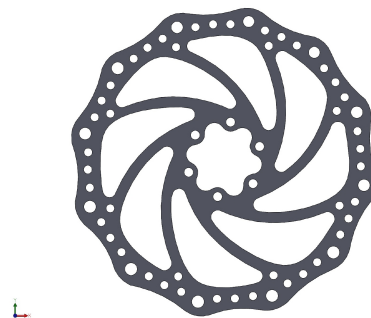
Os equipamentos que compõem o modelo experimental conta com um encoder incremental de 1000 ppr (pulsos por revolução). Este opera na faixa de 8 à 24 V com limite de até 3000 rpm, que é encarregado de efetuar as leituras de posição e velocidade do pêndulo em cada instante de tempo. Um motor cc do modelo CHP-36GP-555-ABHL de 12 V, que consegue atingir 880 rpm em plena carga, sendo capaz de fornecer um torque de $0,29 \text{ Nm}$; o motor já possui um encoder Holzer acoplado em si, que opera na faixa de 3,3 à 5 V em 88 ppr, com o propósito de realizar somente a leitura de velocidade do motor.

Na Figura 5 é apresentada a parte frontal do modelo mostrando o disco de freio de bicicleta, que é utilizado para desempenhar a função da roda de reação, para gerar um momento maior, onde sua composição é de inox e possui $0,203 \text{ m}$ de diâmetro e $0,002 \text{ m}$ de espessura, juntamente com a haste de alumínio, que possui $0,50 \text{ m}$ de comprimento, $0,038 \text{ m}$ de diâmetro e $0,003 \text{ m}$ de espessura.

Figura 5 – Foto frontal PIRR e Roda no Solidworks



(a) Foto do pêndulo



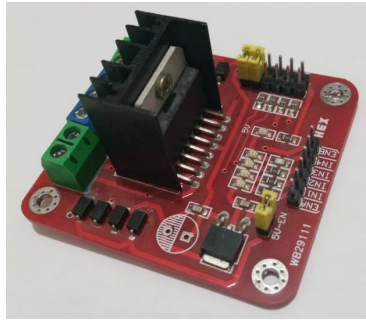
(b) desenho da roda no Solidworks

Fonte: Autor (2022)

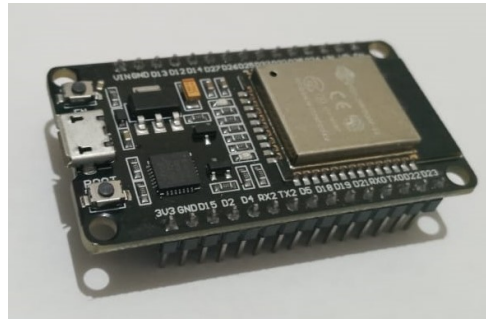
Na Figura 6 são demonstrados os componentes utilizados no controle que conta com apenas 3 dispositivos de baixo custo para controlar o motor cc. O microcontrolador utilizado é um ESP32 Wroom Devkit, que pode ser programado na mesma plataforma

do arduíno, e uma ponte H L209N, responsável pelo acionamento do motor, através modulação por largura de pulsos (PWM). O encoder incremental é responsável por aquistar as posições e velocidades relacionadas ao pêndulo, a partir de pulsos gerado pelo o mesmo.

Figura 6 – Microcontrolador, driver e encoder rotacional



(a) Ponte H L298N



(b) Microcontrolador Esp-32 Wroom



(c) Encoder

Fonte: Autor (2022)

3.1 AQUISIÇÃO DOS PARÂMETROS DO PÊNDULO

A aquisição dos parâmetros do modelo experimental do PIRR deve ser estimada a fim de ser utilizados para as simulações numéricas teóricas. Por fim, a comparação destes com o modelo experimental será feita, a fim de validar as estratégias de controle adotadas. Contudo, devido à aproximações ou mesmo parâmetros que não foram considerados no modelamento, é de se esperar certas discrepâncias entre os dados obtidos. O momento de inércia do pêndulo foi calculado a partir dos extremos do pêndulo e as respectivas massas de cada componente, foram identificadas separadamente, m_i , $i = a, c, h, m, r$.

Em seguida, obteve-se os momento de inércia da roda; é importante comentar que o momento de inércia da roda I_r foi identificado pelo o *software* SolidWorks, que é uma ferramenta que permite a criação e renderização de objetos em 3D, simulação de movimentos, projetos de animação, entre outros. Desta maneira, foi feita a modelagem

3D do componente por medidas e aproximações; o modelo é exibido no apêndice A, pela Figura A.3.2. A razão por ter sido determinado pelo *software*, deve-se ao elemento de inox vazado e curvas presentes na roda, que torna difícil o cálculo de seus respectivos momentos de inércia de maneira analítica. Na Tabela 1 são demonstrados os parâmetros físicos utilizados para a simulação do PIRR.

Tabela 1 – Parâmetros físicos do PIRR

Símbolos	Parâmetros	Valores
L	Comprimento da haste $[m]$	0,50
$L_{hi}, i = 1, 2$	Comprimento da haste em relação ao pivô $[m]$	0,25
$L_{chj}, j = 1, 2$	Distância do pivô ao centro de massa da haste $[m]$	0,125
L_{h3}	Distância do pivô ao centro de massa da roda $[m]$	0,2278
L_{h4}	Distância do pivô ao centro de massa do contrapeso $[m]$	0,2272
M_c	Massa do contrapeso $[kg]$	0,350
M_h	Massa da haste $[kg]$	0,160
$M_{hk}, k = 1, 2$	Massa de meia haste $[kg]$	0,08
M_m	Massa do motor $[kg]$	0,390
M_r	Massa da roda $[kg]$	0,190
M_a	Massa do acoplamento da roda $[kg]$	0,065
I_r	Momento de inércia da roda $[kgm^2]$	$11,18 \times 10^{-4}$
g	Aceleração da gravidade $[m/s^2]$	9,81

Fonte: Autor(2022)

3.2 AQUISIÇÃO DOS PARÂMETROS DO MOTOR

Para obter os parâmetros do motor foi utilizado o *datasheet* do modelo CHP-36GP-555-ABHL, que pode ser visto na Tabela 2. Como apenas alguns parâmetros estavam disponíveis ou apresentavam incompatibilidade entre si, foi necessário calculá-los pelas equações do motor cc de ímã permanente. Vale mencionar que alguns dados foram calculados ou testados experimentalmente, para avaliar a confiabilidade com o *datasheet*, e mesmo que não demonstrem exatidão com os valores fornecidos, possuem uma boa aproximação, com exceção da constante de torque e constante de campo.

Considerando o fluxo de campo constante, de acordo com (11), foi elaborado um sistema de equações considerando dados com torque nominal e torque à vazio. Desta maneira, foi possível aquisitar as constantes do motor cc, a resistência de armadura R_a , constante de torque K_t e de campo K_v . Contudo, as constantes de torque e campo eram incompatíveis com o torque elétrico atribuído pelo *datasheet*, desta forma foi decidido utilizar as constantes encontradas através dos cálculos. Na Tabela 2 são apresentados os

valores obtidos dos dados do motor cc de imã permanente em 12 V. Como o *datasheet* não oferecia os dados para o coeficiente de atrito do motor, seu valor teve que ser estimado de maneira empírica, utilizando o controlador *swing up*. Diferentes valores para o coeficiente foram testados no modelo teórico, e analisando de tal forma que o resultado final fosse semelhante com o do experimento prático.

Tabela 2 – Parâmetros físicos do motor cc de imã permanente

Símbolos	Parâmetros	Datasheet	Aquisitado
A_m	Coeficiente de atrito do motor $[\frac{Nm}{rad/s}]$	–	0,0014
I_a	Corrente de armadura à vazio $[A]$	$\leq 0,40$	0,28
$\dot{\phi}$	Velocidade à vazio [rpm]	1150	1100
τ_e	Torque elétrico nominal $[Nm]$	0,2941	0,1882
$\dot{\phi}$	Velocidade nominal [rpm]	880	-
I_a	Corrente de armadura nominal $[A]$	$\leq 2,0$	-
τ_{eb}	Torque de rotor bloqueado $[Nm]$	$\geq 7,0$	-
I_{ab}	Corrente de armadura de rotor bloqueado $[A]$	$\leq 9,5$	-
R_a	Resistência armadura $[\Omega]$	-	1,6632
K_t	Constante de torque $[Nm/A]$	-	0,0181
K_v	Constante de campo $[\frac{V}{rad/s}]$	-	0,0181
n	Redução do motor	5,2	-

Fonte: Autor (2022)

Os dados aquisitados com a roda acoplada foram considerados à vazio, pois a diferença que ocorreu no motor com o acoplamento foram baixas, e longe de serem consideradas em nominal.

3.3 ATRITO NO PÊNDULO

Quando se está trabalhando com um experimento real, que envolve movimento oscilatório, duas formas de atritos podem se manifestar; quando isto ocorre, a oscilação do pêndulo passa ter uma característica em seu comportamento de não linearidade, desta maneira, se torna indispensável de obter os coeficientes de atrito para projetá-los na simulação (ZONETTI *et al.*, 1999). O pêndulo possui baixa velocidade angular e o atrito de coloumb, F_c , atua em módulo constante em baixas velocidades; as contribuições para este atrito são causadas principalmente de movimentações das partes mecânicas, assim como o atrito presente entre os mancais e no encoder utilizado no experimento. O atrito viscoso, F_v , é proporcional à velocidade e ocorre devido ao atrito com um fluido, no caso do experimento o ar (BUTIKOV, 2015). A equação considerando ambos atritos é descrita em (14).

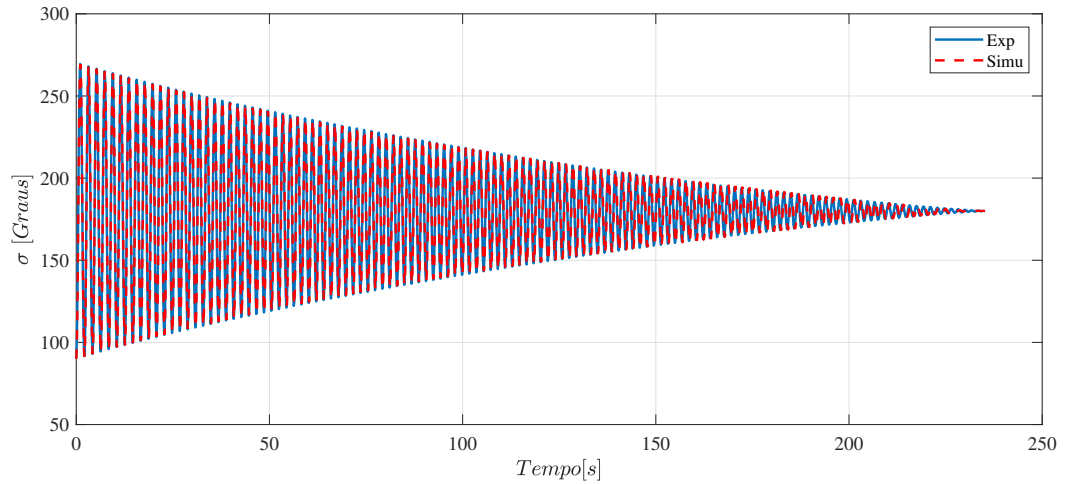
$$F_{rwp} = \text{sgn}(\dot{\sigma}(t))(F_v \mid \dot{\sigma}(t) \mid F_c). \quad (14)$$

Os valores dos atritos foram identificados empiricamente. O pêndulo tem sua referência no ponto de equilíbrio estável em 180° . Assim, o pêndulo é solto em uma posição de $\sigma(t) = 90^\circ$, e desta maneira o encoder comunicava o microcontrolador com as posições a cada instante de tempo até atingir seu ponto de equilíbrio. A equação considerando o atrito é descrita por (15).

$$I_p \ddot{\sigma}(t) + I_r \ddot{\phi}(t) = \bar{M}_1 g \sin \sigma(t) - \bar{M}_2 g \sin \sigma(t) - F_{rwp}. \quad (15)$$

Com base nos dados obtidos pelo ensaio, como mostrado pela Figura 7, obteve-se a identificação dos atritos utilizados.

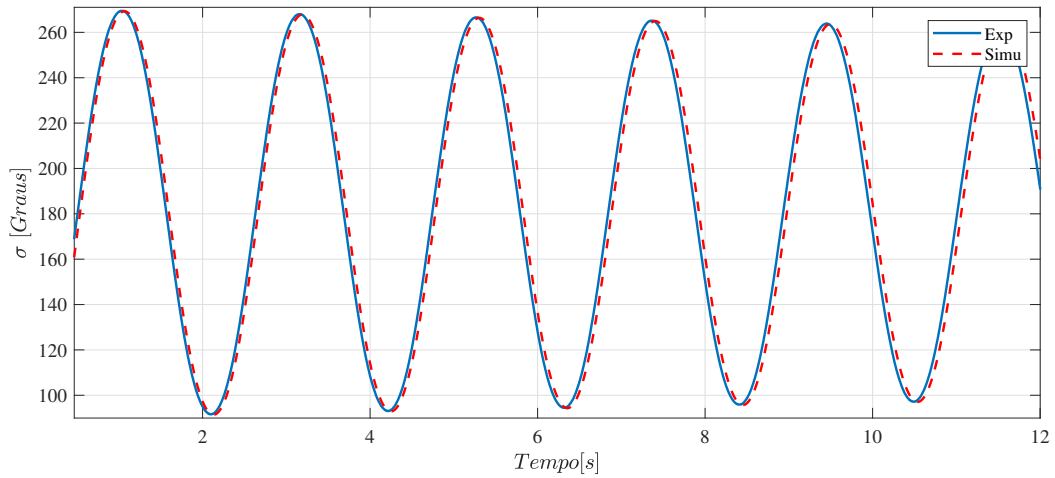
Figura 7 – Variação do ângulo da haste (σ) para estimar o atrito do pêndulo



Fonte: Autor (2022)

Os atritos identificados $F_v = 0,000425 \frac{Nm}{Rad/s}$ e $F_c = 0,001165 Nm$ são consideravelmente baixos, como pode ser observado pela Figura 7, os resultados utilizando a (15) é representada pela curva vermelha, sendo os resultados obtidos pelo modelo teórico, e a curva azul, os resultados do experimento de bancada. Devido ao número elevado de amostras causadas pelas oscilações, é apresentada a Figura 8 para uma análise mais precisa do gráfico, com um número menor de amostras.

Figura 8 – Aproximação gráfica das amostas do ensaio



Fonte: Autor (2022)

O pêndulo possui um momento de inércia considerável no extremo da haste causada pela massa dos componentes principalmente a do motor, que pode comprometer um pouco a controlabilidade do controlador, devido a isso houve necessidade de implementar um contrapeso no sistema. O modelo apresenta uma ótima resposta, e uma defasagem muito baixa que se torna visível no fim de sua oscilação, deve-se em consequência de uma pequena diferença em algum parâmetro.

3.4 FILTRO DE VELOCIDADE

Os sinais de velocidade do modelo experimental, são aquisitados a partir da derivada da posição, ou seja, é calculada a variação através de dois pontos de posição do sensor em um dado tempo de amostragem predefinido. Porém, isto pode gerar oscilações que podem impactar de maneira significativa no controlador, dito isso, é importante que seja inserido um filtro para reduzir essas oscilações causada por essas incertezas produzidas pela derivada (BLOCK; ÅSTRÖM; SPONG, 2007). O filtro que será utilizado no sistema, é de primeira ordem com o intuito de reduzir as frequências elevadas presentes na velocidade. Primeiramente é definida uma relação entrada/saída na qual.

$$Y_f(s) = \frac{1}{1 + sT} Y(s), \quad (16)$$

sendo $Y_f(s)$ e $Y(s)$ o sinal de velocidade filtrado e calculado, com T como constante de

tempo de filtragem. Aplicando a transformada inversa de Laplace obtém-se a equação diferencial:

$$y = T \frac{dy_f}{dt} + y_f. \quad (17)$$

Para simplificação da equação diferencial e sua resolução futura, é aplicado o método das diferenças finitas utilizando a fórmula das diferenças regressivas, desta forma resulta em (18).

$$y(t) = \frac{T}{h_v} (y_f(t) - y_f(t - h_v)) + y_f(t), \quad (18)$$

no qual o h_v representa o intervalo, ou tempo de amostragem da velocidade. Simplificando e rearranjando novamente as equações, obtém-se (19).

$$y(t) = \frac{T + h_v}{h_v} y_f(t) - \frac{T}{h_v} y_f(t - h_v), \quad (19)$$

sendo a equação simplificada com a presença do filtro mostrada em (20).

$$y_f(t) = \frac{h_v}{T + h_v} y(t) + \frac{T}{T + h_v} y_f(t - h_v). \quad (20)$$

Desta maneira será possível reduzir essas oscilações produzidas na velocidade que são causadas por ruídos do sinal calculado a partir dos sensores.

4 CONTROLADORES EMPREGADOS NO SISTEMA

Para a estabilização do pêndulo, é empregado um controle por realimentação de estados com um controlador *swing up* e um algoritmo para a troca dos controladores, que será explicado ao longo dessa seção.

4.1 CONTROLE POR REALIMENTAÇÃO DE ESTADOS

Uma das leis de controle que possibilita a convergência para estabilidade é o método que funciona a partir da fundamentação de alocação de polos em malha fechada, em que o produto de um ganho com os estados do sistema são obtidos, permitindo desta maneira alocar os polos para o semiplano esquerdo (RYBOVIC; PRIECINSKY; PASKALA, 2012). Entretanto, um dos principais problemas é a necessidade de ter a acessibilidade dos estados a serem realimentados, sejam eles obtidos ou estimados.

Os estados controlados do sistema são medidos a partir dos *encoders*: a posição do pêndulo, definido por σ , a velocidade do pêndulo $\dot{\sigma}$, e a velocidade da roda de reação $\dot{\phi}$. Vale mencionar que a posição da roda, é um estado que pode ser desconsiderado no controle do sistema. A representação em espaços de estados do modelo linearizado do sistema, descrito em (13), será usada para o projeto dos ganhos do controlador. A representação $\dot{x}(t) = Ax + Bu$, pode ser observada em (21), sendo $x(t)$ os estados, no qual $x(t) = [\sigma \ \dot{\sigma} \ \dot{\phi}]^T$.

$$\begin{bmatrix} \dot{\sigma} \\ \ddot{\sigma} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{(I_r \bar{M}_1 g - I_r \bar{M}_2 g)}{C_i} & 0 & \frac{I_r n^2 K_t K_v + I_r A_m R_a}{C_i R_a} \\ \frac{(I_r \bar{M}_2 g - I_r \bar{M}_1 g)}{C_i} & 0 & -\frac{I_p n^2 K_t K_v + I_p A_m R_a}{C_i R_a} \end{bmatrix} \begin{bmatrix} \sigma \\ \dot{\sigma} \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{I_r n K_t}{C_i R_a} \\ \frac{I_p n K_t}{C_i R_a} \end{bmatrix} u, \quad (21)$$

tendo como saída do sistema o próprio vetor de estados (todos com acesso através de medições de sensores). Em suma, os estados devem ser realimentados por um ganho K ,

permitindo alocação em valores desejados dos polos de malha fechada. Neste caso, os valores desejados para os estados estão nas proximidades da origem, assim $x = [0, 0, 0]^T$, portanto é necessário identificar os ganhos que possibilitem estabilizar o pêndulo (NISE, 2020).

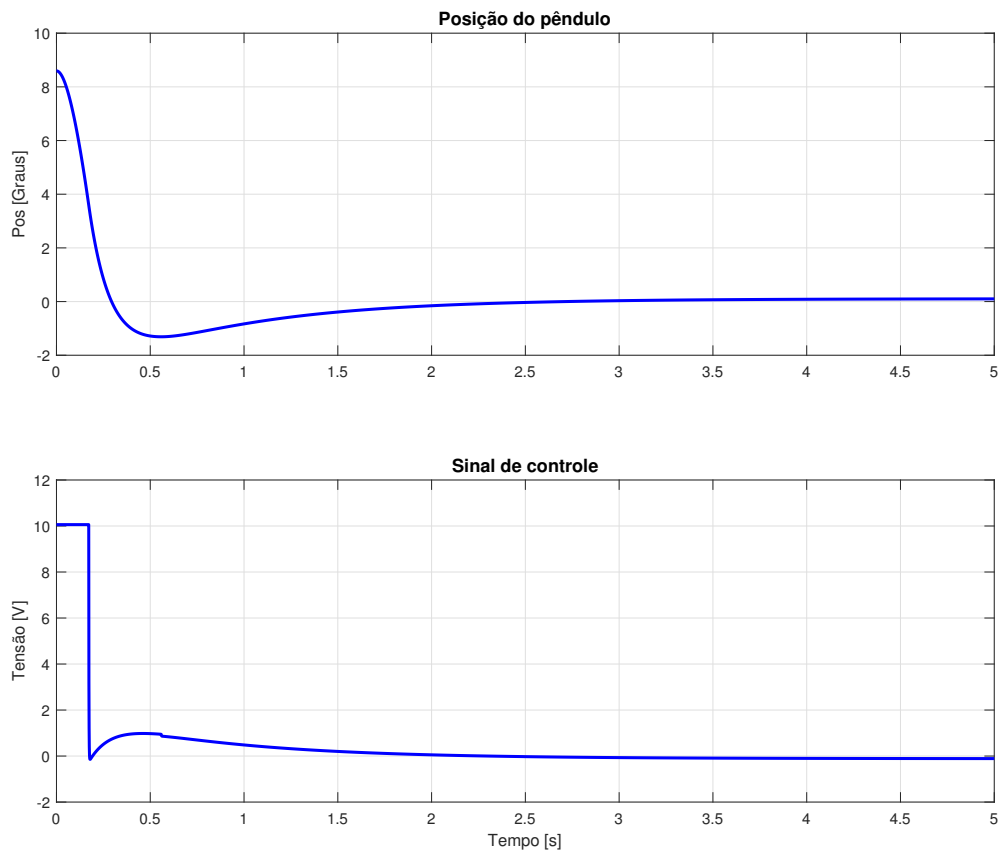
Os polos para o sistema em malha aberta, foram identificados, sendo eles $P_a = [3, 44 - 6, 19 - 3, 39]$. Como um dos polos identificados possui parte real positiva, é correto afirmar que o sistema é instável, sendo necessário projetar ganhos para alocar esses polos e deixá-los todos com parte real negativa. Para implementar um controlador por realimentação de estados, será desenvolvido um ganho com três elementos para multiplicar com os estados. A entrada de controle $u_1(t)$ dada em volts, utilizada para controlar o pêndulo sobre o ponto de equilíbrio instável, é descrita em (22).

$$u_1(t) = -Kx(t). \quad (22)$$

A matriz de ganhos foi obtida a partir dos polos alocados no ponto de interesse, sendo eles $P_f = [-2210, 37 - 8, 67 - 1, 301]$. Tem-se o intuito de estabilizar, o estado posição do pêndulo σ , no ponto de interesse. Esses ganhos fazem o sistema se comportar de maneira semelhante que um de segunda ordem. E desta forma resultou na seguinte matriz de ganhos $K = [-21401, 90 - 4201, 09 - 41, 01]$, sendo o primeiro elemento multiplicado pelo estado σ , o segundo por $\dot{\sigma}$, e o terceiro por $\dot{\phi}$. Foi realizada a simulação do modelo não linear, no qual foi substituído (12) em (8), e usando (22) em substituição do sinal da tensão dada em (12). Para verificar apenas o desempenho do controle no modelo teórico, pfoi estabelecido uma condição inicial aproximada de C.I = $[8, 6^\circ \ 0 \ 0]^T$ para os estados do sistema, para analisar a posição do pêndulo durante a atuação do sinal de controle, como pode ser observado na Figura 9 que demonstra o controle de posição, e o sinal de controle no sistema ideal sem ruídos.

O sinal de controle representa a tensão de alimentação do motor; para simular o comportamento experimental, um saturador foi implementado para manter a relação de tensão de alimentação máxima em 10,06 V. Os gráficos apresentaram uma boa resposta, possibilitando a estabilização do pêndulo na posição invertida, no entanto, nos primeiros instantes de tempo, o controlador atingiu a saturação.

Figura 9 – Gráficos do controle de estabilização

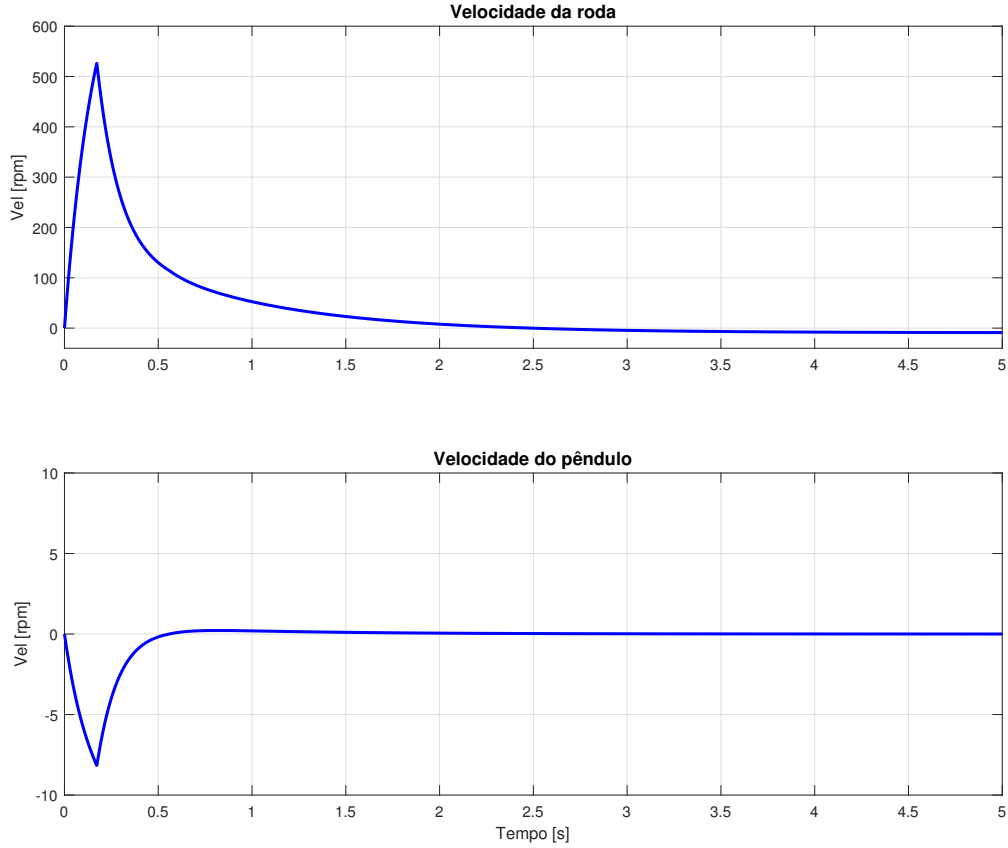


Fonte: Autor (2022)

Os estados de velocidade também são obtidos. Apesar do sinal de controle apresentar saturação, a velocidade não apresentou saturação, dada a limitação mecânica do motor, conforme a Figura 10. No sistema ideal, diferentes valores para C.I foram testados para encontrar a distância em que ocorre a saturação do motor, a configuração do modelo PIRR, implementado no momento da simulação, resultou em saturação após $11,45^\circ$.

Tal efeito pode ser reduzido caso seja utilizado o motor em operação 12 V, aumentando a massa do contrapeso, entre outros; o principal motivo na agravação deste efeito está na massa dos componentes e características do motor que inclusive foi o principal ponto para implementar o contrapeso. Pode ser implementado um motor com sensores e especificações superiores a fim de produzir melhores resultados, mas isso gera custo adicionais.

Figura 10 – Gráfico de velocidade do controle de estabilização



Fonte: Autor (2022)

É importante mencionar que o estado relacionado com a posição do pêndulo, calculado no controlador $\delta(t)$, é diferente do utilizado para análise de dados $\sigma(t)$. As posições do pêndulo, por se tratar de um círculo trigonométrico, produz certo problemas na leitura, como identificar se a posição 360° é a mesma de 0° , fazendo com que o controlador não atue nessa região e dificultando o seu funcionamento em conjunto com o controlador *swing up*. Assim foi estabelecida uma regra que altera a forma do sinal lida pelo controlador; como o modelo dinâmico do pêndulo utilizado tem uma condição inicial de 180° , para valores $> 180^\circ$ a regra considera sua saída como $\sigma(t) - 2\pi$, e para valores $< 180^\circ$ a saída é igual posição $\sigma(t)$. É demonstrada esta equação condicional em (23).

$$\delta(t) = \begin{cases} \sigma(t) - 2\pi & \text{se } \sigma(t) > \pi \\ \sigma(t) & \text{se } \sigma(t) < \pi. \\ \pi & \text{se não} \end{cases} \quad (23)$$

Desta maneira ambos lados tendem para 0° de acordo com a amplitude de σ , possibilitando que o controlador atue tanto próximo à 360° como 0° , ou seja, com a haste vindo da esquerda ou direita do ponto de equilíbrio instável (posição vertical em zero graus).

4.2 CONTROLE *SWING UP*

Um dos principais problemas encontrado no sistema do PIRR é elevar o pêndulo para a região de estabilidade invertida, utilizando pouca energia. A força necessária do motor para impulsionar o pêndulo, somente com o controle de estabilização, é muito alta, tornando um processo inviável. Contudo, com a adição de um controle específico, como o *swing up* o pêndulo consegue atingir a região invertida, utilizando a própria energia acumulada do movimento oscilatório para proporcionar o movimento do pêndulo. O método *swing up*, empregado no sistema apresentado por (24), tem o propósito de proporcionar a energia necessária para que o pêndulo alcance a vizinhança da posição de equilíbrio instável, inspirado nos métodos de controle de energia (ÅSTRÖM; FURUTA, 2000; XIN; KANEDA, 2002).

A energia potencial do pêndulo não controlado é definida por $E_p = \frac{1}{2}I_s\dot{\sigma}^2(t) + mgl(\cos(\sigma(t)) - 1)$. Derivando em função do tempo a equação de energia potencial obtém-se o termo $\dot{E}_p = I_s\dot{\sigma}(t)\ddot{\sigma}(t) - mgl\dot{\sigma}(t)\sin(\sigma(t))$, este termo pode ser aplicado na equação de movimento do pêndulo $I_s\ddot{\sigma}(t) - mgl\sin(\sigma(t)) + mu_{sw}(t)l\cos(\sigma(t)) = 0$ estabelecida por Åström e Furuta (2000), encontrando assim $\dot{E}_p = -mu_{sw}(t)l\dot{\sigma}(t)\cos(\sigma(t))$, em que o $u_{sw}(t)$ é definido como a aceleração do pivô. A equação derivada da energia implica que em certos pontos a controlabilidade é perdida, essas posições estão localizadas nas posições $\sigma(t) = \pm 90^\circ$ ou quando velocidade do pêndulo é nula. A equação de Åström e Furuta (2000) também demonstra que o sinal de controle deverá positivo enquanto o termo $\dot{\sigma}(t)\cos(\sigma(t))$ for negativo ou vice-versa.

A derivada da função de Lyapunov $L_y = \frac{(E - E_{ref})^2}{2}$ com a lei de controle $u = k(E - E_{ref})\dot{\sigma}\cos\sigma$ implica que a função de Lyapunov decresce contanto que não esteja nos pontos no qual a controlabilidade é perdida. Uma outra lei de controle é desenvolvida no qual $u_{sw}(t) = \text{sat}_{u_{sw}\text{máx}}(k(E - E_{ref})\text{sgn}(\dot{\sigma}(t)\cos(\sigma(t))))$ resulta em uma estratégia que atua como controlador linear. A função de energia de referência E_{ref} fará o controlador atuar até atingir 0° , onde a energia potencial será nula, isto é válido enquanto o cosseno da posição e aceleração do pêndulo forem diferentes de 0.

A equação proposta por Alves, Neves e Angélico (2019) possui o fundamento semelhante com o proposto por Åström e Furuta (2000). No entanto ao invés de utilizar uma função de Lyapunov na lei de controle, é utilizado um ângulo que pondera o torque pela lei de controle, que é dada por (24).

$$u_2(t) = -K_{sw}\sigma^n(t)\text{sgn}(\dot{\sigma}(t)\cos(\sigma(t))). \quad (24)$$

O termo $\sigma^n(t)$ tem um comportamento semelhante de uma função de energia desejada, em que para valores de ângulos próximos à 0° sua atuação é baixa, enquanto para ângulos maiores há um aumento. O sinal de controle será positivo enquanto o termo $\dot{\sigma}(t)\cos(\sigma(t))$ for negativo ou vice-versa. Desta forma, em conjunto com o termo $\sigma^n(t)$ a velocidade do pêndulo é aumentada gradualmente até alcançar a região próxima ao ponto desejado.

Para que o pêndulo balance de forma semelhante tanto na região maior que π e menor que π , é necessário criar outra variável para o *swing up*, como se trata de uma posição que opera em 0 à 360° , resulta em diferentes atuações do controlador nas regiões especificadas, fazendo com que o pêndulo tenha uma tendência maior de estabilizar em apenas uma região. A variável $\delta_s(t)$ é obtida em (25).

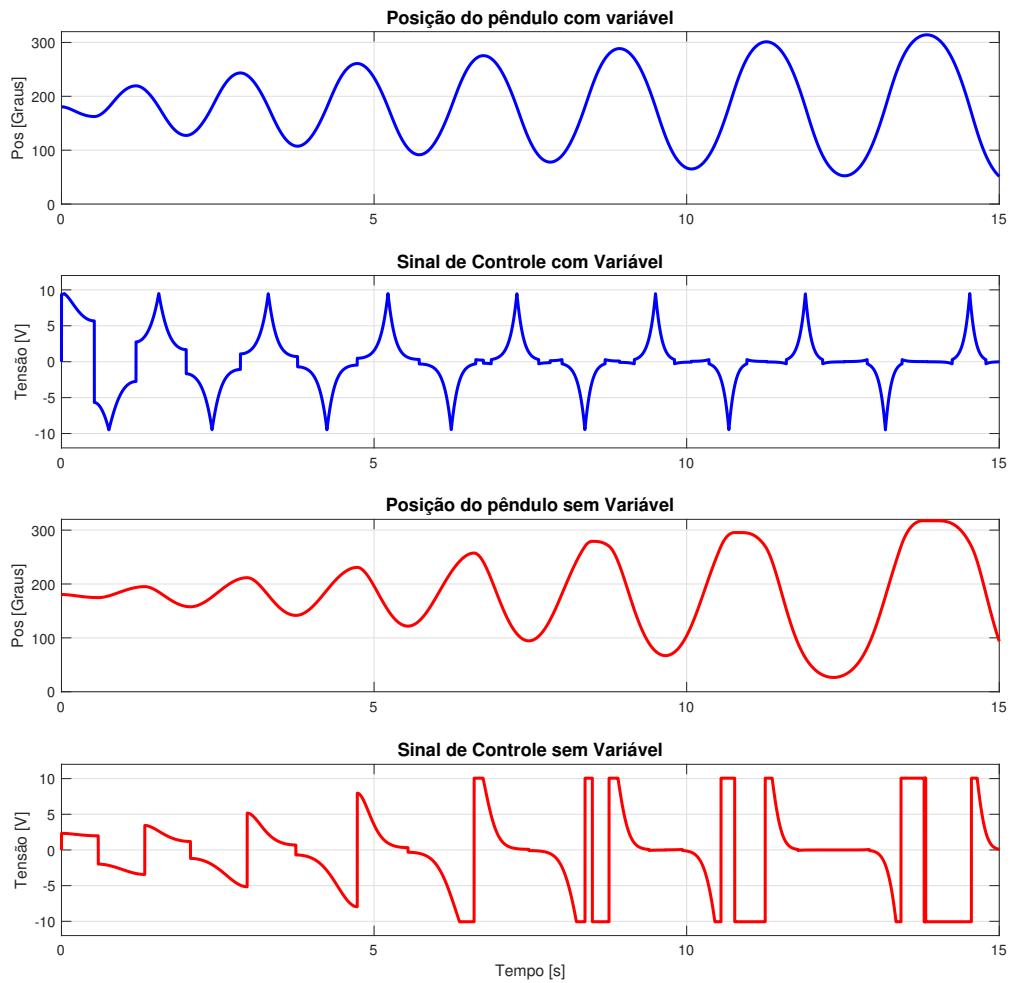
$$\delta_s(t) = \begin{cases} |\sigma(t) - 2\pi| & \text{se } \sigma(t) > \pi \\ \sigma(t) & \text{se } \sigma(t) < \pi. \\ \pi & \text{se não} \end{cases} \quad (25)$$

É importante ressaltar que essa variável tem um impacto significativo em ganhos que não estão saturados, caso o ganho do controlador *swing up* seja muito elevado, os valores máximos do controlador serão empregados em cada região, desta forma, a resposta será independente desta nova variável (25) a ser empregada ou não no sistema.

Desta maneira mesmo na referência de 180° é garantido que o pêndulo irá atingir seu ponto de equilíbrio instável na região invertida com energia potencial e cinética aproximadamente nulas. Com o $\delta_s(t)$ introduzido, a equação passou a ter uma área de atuação menor e consequentemente foi necessário aumentar o ganho para apresentar um tempo de resposta semelhante. Os ganhos foram definidos com o valor de $K_{sw} = 0,031$ com a variável $\delta(t)$ e $K_{sw} = 0,0075$ sem a variável $\delta(t)$ e soltos a partir da C.I de $180,48^\circ$. Analisando a resposta do controlador na Figura (11), é possível perceber a diferença significativa entre os dois métodos. Contudo, utilizando (25) o controlador chega à valores

para graus de módulo máximo de 180° diferente da utilizada sem a variável, que pode utilizar a posição real do pêndulo que possui a capacidade de atingir 360° , e por conta disso foi necessário aumentar o ganho do modelo considerando a variável. A oscilação do pêndulo se tornou simétrica com uma forma de onda mais suave atuando igualmente em ambos sentidos, possibilitando desta maneira que o pêndulo estabilize tanto em 0° como 360° .

Figura 11 – Gráficos de resultados com variável *swing up* δ_s



Fonte: Autor (2022)

4.3 ESTRATÉGIA DE CONTROLE *SWITCH*

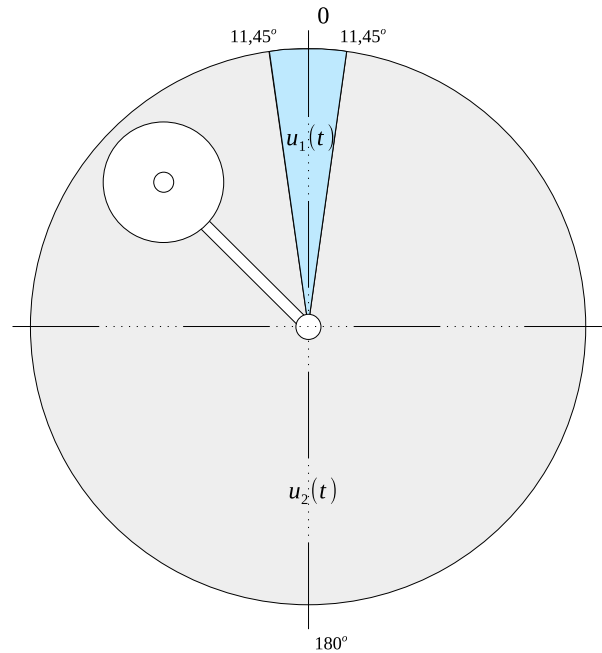
É necessário que uma estratégia de controle seja definida para alternar os controladores em seus respectivos pontos de operação. A estratégia de controle utiliza como

referência o ângulo delta, (23), projetado para o controle de estabilização, então no momento que o delta está localizado entre os ângulos menores que $11,45^\circ$ e maiores que $-11,45^\circ$ é acionado o controlador de estabilização $u_1(t)$, e para ângulos fora dessa faixa o controlador *swing up* $u_2(t)$ é ativo. A estratégia de controle projetada é apresentada por (26)

$$u(t) = \begin{cases} u_1(t) & \text{se } 11,45^\circ \geq \delta(t) \geq -11,45^\circ \\ u_2(t) & \text{se não} \end{cases}. \quad (26)$$

Desta maneira, são obtidas todas ferramentas que possibilitem estabilizar o pêndulo na região de equilíbrio instável a partir de uma condição inicial próxima do ponto estável de equilíbrio, que é demonstrado pela Figura 12.

Figura 12 – Área de atuação dos controladores



Fonte: Autor (2022)

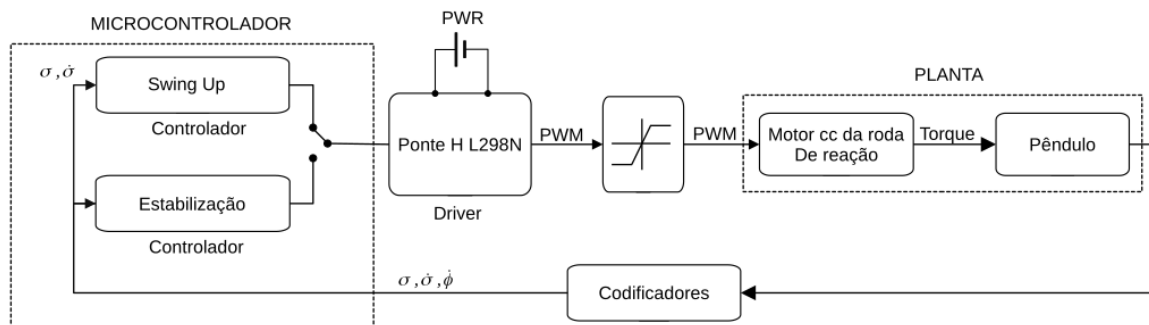
4.4 MALHA DE OPERAÇÃO DO SISTEMA

- Primeiramente, os *encoders* rotativos são responsáveis por obter rotações em seus respectivos eixos em forma de onda de pulsos, em 2 canais, que indicam o sentido de giro. Como seus sinais de saída são digitais não há a necessidade de realizar conversões para entrada do esp32; esses pulsos são calculados por uma *library* da IDE do arduino e estão relacionados de acordo com o número de revolução dos

encoders; são então convertidos utilizando um algoritmo em forma de velocidade e posição angulares;

- É calculado através do algoritmo o produto da posição e velocidades angulares, pelo $-K$, que foi obtido a partir do método por alocação de polos. É somado o valor de cada produto, obtendo a lei de controle de estabilização, $u_1(t) = -Kx$;
- A resolução em bits que o PWM opera, foi definida em 10 bits; significa que o mesmo opera em uma região de valor 0 à 1023. Contudo, existe uma região de zona morta presente nos níveis de tensão do motor, sendo que valores de 0 à 590 são insuficientes para produzir movimento no eixo, sendo assim, há a necessidade de que os valores sejam mapeados, para que o motor opere na região fora dessa zona. Assim, foi estabelecida uma região de operação de 590 à 1023;
- Os sinais de controle, contendo as informações de sentido e valores de tensão necessários para aquele instante de tempo, são enviados para uma ponte H.
- Por fim, a ponte H, ligada à uma fonte de potência externa, alimenta o motor cc com PWM, através dos sinais recebidos pelo microcontrolador, para que o pêndulo convirja para a posição estabelecida pelo controlador.

Figura 13 – Diagrama de blocos



Fonte: Autor (2022)

Um ponto importante a ser comentado, é que a ponte H trabalha em uma operação de 7 V à 35 V e produz uma queda de tensão nos seus transistores, assim como o cabo de alimentação do motor que também possui uma pequena queda de tensão. Desta forma, a tensão de alimentação do motor será menor que da fonte externa, e isso implica em perdas de velocidade e torque, que são essenciais para o controle do pêndulo.

No caso do experimento, os níveis de tensão da alimentação do motor ligado diretamente na fonte de tensão era de 12 V, e por meio da ponte H, junto com a queda de

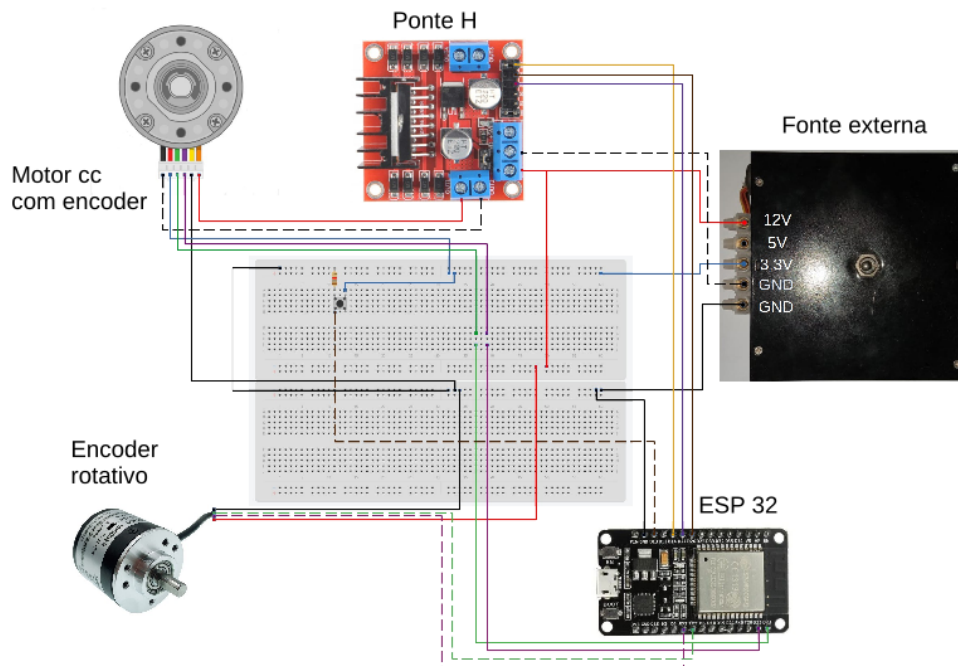
tensão no cabo, resultou em uma tensão de 10,06 V, ou seja uma queda de 1,74 V em seus transistores e 0,20 V nos cabos. Em consequência desta inconveniência, houve queda de 220 rpm, fazendo o motor operar com velocidade $\dot{\phi}(t)$ máxima de aproximadamente 880 rpm durante os ensaios.

É possível contornar esse problema utilizando uma fonte superior a 12 V, em conjunto com um regulador de tensão, permitindo desta maneira ajustar o nível de tensão para suprir a diferença na alimentação da ponte, mas com um aumento de custo do projeto. A queda de tensão não comprometeu o controle.

Uma fonte de alimentação para computadores de 300 W foi utilizada para fornecer energia para o sistema, devido seu baixo custo e sua capacidade de fornecer corrente suficiente para acelerações do motor. A fonte foi adaptadas com bornes para suprir as tensões e GND necessárias no sistema.

Na Figura 14 pode ser observado o sistema elétrico dos componentes do modelo experimental.

Figura 14 – Sistema elétrico do modelo PIRR



Fonte: Autor (2022)

É importante comentar, que para o pêndulo conseguir estabilizar com uma maior facilidade sem exigir muito do motor, é necessário que o sistema do pêndulo esteja com as massas perfeitamente equilibradas. Esse não é o caso do experimental, pois até mesmo um ínfimo erro de medida em sua construção, ou mesmo o cabo que liga os componentes do

motor podem alterar sua posição de equilíbrio real, mesmo que seja uma diferença muito pequena no ângulo; devido à isso foi implementado um botão para redefinir a posição de equilíbrio 0° , ao invés de simplesmente inicializar na posição 180° para baixo.

5 RESULTADOS COMPARATIVOS

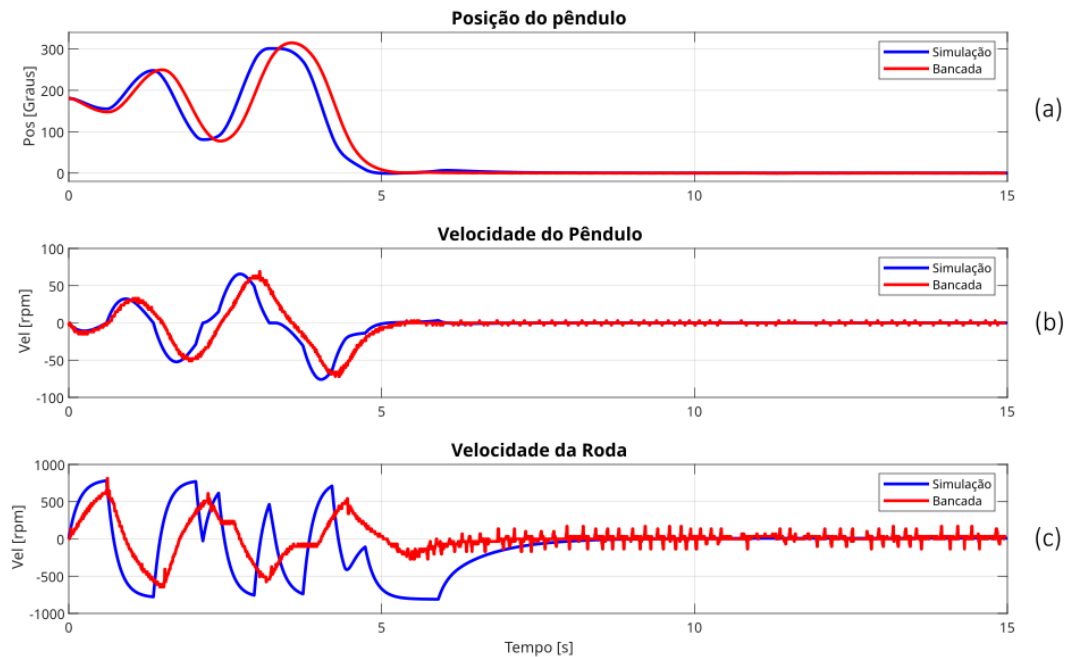
Por fim, é abordado a comparação entre os gráficos dos dados obtidos experimentalmente e teoricamente. É feita a análise dos gráficos e suas principais diferenças, de forma explicativa sobre a incompatibilidade dos dados.

Primeiramente, para que fosse realizada a comparação entre os modelos experimental e teórico, foram consideradas condições iniciais de posição similares e o ganho projetado para ambos. O mesmo é dito para os valores das variáveis consideradas no experimental, no qual os ganhos dos controladores foram adotados de acordo com os valores obtidos no projeto teórico. O atrito das escovas presente no motor foi considerado e aproximado de maneira empírica para fornecer uma resposta mais similar com o do modelo experimental, devido à seu desprovimento na tabela de dados do produto. Tal variável pode gerar uma diferença significativa em momentos que ocorre a atuação do motor. Para o realizar o experimento, foi adotado uma condição inicial da posição do pêndulo de aproximadamente $180,48^\circ$ graus, e o atrito do motor, A_m , encontrado para o modelo teórico, foi de 0,0014.

Para o cálculo de velocidade, tanto do pêndulo como o da roda, foi adotado um período de 0,01 s, devido à resolução baixa do motor; valores inferiores ao adotado, resultaram em erros que comprometiam a efetividade do controlador. Para o ganho do *swing up*, foi adotado um ganho a partir de teste com o modelo experimental de $K_{sw} = -50$. Na Figura 15 são demonstrados os dados das variáveis de estado.

Como visto, a Figura 15(a) apresenta um resultado semelhante de ambos modelos tanto para realizar o controle de *swing up* como para estabilizar o pêndulo na posição invertida. Essa defasagem, que ocorre durante o *swing up*, está sendo causada por erros de aproximações feitas com os dados e atrito do motor que foram insuficientes para produzir um modelo com precisão, ou mesmo os sensores utilizados para medição de velocidade.

Figura 15 – Resultado dos gráficos de estados

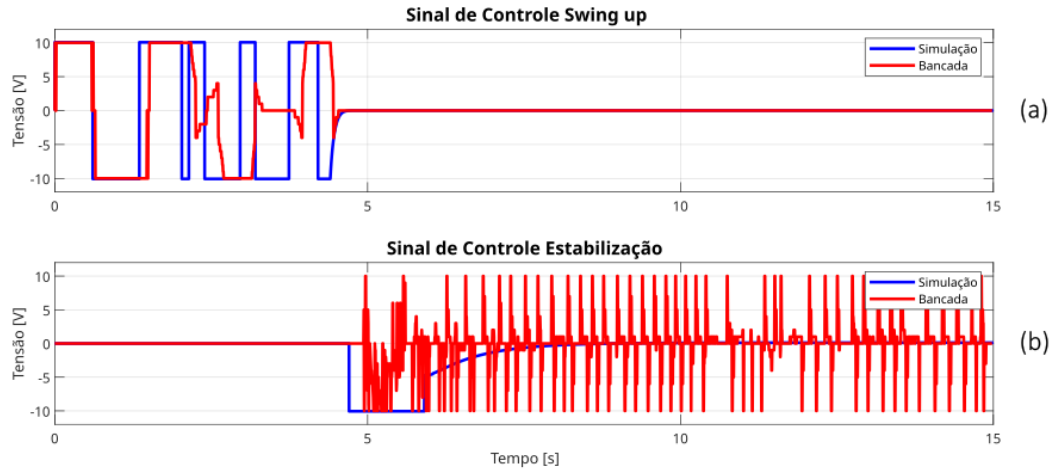


Fonte: Autor (2022)

Assim como a posição do pêndulo, a Figura 15(b), produziu uma resposta semelhante com a desenvolvida a partir do modelo teórico, devido ao encoder do pêndulo que possui um número maior de pulsos, em relação com o do motor. Contudo, mesmo com um sensor de 1000 pulsos por revolução, é possível observar durante a atuação do controle de estabilização, que existe uma variação mínima considerável na velocidade de até 3 rpm, impossibilitando de distinguir valores abaixo deste limite que é causado por limitações do sensor.

O sensor presente no motor, responsável por obter o estado de velocidade da roda, que pode ser visto de acordo com a Figura 15(c), possui um número menor de pulsos em relação com o do pêndulo, por isso é esperado uma resposta com uma variação significativa. Este efeito, semelhante a sinal ruidoso, assim como na velocidade do pêndulo, é visível durante a atuação do controle de estabilização, porém, os valores são muito mais significativos, atingindo no mínimo 33,9 rpm, e obtendo uma resposta menos suave para o sistema. Devido ao fato de uma boa parte das constantes do motor serem estimadas, por serem insuficientes em datasheet, ou o atrito ser obtido empiricamente, resulta em um gráfico com uma tendência maior de erros. Os gráficos da Figura 16 representam os controladores *swing up* e de estabilização, separadamente.

Figura 16 – Resultado do gráfico dos controladores



Fonte: Autor (2022)

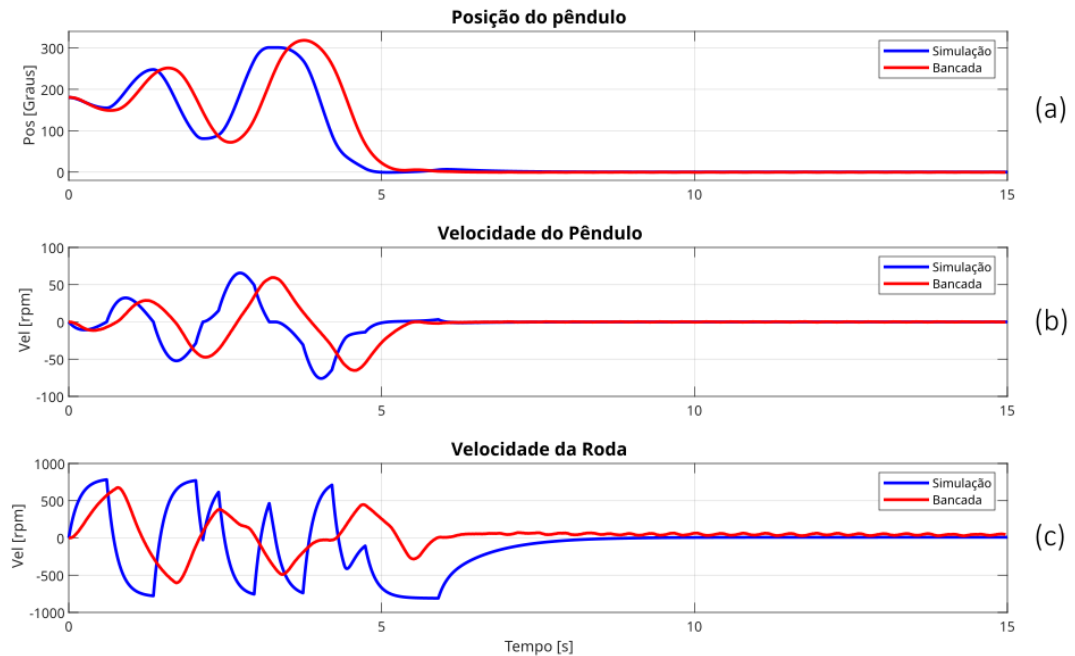
Analisando a Figura 16(a), pode ser observado as saturações decorrentes do alto valor de ganho adotado, que consequentemente diminui muito o tempo que leva para atingir a posição de equilíbrio invertida. Por fim, a Figura 16(b) do controle de estabilização possui uma atuação semelhante com a do teórico, exceto pelas oscilações que são causados pelos erros de medição de velocidade, sendo obtida a partir da derivada da posição, que junto com a baixa resolução do sensor do motor aumenta significativamente tal efeito. Esses erros podem ser amenizados por um simples filtro de primeira ordem, ou mesmo um observador de velocidade para produzir uma melhora significativa na resposta do controlador.

5.1 RESULTADOS COMPARATIVOS COM FILTRO

Como foi observado nos gráficos anteriormente, a ausência de um filtro nos cálculos para obter a velocidade resultou em ruídos que impactou de maneira significativa os controladores. Devido a isso, foram realizados ensaios considerando o filtro na programação do modelo experimental, a fim de obter repostas mais suavizadas e menos agressivas dos controladores. As constantes utilizadas para o experimento possuem os mesmos valores do anterior, com exceção de uma nova constante, que é o tempo de filtragem, que foi declarada com um valor de $T = 0,1$ s. Contudo é importante comentar que algumas variáveis podem não ser idênticas com a do experimento anterior, devido a condições iniciais, como no caso de aplicar um leve impulso para o pêndulo iniciar o *swing up*.

Na Figura 17 são representadas as variáveis de estado com filtro aplicado ao modelo experimental.

Figura 17 – Resultado dos gráficos de estados com filtro



Fonte: Autor (2022)

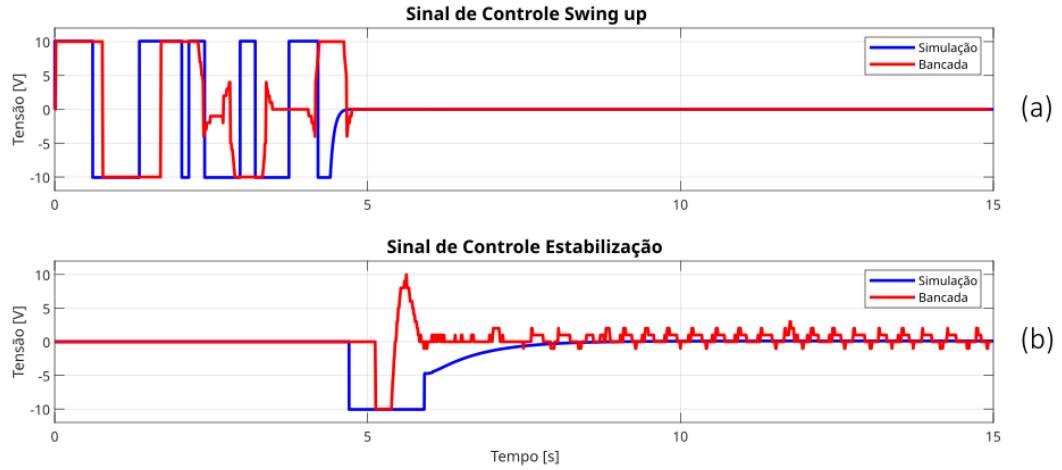
De início, é possível observar, de acordo com a Figura 17 que os dados sofreram uma leve defasagem em comparação com a Figura 15. As Figuras 17(b) e 17(c) apresentaram uma leve queda em suas amplitudes se comparadas com os dados sem o filtro implementado, porém os ruídos foram reduzidos de maneira significativas, fazendo com que o sinais tornassem um aspecto mais suave e menos agressivo. Na Figura 18 são apresentados os sinais de controle com filtro implementado no modelo experimental.

De acordo com a Figura 18(a), é possível observar a causa da defasagem que ocorreram nas variáveis de estado durante a atuação do *swing up*, com a implementação do filtro; a atuação dos controladores ficaram menores, com amplitudes significativamente inferiores que resultou nessa defasagem das variáveis.

Contudo como visto na Figura 18(b), houve uma redução drástica da atuação do controlador, isso pode aumentar o tempo que leva para o controlador estabilizar o pêndulo em posições bem próximas de 0° , mesmo que não tendo sido observada nenhuma piora significativa. Ao contrário dos ruídos que causavam saturação do modelo sem filtro, o sinal de estabilização passou a apresentar uma resposta menos agressiva ao motor, aumentando

seu tempo de vida útil, e assim prejudicando menos o sistema ou mesmo reduzindo seu custo ao longo do tempo.

Figura 18 – Gráfico dos controladores com filtro



Fonte: Autor (2022)

5.2 ÍNDICE DE DESEMPENHO DOS CONTROLADORES

O índice de desempenho é uma ferramenta de medida quantitativa de desempenho de um sistema que permite analisar o desempenho do sistema ou projetar um sistema de controle, possibilitando desta maneira que seja utilizado para melhorar o projeto desenvolvido (DORF; BISHOP, 2008). Existem diversos índices que são escolhidos de acordo com a especificação do sistema. O índice escolhido para realizar esta análise, foi o IAE, integral da magnitude absoluta do erro, que é um índice útil para simulações computacionais, descrito por (27).

$$IAE = \int_0^T |e(t)| dt. \quad (27)$$

Neste caso, será aplicado o índice para verificar o desempenho do controlador de estabilização com e sem filtro, em específico, o desempenho energético. Desta forma, foi calculado os índices para as duas situações, sendo apresentados pela Tabela 3.

Tabela 3 – Índice de desempenho

Controle	Valor
Estabilização sem filtro	2009
Estabilização com filtro	976
<i>Swing up</i> sem filtro	3278
<i>Swing up</i> com filtro	3463

Fonte: Autor (2022)

Como visto na Tabela 3, o controlador de estabilização com filtro apresentou um índice com valor significativamente inferior, em relação com o mesmo controle sem filtro, que representa um diferença de até 51,5%. Já no caso do controlador *swing up* com filtro, houve uma agravamento em seu índice, apesar de ser relativamente baixo, com uma diferença de até 5,4%. Portanto, é possível afirmar que o filtro aperfeiçoou de maneira significativa o custo energético do sistema.

5.3 ELIPSE E3

O *software* Elipse E3, é um sistema de supervisão e aquisição de dados. O *software* tem o principal propósito de fornecer monitoramento e controle de processos utilizados no modelo experimental em tempo real com ferramentas que permitem editar a interface. Desta maneira, é possível montar desde as mais simples até as mais complexas interfaces. A conexão com o dispositivo e o *software* foram definidas em um tempo de 150 ms, esse tempo é relativamente alto para a dinâmica do pêndulo, contudo, a conexão foi realizada via wi-fi para ter um impacto menor no tempo de resposta do microcontrolador, e o tempo de resposta que leva para a troca de informações entre os dois dispositivos varia de 100-150 ms.

A interface desenvolvida para o modelo experimental deste trabalho, tem o intuito de ser simples com elementos que ajudam a entender o funcionamento e atuação dos controladores e o impacto dos ganhos na controlabilidade do modelo pêndulo. As interfaces desenvolvidas no *software* SCADA podem ser vistas no Apêndice A.4. Cada botão possui a finalidade de alterar as telas da interface.

A Figura A.4.1, demonstrada no apêndice, apresenta a tela inicial da interface, na qual a imagem do pêndulo está ligada de acordo com a variável de posição, desta forma a imagem replica os movimentos semelhante com a do modelo experimental. A tela de

controle, representada pela Figura A.4.2, possibilita alterar os ganhos de cada controlador, e também visualizar os valores das variáveis de estado e controladores.

Na Figura A.4.3 apresenta os gráficos das variáveis de estado e controladores presentes no sistema, e também conta com duas luzes indicadoras, para sinalizar qual controlador está ativo naquele instante de tempo.

Por fim, a Figura A.4.4 tem apenas um único gráfico da variável de estado posição do pêndulo, para fornecer uma visibilidade melhor para sua análise. Foi produzido um vídeo para mostrar o controle em tempo real do *software* Elipse E3, e também o funcionamento do modelo experimental do pêndulo finalizado.

<https://www.youtube.com/watch?v=s5Tv4yxJm1g>

5.4 LISTA DE VALORES DOS MATERIAIS UTILIZADOS

Os materiais utilizados para construção do modelo experimental são materiais de baixo custo, e alguns são completamente opcionais, de forma que seja possível reduzir ainda mais o custo do experimental. Os valores dos materiais listados abaixo na Tabela 4 podem estar sujeitos à alterações em seu valores.

Tabela 4 – Tabela de valores

Material	Reais	Dólares
Disco de freio de bicicleta Absolute 203mm	40,99	7,13
Motor CHP-36GP-555-ABHL	191,35	33,31
Codificador rotativo 1000ppr	90,44	15,74
Ponte H L298N	25,00	4,35
Microcontrolador ESP32	50,00	8,71
Protoboard Hikari 3220 furos	235,00	40,92
Barra chata de alumínio	52,00	9,24
Mancal pedestal	61,00	11,02
Fonte de alimentação para PC	99,00	19,91
Total	844,48	150,33

Fonte: Autor (2022)

A Protoboard, por exemplo pode ser feita em uma placa de circuito integrado ou mesmo um outro modelo mais barato a fim de reduzir os custos. As arruelas também

são opcionais; existe uma grande variedade de materiais que podem ser utilizados como contrapeso. O disco de freio de bicicleta é um substituto de uma grande variedade de rodas para ser utilizada no experimental, contudo, o diâmetro, peso ou mesmo sua forma pode gerar mudanças no momento de inércia e solicitar mais torque do motor; o mesmo pode ser dito para a barra de alumínio que será o suporte do sistema.

O motor escolhido foi levado em consideração o torque e a velocidade. São dois fatores importantes para gerar uma aceleração angular satisfatória para controlar o pêndulo, contudo, esses fatores proporcionam um aumento no peso do motor, causando dificuldades ou mesmo impossibilitando de controlar o sistema; juntando esse fato com a necessidade do motor possuir um sensor de velocidade no motor, há um encarecimento do sistema.

Se desconsiderar a protoboard, por ser um objeto totalmente opcional, o valor total da configuração do modelo experimental, é de 110,00 dólares, ou aproximadamente 550,00 reais.

6 OBSERVAÇÕES FINAIS

Este capítulo apresenta as observações finais sobre essa dissertação, e propostas para continuidade de pesquisa sobre o assunto deste trabalho.

6.1 CONCLUSÃO

Foi desenvolvido, ao longo deste trabalho, o modelamento dinâmico e simulação com controladores do pêndulo invertido por roda de reação, utilizando os dados dos parâmetros adquiridos experimentalmente.

O sistema para ensaios experimentais foi construído com o intuito de ter uma base em ferramentas de baixo custo, permitindo desta forma oferecer fácil acesso e a capacidade de melhorias futuras. Seu valor para todos materiais necessários utilizados foi de aproximadamente R\$ 550,00.

Um método de controle por realimentação de estados foi desenvolvido para estabilizar o pêndulo. Os ganhos foram projetados a partir dos polos obtidos em simulação de um modelo linearizado. Com o intuito de elevar o pêndulo, foi utilizado um controlador inspirado nos métodos do controle de energia. Como houve problemas de flutuação causado pela derivada, a fim de obter os sinais de velocidade, foi desenvolvido um filtro que apresentou uma melhora significativa no sistema. O *software* Elipse E3 foi utilizado como plataforma de supervisão e controle em tempo real para análise do modelo experimental.

Foi constatado a eficiência e sucesso dos controladores no modelo experimental a partir dos dados obtidos experimentalmente, e pela comparação com os dados do modelo teórico. O resultado deste trabalho pode ser observado no link:

<https://www.youtube.com/watch?v=s5Tv4yxJm1g>

6.2 PROPOSTAS DE CONTINUIDADE DO TRABALHO

Serão apresentadas as sugestões para a continuidade do trabalho:

- Implementação de Controladores Discretos;
- Comparação com modelos de controladores não lineares e robustos;
- Implementação de uma nova Roda de Reação, com mais graus de liberdade e/ou outra roda.
- Implementação de um controle chaveado no sistema.
- Considerar a saturação no projeto.

REFERÊNCIAS

- ALVES, R. M. R.; NEVES, G. P. d.; ANGÉLICO, B. A. Modelagem construção e controle de um pêndulo invertido com roda de reação. In: **Congresso Brasileiro de Automática-CBA**. [S.l.: s.n.], 2019. v. 1, n. 1.
- ANDRIEVSKY, B. Global stabilization of the unstable reaction-wheel pendulum. **Automation and Remote Control**, Moscow, v. 72, n. 9, p. 1981--1993, 2011.
- ÅSTRÖM, K. J.; FURUTA, K. Swinging up a pendulum by energy control. **Automatica**, Oxford, v. 36, n. 2, p. 287--295, 2000.
- BAPIRAJU, B.; SRINIVAS, K.; KUMAR, P. P.; BEHERA, L. On balancing control strategies for a reaction wheel pendulum. In: IEEE. **Proceedings of the IEEE INDICON 2004. First India Annual Conference, 2004**. [S.l.], 2004. p. 199--204.
- BLOCK, D. J.; ÅSTRÖM, K. J.; SPONG, M. W. The reaction wheel pendulum. **Synthesis Lectures on Control and mechatronics**, San Rafael, v. 1, n. 1, p. 1--105, 2007.
- BUTIKOV, E. I. Spring pendulum with dry and viscous damping. **Communications in Nonlinear Science and Numerical Simulation**, Amsterdam, v. 20, n. 1, p. 298--315, 2015.
- DORF, R. C.; BISHOP, R. H. **Modern control systems**. Upper Saddle River: Pearson Prentice Hall, 2008.
- DRIESSEN, J. J.; GKIKAIS, A. E.; FEATHERSTONE, R.; SINGH, B. R. P. Experimental demonstration of high-performance robotic balancing. In: IEEE. **2019 International Conference on Robotics and Automation (ICRA)**. [S.l.], 2019. p. 9459--9465.
- FANTONI, I.; LOZANO, R.; SINHA, S. Non-linear control for underactuated mechanical systems. **Applied Mechanics Reviews**, New York, v. 55, n. 4, p. B67--B68, 2002.
- GOLZARI, A.; PISHKENARI, H. N.; SALARIEH, H.; ABDOLLAHI, T. Quaternion based linear time-varying model predictive attitude control for satellites with two reaction wheels. **Aerospace Science and Technology**, Issy les Moulineaux Cedex, v. 98, p. 105677, 2020.
- HERNANDEZ, V. M.; SIRA-RAMIREZ, H. Generalized pi control for swinging up and balancing the inertia wheel pendulum. In: IEEE. **Proceedings of the 2003 American Control Conference, 2003**. [S.l.], 2003. v. 4, p. 2809--2814.
- JAE-OH, L.; IN-WOO, H.; JANG-MYUNG, L. Fuzzy sliding mode control of unicycle robot. In: IEEE. **2011 8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)**. [S.l.], 2011. p. 521--524.

KIM, D.-K. Micro-vibration model and parameter estimation method of a reaction wheel assembly. **Journal of Sound and Vibration**, London, v. 333, n. 18, p. 4214--4231, 2014.

KIM, H.-W.; AN, J.-W.; YOO, H. dong; LEE, J.-M. Balancing control of bicycle robot using pid control. In: IEEE. **2013 13th International Conference on Control, Automation and Systems (ICCAS 2013)**. [S.l.], 2013. p. 145--147.

LEE, S.-H.; GOSWAMI, A. Reaction mass pendulum (rmp): An explicit model for centroidal angular momentum of humanoid robots. In: IEEE. **Proceedings 2007 IEEE International Conference on Robotics and Automation**. [S.l.], 2007. p. 4667--4672.

LUBLOVARY, G. G.; INSPERGER, T. Transient stabilization of an inverted pendulum with digital control. **IFAC-PapersOnLine**, [s. l.], v. 51, n. 22, p. 197--202, 2018.

MASTERSON, R. A.; MILLER, D. W.; GROGAN, R. L. Development and validation of reaction wheel disturbance models: empirical model. **Journal of Sound and Vibration**, London, v. 249, n. 3, p. 575--598, 2002.

MONTOYA, O. D.; GIL-GONZÁLEZ, W. Nonlinear analysis and control of a reaction wheel pendulum: Lyapunov-based approach. **Engineering Science and Technology, an International Journal**, Amsterdam, v. 23, n. 1, p. 21--29, 2020.

MUEHLEBACH, M.; DÍANDREA, R. Nonlinear analysis and control of a reaction-wheel-based 3-d inverted pendulum. **IEEE Transactions on Control Systems Technology**, Piscataway, v. 25, n. 1, p. 235--246, 2016.

NISE, N. S. **Control systems engineering**. Hoboken: John Wiley & Sons, 2020.

OLIVARES, M.; ALBERTOS, P. Linear control of the flywheel inverted pendulum. **ISA transactions**, New York, v. 53, n. 5, p. 1396--1403, 2014.

RAN, D.; CHEN, X.; RUITER, A. de; XIAO, B. Adaptive extended-state observer-based fault tolerant attitude control for spacecraft with reaction wheels. **Acta Astronautica**, Elsevier, v. 145, p. 501--514, 2018.

RIZAL, Y.; MANTALA, R.; RACHMAN, S.; NURMAHALUDIN, N. Balance control of reaction wheel pendulum based on second-order sliding mode control. In: IEEE. **2018 International Conference on Applied Science and Technology (iCAST)**. [S.l.], 2018. p. 51--56.

RYBOVIC, A.; PRIECINSKY, M.; PASKALA, M. Control of the inverted pendulum using state feedback control. In: IEEE. **2012 ELEKTRO**. [S.l.], 2012. p. 145--148.

SOWMAN, J.; LAILA, D. S.; LONGO, S. Real-time approximate explicit nonlinear model predictive control for the swing-up of a reaction wheel pendulum. In: IEEE. **2015 54th IEEE Conference on Decision and Control (CDC)**. [S.l.], 2015. p. 4308--4313.

SPONG, M. W.; CORKE, P.; LOZANO, R. Nonlinear control of the reaction wheel pendulum. **Automatica**, Oxford, v. 37, n. 11, p. 1845--1851, 2001.

SRINIVAS, K.; BEHERA, L. Swing-up control strategies for a reaction wheel pendulum. **International Journal of Systems Science**, Abingdon, v. 39, n. 12, p. 1165--1177, 2008.

TRENTIN, J. F. S.; SILVA, S. D.; RIBEIRO, J. M. D. S.; SCHAUB, H. Inverted pendulum nonlinear controllers using two reaction wheels: design and implementation. **IEEE Access**, Piscataway, v. 8, p. 74922--74932, 2020.

XIN, X.; KANEDA, M. The swing up control for the acrobot based on energy control approach. In: IEEE. **Proceedings of the 41st IEEE Conference on Decision and Control, 2002**. [S.l.], 2002. v. 3, p. 3261--3266.

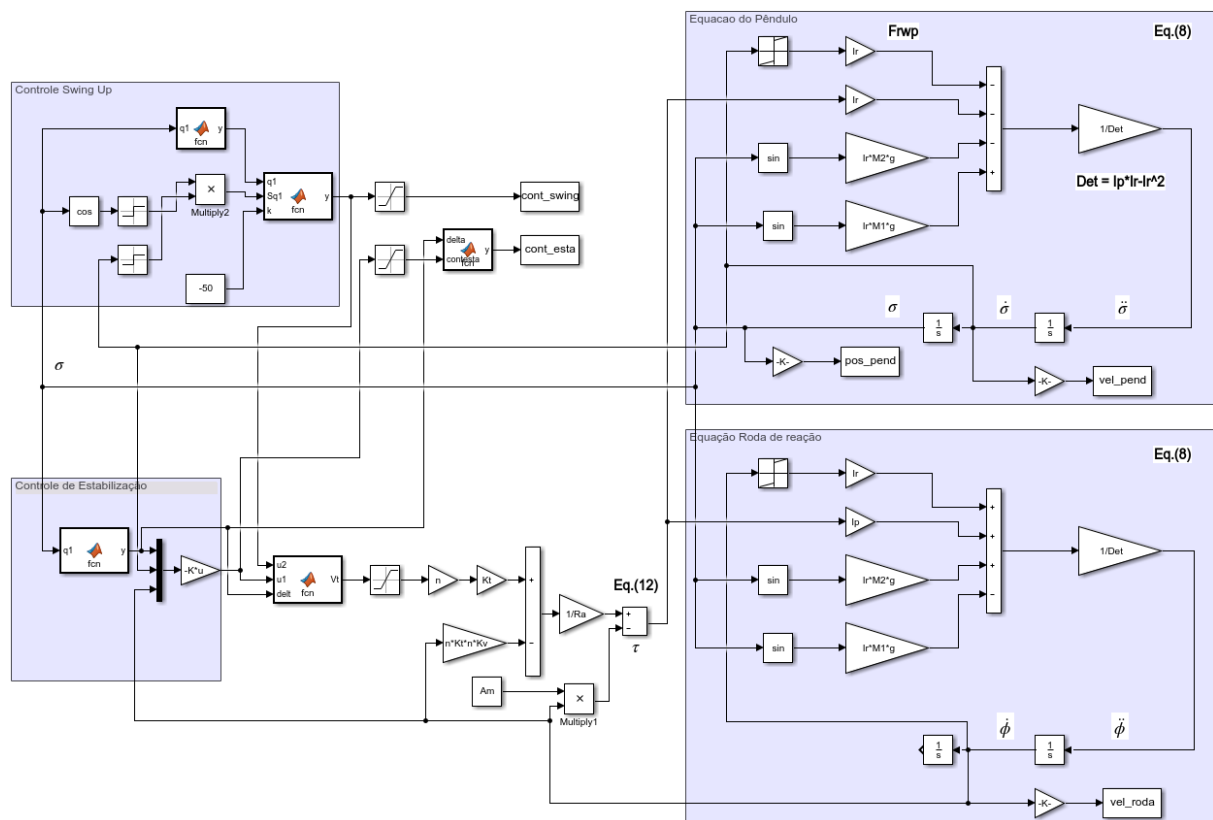
ZONETTI, L.; CAMARGO, A.; SARTORI, J.; SOUSA, D. D.; NUNES, L. A demonstration of dry and viscous damping of an oscillating pendulum. **European journal of physics**, Bristol, v. 20, n. 2, p. 85, 1999.

A APÊNDICE

A.1 PROGRAMA PARA SIMULAÇÃO SIMULINK

Este anexo apresenta o modelo simulink do PIRR controlado que foi utilizado para simular e obter os dados.

Figura A.1.1 – Modelo simulink PIRR



Fonte: Autor (2022)

A.2 PROGRAMAÇÃO EXPERIMENTAL

```
1 #ifdef ESP8266
2 #include <ESP8266WiFi.h>
3 #else //ESP32
4 #include <WiFi.h>
5 #endif
6 #include <ModbusIP_ESP8266.h>
7 #include <ESP32Encoder.h>
8
9 #define inPin 13
10 #define enable1Pin 14
11 #define motor1Pin1 26
12 #define motor1Pin2 27
13
14 ESP32Encoder encoder;
15 ESP32Encoder encoder2;
16
17 const unsigned long timePeriod = 10; // TIMER PERIODO DA RODA
18 const unsigned long timePeriod1 = 10; //TIMER PERIOD DO PENDULO
19 const unsigned long Displaytimer = 10;
20 unsigned long startTime; // TIMER RODA
21 unsigned long startTime1; // TIMER PEND
22 unsigned long startTimeDisp; // TIMER DISPLAY
23 unsigned long now; // TIMER RODA
24 unsigned long now1; // TIMER PEND
25 unsigned long nowDisp; // TIMER DISPLAY
26
27
28
29 long startPositionPend;
30 long startPosition;
31 long newPosition;
32
33 float Vf = 0;
34 float Vfn = 0;
```

```
35 float Vfp = 0;
36 float Vfpm = 0;
37 float RPM=0;
38 float RPMPend=0;
39 float newPositionPend;
40 float PositionPendRad;
41 float speedmotor = 0;
42 float speedpend = 0;
43 float ContSwing = 0;
44
45
46 // Setting PWM properties
47 const int freq = 30000;
48 const int pwmChannel = 0;
49 const int resolution = 10;
50 //For example, for 20kHz PWM frequency:
51 /* 80 MHz / 20 kHz = 4000
52 /* log 2 (4000) = 11.965...
53 /* integer(11.96) = 11 bits
54
55 int gain = 0;
56 int gainEst1 = 0;
57 int gainEst2 = 0;
58 int gainEst3 = 0;
59 int ControleEstaflag = 0;
60 int ControleSwingflag = 0;
61 int graus = 0;
62 int entryEst1 = 0;
63 int entryEst2 = 0;
64 int entryEst3 = 0;
65 int entry1 = 0;
66 int ContAtual=0;
67 int Cont1=0;
68 int ContSwing1 = 0;
69 int ControleSwing = 0;
70 int VoltageEsta=0;
```

```
71 int ConstVolt=0;
72 int VoltageSwing=0;
73 int ConstVoltSwing=0;
74 int ControleAbs = 0;
75 int val = 0;      // variable for reading the pin status
76
77 // Modbus Registers Offsets
78
79 //ModbusIP object
80 ModbusIP mb;
81 void setup() {
82     Serial.begin(115200);
83
84     WiFi.begin("TP-Link", "*****");
85
86     while (WiFi.status() != WL_CONNECTED) {
87         delay(500);
88         Serial.print(".");
89     }
90
91     Serial.println("");
92     Serial.println("WiFi_connected");
93     Serial.println("IP_address:_");
94     Serial.println(WiFi.localIP());
95
96     mb.server();
97     mb.addHreg(100);
98     mb.addHreg(200);
99     mb.addHreg(201);
100    mb.addHreg(202);
101    mb.addHreg(300);
102    mb.addHreg(301);
103    mb.addHreg(302);
104    mb.addHreg(303);
105    mb.addHreg(304);
106    mb.addHreg(400);
```

```
107  mb.addHreg(401);
108
109  // sets the pins as outputs:
110  pinMode(motor1Pin1, OUTPUT);
111  pinMode(motor1Pin2, OUTPUT);
112  pinMode(enable1Pin, OUTPUT);
113
114  pinMode(inPin, INPUT);
115
116  // configure LED PWM functionalitites
117  ledcSetup(pwmChannel, freq, resolution);
118
119  // attach the channel to the GPIO to be controlled
120  ledcAttachPin(enable1Pin, pwmChannel);
121
122  // Attache pins for use as encoder pins
123  encoder.attachHalfQuad(22, 23);
124  // Attache pins for use as encoder pins
125  encoder2.attachHalfQuad(16, 17);
126
127  // Posicoes iniciais para obter a velocidade do motor.
128  startPositionPend = encoder2.getCount();
129  startPosition = encoder.getCount();
130  startTime = millis();
131  startTime1 = millis();
132  startTimeDisp = millis();
133 }
134
135 long oldPosition = -999;
136 long oldPositionPend = -999;
137
138 void loop() {
139
140  //AJUSTE DE POSICAO
141  //
142  while (val == LOW) {
```

```
143     val = digitalRead(inPin);
144     encoder2.setCount(0);
145     if (val == HIGH){
146         for (int i = 0; i <= 100; i++) {
147             delay(50);
148         }
149     }
150 }
151
152 //INICIAR MODBUS.
153 mb.task();
154 // VARIAVEIS CONTROLADAS EM TEMPO REAL (APENAS INT)
155 entry1 = mb.Hreg(100);
156 entryEst1 = mb.Hreg(200);
157 entryEst2 = mb.Hreg(201);
158 entryEst3 = mb.Hreg(202);
159
160 // VARIAVEIS SUPERVISIONADAS
161 graus = PositionPendRad*180/M_PI;
162 mb.Hreg(300, graus);
163 mb.Hreg(301, RPM);
164 mb.Hreg(302, RPMPend);
165
166 if (delta() < 0.20 && delta() > -0.20){
167     ConstVolt = constrain(ContAtual, -1023, 1023);
168     VoltageEsta = ConstVolt/101.6898;
169     mb.Hreg(303, VoltageEsta);
170 }
171 else{
172     VoltageEsta = 0;
173 }
174
175 ConstVoltSwing = constrain(ContSwing, -1023, 1023);
176 VoltageSwing = ConstVoltSwing/101.6898;
177 mb.Hreg(304, VoltageSwing);
178
```



```

179  mb.Hreg(400, ControleEstaflag);
180  mb.Hreg(401, ControleSwingflag);
181
182
183  //TRANSFORMANDO VARIAVEIS INT CONTROLADAS EM FLOAT
184  gain = entry1;
185  gainEst1 = entryEst1;
186  gainEst2 = entryEst2;
187  gainEst3 = entryEst3;
188
189  //LEITURA DOS ENCODERS PARA VELOCIDADE e POSICAO.
190
191  PositionPendRad = encoder2.getCount()*0.0031415;
192  newPositionPend = encoder2.getCount();
193  newPosition = encoder.getCount();
194
195  // TIMER DO MOTOR PARA CALCULAR VELOCIDADE MEDIA DA RODA
196  now = millis();
197  if ( now - startTime >= timePeriod ) {
198      speedmotor = (newPosition - startPosition) * 0.0355 / 0.01;
199      //Filtro/
200      Vf = 0.90909*Vfn + 0.09090*speedmotor; //UTILIZEI T=0.1 e
          meu h = 0.01
201      Vfn = Vf;
202      startTime = now;
203      startPosition = newPosition;
204  }
205  if (newPosition != oldPosition) {
206      oldPosition = newPosition;
207  }
208
209  // TIMER DO MOTOR PARA CALCULAR VELOCIDADE MEDIA DO PENDULO
210  now1 = millis();
211  if ( now1 - startTime1 >= timePeriod1 ) {
212      speedpend = (newPositionPend - startPositionPend) *
          0.0031415 / 0.01;

```

```

213 //Filtro//
214 Vfp = 0.90909*Vfpn + 0.09090*speedpend; //UTILIZEI T=0.1 e
    h = 0.01
215 Vfpn = Vfp;
216 startTime1 = now1;
217 startPositionPend = newPositionPend;
218 }
219 if (newPositionPend != oldPositionPend) {
220     oldPositionPend = newPositionPend;
221 }
222
223 // CONTROLADOR -Kx PARA ESTABILIZACAO
224
225 ContAtual = gainEst1*delta()+gainEst2*Vfp+gainEst3*Vf;
226 Cont1 = constrain(abs(ContAtual), 0, 1023);
227 ControleAbs = map(Cont1, 0, 1023, 585, 1023);
228
229 // CONTROLADOR SWING UP
230
231 ContSwing = gain*pow(deltaSwing(), 5)*SgnSpeed()*SgnPosPend()
    ;
232 ContSwing1 = constrain(abs(ContSwing), 0, 1023);
233 ControleSwing = map(ContSwing1, 0, 1023, 585, 1023);
234
235 // CONTROLE DE VELOCIDADE DO MOTOR POR PWM
236
237 if (delta() < 0.20 && delta() > -0.20){
238     if (SentidoControle() > 0){
239         digitalWrite(motor1Pin1, HIGH);
240         digitalWrite(motor1Pin2, LOW);
241         ledcWrite(pwmChannel, ControleAbs);
242     }
243     else if (SentidoControle() < 0){
244         digitalWrite(motor1Pin1, LOW);
245         digitalWrite(motor1Pin2, HIGH);
246         ledcWrite(pwmChannel, ControleAbs);

```

```

247     }
248     else{
249         ledcWrite(pwmChannel, 0);
250     }
251     ControleEstaflag = 1;
252     ControleSwingflag = 0;
253 }
254 else{
255     if (ContSwing > 0) {
256         digitalWrite(motor1Pin1, HIGH);
257         digitalWrite(motor1Pin2, LOW);
258         ledcWrite(pwmChannel, ControleSwing);
259     }
260     else if (ContSwing < 0){
261         digitalWrite(motor1Pin1, LOW);
262         digitalWrite(motor1Pin2, HIGH);
263         ledcWrite(pwmChannel, ControleSwing);
264     }
265     else{
266         ledcWrite(pwmChannel, 0);
267     }
268     ControleEstaflag = 0;
269     ControleSwingflag = 1;
270 }
271
272 nowDisp = millis();
273 if ( nowDisp - startTimeDisp >= Displaytimer ) {
274     //DYSPLAY
275     Serial.print(PositionPendRad, 4);
276     Serial.print(", ");
277     Serial.print(Vf);
278     Serial.print(", ");
279     Serial.print(VoltageSwing);
280     Serial.print(", ");
281     Serial.print(Vfp, 8);
282     Serial.print(", ");

```

```

283     Serial.print(VoltageEsta);
284     Serial.print(",");
285     startTimeDisp = nowDisp;
286 }
287 }
288 // FUNCOES: SENTIDO CONTROLADOR
289
290 int SentidoControle(){
291     if (ContAtual > 0) return 1;
292     else if (ContAtual < 0) return -1;
293     return 0;
294 }
295
296 int SgnPosPend(){
297     if (cos(PositionPendRad) > 0) return 1;
298     else if (cos(PositionPendRad) < 0) return -1;
299     return 0;
300 }
301
302 int SgnSpeed(){
303     if (Vfp > 0.0000001) return 1;
304     else if (Vfp < -0.0000001) return -1;
305     else return 0;
306 }
307
308 // FUNCAO : EQUACAO CONDICIONAL DELTA PARA O DE ESTABILIZACAO
309
310 float delta(){
311     if (PositionPendRad < M_PI){
312         return PositionPendRad;
313     }
314     else if (PositionPendRad > M_PI){
315         return PositionPendRad - 2*M_PI;
316     }
317     else{
318         return M_PI;

```

```

319     }
320 }
321
322 // FUNCAO : EQUACAO CONDICIONAL DELTA PARA O DE SWING UP
323
324 float deltaSwing(){
325     if (PositionPendRad < M_PI){
326         return PositionPendRad;
327     }
328     else if (PositionPendRad > M_PI){
329         return abs(PositionPendRad - 2*M_PI);
330     }
331     else{
332         return M_PI;
333     }
334 }

```

A.3 PROJETOS SOLIDWORKS

Modelamento 3D da roda para obter seus momentos de inércia.

Figura A.3.1 – Modelo roda de reação solidworks

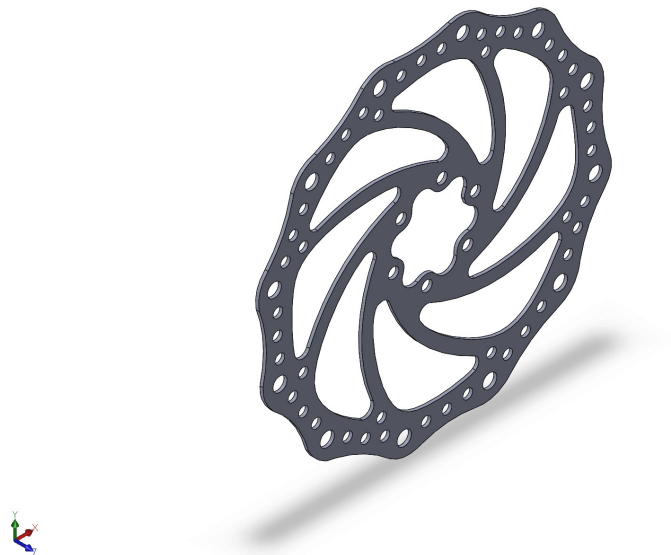
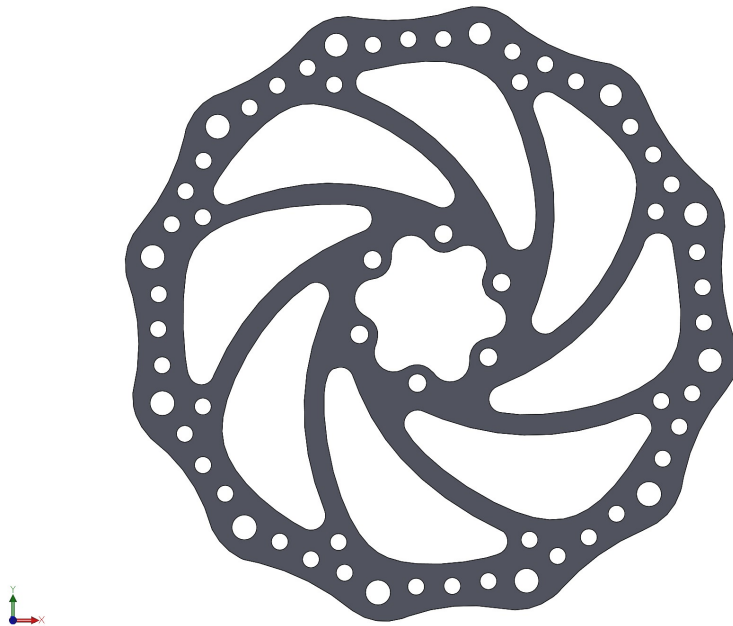


Figura A.3.2 – Modelo roda de reação solidworks vista frontal



A.4 ELIPSE E3

Figura A.4.1 – Tela inicial Elipse E3



Figura A.4.2 – Tela de controle Elipse E3

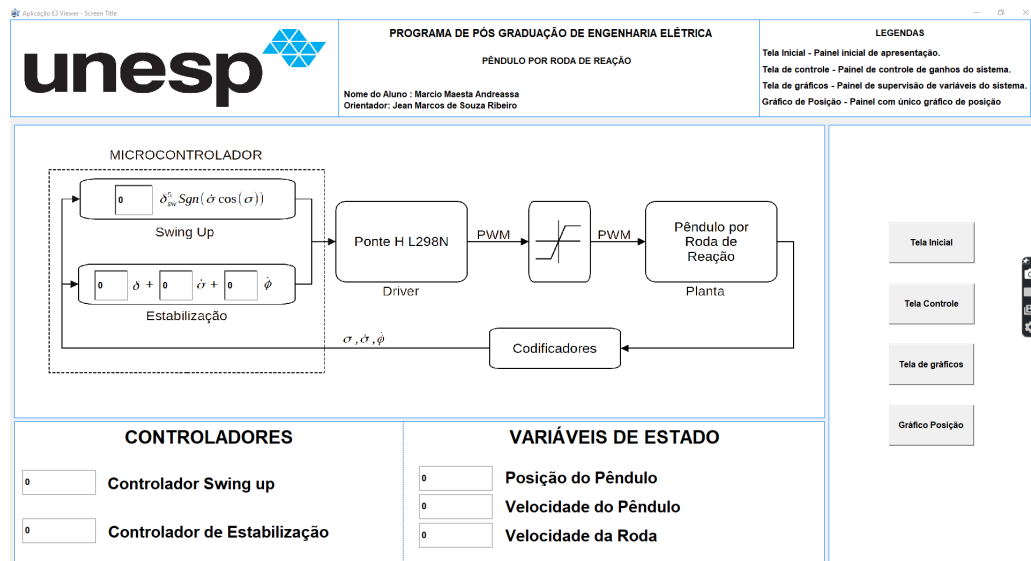


Figura A.4.3 – Tela de gráficos Elipse E3

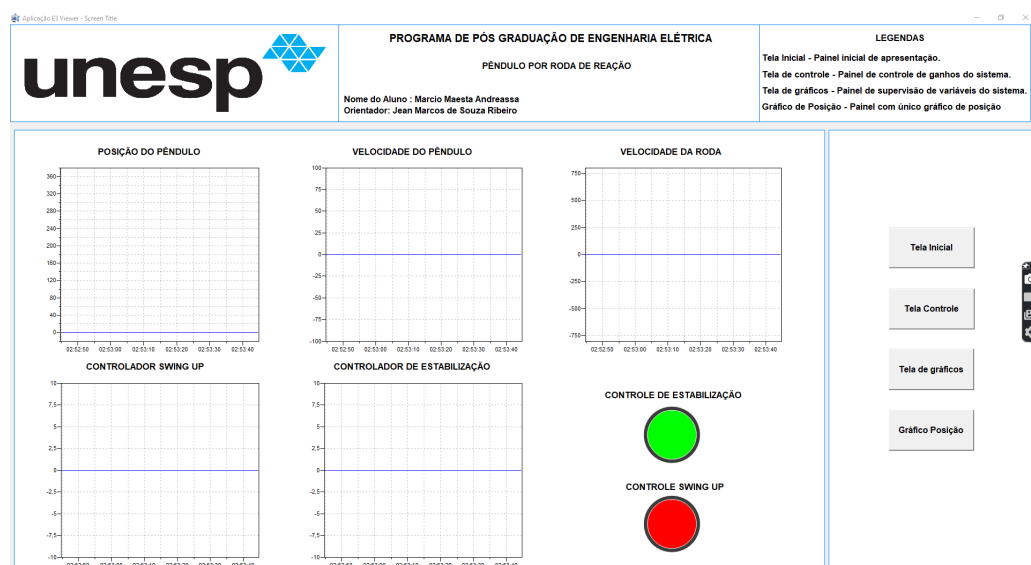


Figura A.4.4 – Gráfico posição Elipse E3

