

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
BACHARELADO EM ENGENHARIA ELÉTRICA**

LUIS GUSTAVO PERES

**MODELAGEM E IMPLEMENTAÇÃO DE PROTÓTIPO DE ENSAIOS DE
PÊNDULO INVERTIDO LINEAR**

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA

2019

LUIS GUSTAVO PERES

**MODELAGEM E IMPLEMENTAÇÃO DE PROTÓTIPO DE ENSAIOS DE
PÊNULO INVERTIDO LINEAR**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia Elétrica, do Departamento Acadêmico de Eletrônica, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Mauricio dos Santos Kaster

PONTA GROSSA

2019



TERMO DE APROVAÇÃO

MODELAGEM E IMPLEMENTAÇÃO DE PROTÓTIPO DE ENSAIOS DE PÊNDULO INVERTIDO
LINEAR

por

LUIS GUSTAVO PERES

Este Trabalho de Conclusão de Curso foi apresentado em 21 de Novembro de 2019 como requisito parcial para a obtenção do título de Bacharel em Engenharia Elétrica. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Dr. Mauricio dos Santos Kaster
Orientador

Prof(a). Dr(a). Fernanda Cristina Correa
Membro Titular

Prof. Dr. Frederic Conrad Janzen
Membro Titular

Prof. Dr. Josmar Ivanqui
Membro Titular e Responsável pelos TCC

Prof. Dr. Sérgio Okida
Coordenador do Curso

AGRADECIMENTOS

Agradeço primeiramente a minha família, minha mãe Dirce e minhas irmãs Juliana e Nayana pela ajuda e apoio durante toda a minha formação.

Agradeço também a minha namorada Amanda de Oliveira Tanaka por todo o incentivo, ajuda e compreensão durante o curso e a realização deste trabalho.

Agradeço ao meu orientador professor Mauricio dos Santos Kaster por toda a dedicação para este trabalho e por me mostrar os caminhos a serem seguidos.

Agradeço todos os professores da UTFPR que fizeram parte da minha graduação.

Enfim, a todos que de alguma forma contribuíram para a conclusão deste trabalho.

RESUMO

PERES, Luis Gustavo. **MODELAGEM E IMPLEMENTAÇÃO DE PROTÓTIPO DE ENSAIOS DE PÊNDBULO INVERTIDO LINEAR**. 2019. 40 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) – Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2019.

O estudo do controle do pêndulo invertido é muito útil pelas características do sistema, este sendo um sistema não linear que deve ser controlado em seu ponto de equilíbrio instável, sendo uma ótima ferramenta para o estudo tanto de teorias clássicas como teorias modernas de controle. Esse trabalho propõe a construção de um protótipo de pêndulo invertido em laboratório para o estudo de técnicas de controle, assim como a modelagem matemática e o desenvolvimento de um modelo computacional onde podem ser feitos testes e comparar os resultados com a planta física. É proposto um controle em dupla malha utilizando dois controladores, um PID (proporcional-integral-derivativo) e um PI (proporcional-integral) sendo um responsável pelo controle do equilíbrio da haste e outro pela posição do carro.

Palavras-chave: Pêndulo Invertido. Controle PID. Controle dupla malha. Microcontrolador. Modelagem e simulação Computacional.

ABSTRACT

PERES, Luis Gustavo. **MODELLING AND IMPLEMENTATION OF LINEAR INVERTED PENDULUM PROTOTYPE**. 2019. 40 p. Final Coursework (Bachelor's Degree in Electrical Engineering) – Federal University of Technology – Paraná. Ponta Grossa, 2019.

The study of the inverted pendulum is very useful due to the system characteristics, it is a non-linear system which must be controlled at the instability point, being a great tool for the study of classical control theory and modern control theory. This final coursework proposes the construction of a prototype of the inverted pendulum for the study of control theory, the mathematical modeling, and the development of a computational model where tests can be done and results can be compared to the ones from the prototype. It is proposed the use of a system with two control loops using a PID (proportional-integrative-derivative) and a PI (proportional-integrative) controllers, being one responsible for the equilibrium of the rod and the other for the position of the car.

Palavras-chave: Inverted Pendulum. PID Control. Double loop control. Microcontroller. Computational Modeling and Simulation.

LISTA DE ILUSTRAÇÕES

Figura 1 – Esquemas de controle.	12
Figura 2 – Proposta de protótipo do pêndulo invertido.	12
Figura 3 – Diagrama de corpo livre do pêndulo invertido.	14
Figura 4 – Comparação entre simulação e planta física com o carro fixo.	17
Figura 5 – Comparação entre simulação e planta física com o carro solto.	17
Figura 6 – Esquema de controle.	20
Figura 7 – Lugar das raízes da função de transferência do ângulo.	21
Figura 8 – Lugar das raízes com controlador PID da função $P_{z\theta}$	21
Figura 9 – Resposta ao degrau com o controlador C_1	22
Figura 10 – Lugar das raízes da função de transferência da posição.	23
Figura 11 – Lugar das raízes com controlador da função P_{zx}	23
Figura 12 – Modelo da planta não linear.	25
Figura 13 – PWM e conversor A/D.	26
Figura 14 – Modelo do microcontrolador.	26
Figura 15 – Blocos do potenciômetro, encoder e acionamento.	27
Figura 16 – Modelo computacional completo.	27
Figura 17 – Protótipo do pêndulo invertido.	29
Figura 18 – Esquemático do <i>software</i>	31
Figura 19 – Comparação dos resultados de simulação da malha de equilíbrio.	33
Figura 20 – Resultado da malha de equilíbrio aplicada ao protótipo.	34
Figura 21 – Comparação dos resultados de simulação da malha de posição.	34
Figura 22 – Resultado da malha de posição aplicada ao protótipo.	34
Figura 23 – Comparação dos resultados de simulação com dupla malha.	35
Figura 24 – Resultado de dupla malha aplicada ao protótipo.	36
Figura 25 – Distúrbio aplicado à haste com dupla malha.	36

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

SIGLAS

FT	Função de Transferência
IHM	Interface Homem-Máquina
LCD	<i>Liquid Crystal Display</i>
LQR	<i>Linear Quadratic Regulator</i>
LR	Lugar das raízes
MF	Malha Fechada
MFA	<i>Model-Free Adaptive</i>
PID	Proporcional, Integral, Derivativo
PWM	<i>Pulse Width Modulation</i>
SISOTOOL	<i>Single-Input, Single-Output Tool</i>
TCC	Trabalho de Conclusão de Curso
ZOH	<i>Zero Order Hold</i>

LISTA DE SÍMBOLOS

m_1	Massa da haste	[kg]
m_2	Massa do carro	[kg]
l	Distância do centro de massa a extremidade da haste	[m]
J	Momento de inércia da haste	[kg·m ²]
T_s	Tempo de amostragem	[s]
θ	Ângulo da haste	[Graus]
μ_e	Coeficiente de atrito entre a haste e o eixo	
μ_c	Coeficiente de atrito entre o carro e o trilho	
ξ	Coeficiente de amortecimento	

SUMÁRIO

1	INTRODUÇÃO	10
1.1	PROBLEMA	10
1.2	JUSTIFICATIVA	11
1.3	OBJETIVO	11
1.3.1	Objetivos Específicos	11
1.4	ORGANIZAÇÃO DO TRABALHO	11
2	MODELAGEM MATEMÁTICA	14
2.1	MODELO MATEMÁTICO DA PLANTA	14
2.2	LINEARIZAÇÃO	16
2.3	FUNÇÕES DE TRANSFERÊNCIA CONTÍNUAS	16
2.4	DEFINIÇÃO DOS PARÂMETROS DO SISTEMA	16
2.5	FUNÇÕES DE TRANSFERÊNCIA DISCRETAS	18
3	ESTRATÉGIA DE CONTROLE	20
4	IMPLEMENTAÇÃO DO SISTEMA	25
4.1	MODELAGEM COMPUTACIONAL	25
4.2	ESTRUTURA DA PLANTA FÍSICA	28
4.3	DESENVOLVIMENTO DO SOFTWARE DE CONTROLE	29
5	RESULTADOS EXPERIMENTAIS	32
5.1	MALHA DE EQUILÍBRIO	32
5.2	MALHA DE POSIÇÃO	32
5.3	MALHA DUPLA	34
6	CONCLUSÕES	37
	REFERÊNCIAS	39

1 INTRODUÇÃO

O pêndulo invertido é um sistema importante no estudo da teoria de controle e tem sido utilizado amplamente para investigar e desenvolver novas técnicas de controle que podem tratar efetivamente com sistemas não lineares, sendo que o principal desafio é o sistema ser não linear, sub-atuado, que deve ser controlado em seu ponto de equilíbrio instável (CHANCHAREON; SANGVERAPHUNSIRI; CHANTRANUWATHANA, 2006). Ainda segundo Åström e Furuta (2000) o pêndulo invertido é uma ferramenta clássica nos laboratórios de controle desde 1950. Ele possui um ponto de equilíbrio instável onde se busca tornar o sistema estável através da aplicação de sistemas de controle.

Na literatura são encontrados diversos exemplos de estudos e aplicações de diferentes tipos de controladores, como em Vendramini e Silva (2010) onde é mostrado a utilização de um controle PID (Proporcional-Integral-Derivativo) assim como de um controle inteligente adaptativo, chamado MFA (*Model-Free Adaptive*). Em Kumar e Jerome (2013) é utilizado um controlador LQR (Regulador Quadrático Linear) que apresenta uma boa performance dinâmica e robustez a distúrbios externos. Em Filho et al. (2013) pode ser vista a utilização de um controlador *fuzzy* e Cho e Jung (2003) utiliza redes neurais para o controle do pêndulo invertido.

Diversos sistemas utilizam os conceitos de controle aplicados ao pêndulo invertido como, por exemplo, foguetes, mísseis, drones, meios de transporte alternativos de uma ou duas rodas autobalanceados como apresentados em Baloh e Parent (2003) e em Huang (2010), robôs (NAWAWI; AHMAD; OSMAN, 2008), entre outros. Tais sistemas são inviáveis sem o uso de controle.

O presente trabalho se propõe a construir um protótipo de pêndulo invertido linear controlado digitalmente com microcontrolador. Neste trabalho será utilizado um controlador PID que, como mostrado em Wang (2011), pode ser utilizado para se obter uma resposta satisfatória do sistema mesmo que este possua mais de um grau de liberdade. O PID é um controlador linear tradicional, amplamente utilizado em uma vasta gama de aplicações e com ferramentas bastante consolidadas de projeto.

A modelagem matemática necessária para o desenvolvimento do modelo computacional para simulação e para o projeto do controlador foi realizada utilizando a mecânica Newtoniana (OGATA, 1982). O protótipo físico possui a parte mecânica, a elétrica e um microcontrolador onde o controle digital é embarcado.

1.1 PROBLEMA

O problema consiste em construir e controlar um sistema de pêndulo invertido de modo com que ele permaneça equilibrado na posição vertical utilizando um controle PID digital aplicado por meio de um microcontrolador.

1.2 JUSTIFICATIVA

Sendo o pêndulo invertido uma importante ferramenta para o estudo de técnicas de controle, que possibilita o estudo tanto de teorias clássicas como teorias modernas de controle, a planta poderá ser utilizada em aulas de laboratório para a experimentação em disciplinas da graduação, pois envolve conhecimento de técnicas de controle contínuo e discreto. Esta planta também pode auxiliar em pesquisas, visto que diversos tipos de controladores mais avançados podem ser estudados utilizando esta ferramenta, como controle adaptativo, controle ótimo, modos deslizantes, entre outros.

1.3 OBJETIVO

Construir um pêndulo invertido, desenvolver seu modelo computacional e aplicar um controle digital.

1.3.1 Objetivos Específicos

- Obtenção do modelo matemático da planta.
- Desenvolvimento de um modelo para simulação computacional.
- Construção de uma planta física para experimentações em laboratório.
- Desenvolvimento de um controle PID (Proporcional, integral e derivativo).
- Implementação do controle digital utilizando microcontrolador.
- Elaborar documentação técnica para o TCC e para subsidiar novos desenvolvimentos e produção de artigos científicos.

1.4 ORGANIZAÇÃO DO TRABALHO

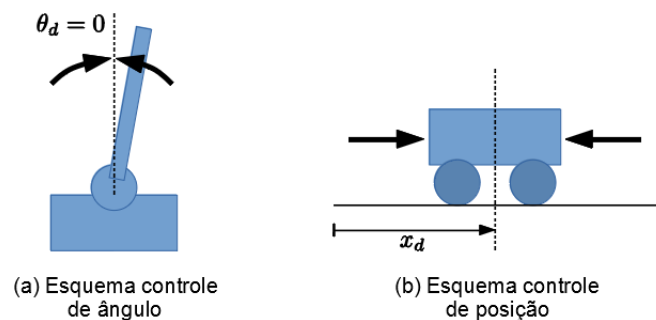
No capítulo 2, é deduzido o modelo matemático do sistema, necessário para o projeto do controlador e simulação antes de ser implementado na planta física. Este modelo foi deduzido por meio da mecânica newtoniana, que posteriormente foi linearizado para se obter as funções de transferência que descrevem o comportamento do sistema. Também se obtém a função de transferência discreta usando o método de discretização ZOH (*Zero Order Hold*) para possibilitar o projeto do controle discreto digital.

No capítulo 3 é realizado o projeto de controle, que faz uso da ferramenta SISOTOOL (Single-Input, Single-Output Tool), do Matlab, que possibilita o projeto de controle contínuo de uma planta contínua qualquer. Ela fornece basicamente duas formas clássicas de projeto de controle: Lugar das Raízes e Diagrama de Bode. Dentro dessa ferramenta interativa, pode-se testar o ganho e as posições de polos e zeros do controlador dentro de uma das formas de projeto (Lugar das raízes

ou Bode) e observar em tempo real seu efeito sobre o transitório de saída até que os requisitos de projeto sejam satisfeitos.

Foram utilizados dois controladores, um para controlar o ângulo da haste em relação ao carro e outro para controlar a posição linear do carro, como apresentado na figura 1. Primeiramente foi definido o controle do ângulo, responsável pelo equilíbrio da haste, e somente após este estar funcionando corretamente foi feito o projeto do controle de posição. O controlador total é uma composição dos dois controles dispostos topologicamente em paralelo com as saídas de controle individuais somadas.

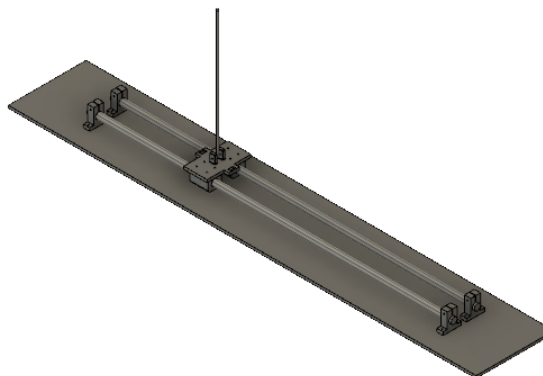
Figura 1 – Esquemas de controle.



No capítulo 4, a partir do modelo matemático, um modelo computacional foi construído utilizando o *software* MatLab/Simulink, descrevendo o modelo não linear da planta, assim como os modelos dos componentes utilizados, como potenciômetro, *encoder*, circuito de acionamento e microcontrolador.

O funcionamento foi verificado através da construção de um protótipo físico implementado em laboratório. O protótipo foi construído utilizando perfis de alumínio, o carro é movimentado por um motor de corrente contínua acoplado por meio de polias e correia GT2, o carro e peças para fixação do motor, microcontrolador e polias foram projetadas no *software* Fusion 360 e confeccionadas utilizando uma impressora 3D. A figura 2 mostra uma proposta do protótipo construído.

Figura 2 – Proposta de protótipo do pêndulo invertido.



O capítulo 5 mostra os resultados da implementação do controle, o funcionamento dos controladores foi testado primeiramente no modelo computacional antes de ser aplicado ao protótipo,

devido ao fato da facilidade de alteração do modelo e da análise dos sinais, que estão todos acessíveis na ferramenta computacional sem a necessidade de uso de instrumentos, e da disponibilidade de uma série de recursos computacionais de análise desses sinais. Posteriormente, o teste foi aplicado no protótipo físico, usando os resultados obtidos na simulação e foi feita a comparação dos resultados, onde tais resultados são similares.

Todos esses procedimentos e resultados foram documentados com vistas à elaboração do próprio TCC, bem como de artigos científicos e fornecer documentação de referência para futuros trabalhos.

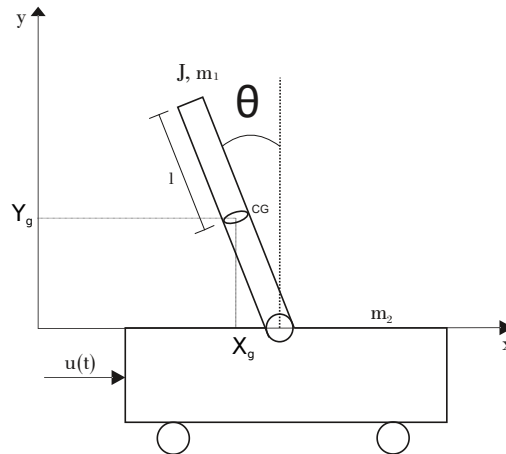
2 MODELAGEM MATEMÁTICA

No estudo de sistemas de controle faz-se necessário o conhecimento de um modelo matemático que descreva o sistema físico ao qual o controle será aplicado, este modelo consiste em equações diferenciais que descrevem o comportamento dinâmico do sistema. Este capítulo irá apresentar o equacionamento matemático do sistema do pêndulo invertido.

2.1 MODELO MATEMÁTICO DA PLANTA

Observando o esquema apresentado na figura 3 tem-se que a plataforma se move com a aplicação de uma força $u(t)$ dada em Newtons, m_1 e m_2 representam as massas em quilogramas da haste e da plataforma respectivamente, θ é o ângulo em radianos entre a haste e o eixo vertical y , e l representa a distância em metros entre o centro de massa da haste e sua extremidade.

Figura 3 – Diagrama de corpo livre do pêndulo invertido.



Para obter as equações de movimento que descrevem o modelo dinâmico do sistema utilizou-se a mecânica Newtoniana. Primeiramente são determinadas as coordenadas generalizadas do centro de massa da haste, estabelecendo as coordenadas (x, y) do centro de massa como (X_g, Y_g) .

$$X_g = x - l \sin(\theta) \quad (1)$$

$$Y_g = l \cos(\theta) \quad (2)$$

Assim, as equações que determinam a velocidade e a aceleração do centro de massa podem ser obtidas simplesmente derivando as equações (1) e (2), resultando em:

$$\dot{X}_g = \dot{x} - l \cos(\theta) \dot{\theta} \quad (3)$$

$$\dot{Y}_g = -l \sin(\theta) \dot{\theta} \quad (4)$$

$$\ddot{X}_g = \ddot{x} - l \cos(\theta) \ddot{\theta} + l \sin(\theta) \dot{\theta}^2 \quad (5)$$

$$\ddot{Y}_g = -l \operatorname{sen}(\theta)\ddot{\theta} - l \cos(\theta)\dot{\theta}^2 \quad (6)$$

A componente horizontal F_x da força aplicada na haste é:

$$\sum F_x = m_1 \ddot{X}_g \quad (7)$$

$$F_x = m_1 \ddot{x} - m_1 l \cos(\theta)\ddot{\theta} + l \operatorname{sen}(\theta)\dot{\theta}^2 \quad (8)$$

A componente vertical F_y da força aplicada na haste é:

$$\sum F_y = m_1 \ddot{Y}_g \quad (9)$$

$$F_y = -m_1 l \operatorname{sen}(\theta)\ddot{\theta} - m_1 l \cos(\theta)\dot{\theta}^2 + m_1 g \quad (10)$$

Então o movimento rotacional da haste em torno de seu centro de massa pode ser descrito como:

$$\sum M_g = J\ddot{\theta} \quad (11)$$

$$J\ddot{\theta} = F_y l \operatorname{sen}(\theta) + F_x l \cos(\theta) - \mu_e \dot{\theta} \quad (12)$$

Onde μ_e é o coeficiente de atrito entre a haste e o eixo. Substituindo (8) e (10) em (12):

$$J\ddot{\theta} = (-m_1 l \operatorname{sen}(\theta)\ddot{\theta} - m_1 l \cos(\theta)\dot{\theta}^2 + m_1 g) l \operatorname{sen}(\theta) + (m_1 \ddot{x} - m_1 l \cos(\theta)\ddot{\theta} + l \operatorname{sen}(\theta)\dot{\theta}^2) l \cos(\theta) - \mu_e \dot{\theta} \quad (13)$$

Isolando $\ddot{\theta}$ tem-se a equação diferencial que descreve a aceleração do ângulo da haste:

$$\ddot{\theta} = \frac{m_1 l \cos(\theta)\ddot{x} + m_1 g l \operatorname{sen}(\theta) - \mu_e \dot{\theta}}{m l^2 + J} \quad (14)$$

O movimento horizontal do carro pode ser descrito como:

$$\sum F_c = m_2 \ddot{x} \quad (15)$$

$$m_2 \ddot{x} = u - F_x - \mu_c \dot{x} \quad (16)$$

Onde μ_c é o coeficiente de atrito entre o carro e o trilho. Substituindo (8) em (16):

$$m_2 \ddot{x} = u - m_1 \ddot{x} + m_1 l \cos(\theta)\ddot{\theta} - m_1 l \operatorname{sen}(\theta)\dot{\theta}^2 - \mu_c \dot{x} \quad (17)$$

Isolando \ddot{x} tem-se a equação diferencial que descreve a aceleração do carro:

$$\ddot{x} = \frac{u + m_1 l \cos(\theta)\ddot{\theta} - m_1 l \operatorname{sen}(\theta)\dot{\theta}^2 - \mu_c \dot{x}}{m_1 + m_2} \quad (18)$$

Portanto, o modelo não linear do pêndulo invertido é descrito por duas equações diferenciais:

$$(m l^2 + J)\ddot{\theta} = m_1 l \cos(\theta)\ddot{x} + m_1 g l \operatorname{sen}(\theta) - \mu_e \dot{\theta} \quad (19)$$

$$(m_1 + m_2)\ddot{x} = u + m_1 l \cos(\theta)\ddot{\theta} - m_1 l \operatorname{sen}(\theta)\dot{\theta}^2 - \mu_c \dot{x} \quad (20)$$

2.2 LINEARIZAÇÃO

Embora existam técnicas de projeto não lineares, elas envolvem uma complexidade bem maior que as técnicas lineares tradicionais, que são suficientes para o trabalho em questão. Dessa forma, é preciso primeiramente linearizar o modelo em torno do ponto de operação para então poder usar tais técnicas.

Este sistema pode ser linearizado considerando-se que o sistema deve ser controlado em seu ponto de equilíbrio instável, ou seja, $\theta \approx 0$ e $x \approx 0$, assim pode-se considerar $\sin(\theta) \approx \theta$ e $\cos(\theta) \approx 1$. Neste ponto a velocidade angular também é muito próxima de zero, $\dot{\theta} \approx 0$, assim $\theta\dot{\theta}^2 \approx 0$. Aplicando essas considerações em (19) e (20), tem-se o sistema linearizado:

$$(ml^2 + J)\ddot{\theta} = m_1l\ddot{x} + m_1gl\theta - \mu_e\dot{\theta} \quad (21)$$

$$(m_1 + m_2)\ddot{x} = u + m_1l\ddot{\theta} - \mu_s\dot{x} \quad (22)$$

2.3 FUNÇÕES DE TRANSFERÊNCIA CONTÍNUAS

Aplicando a transformada de Laplace nas equações (21) e (22) obtém-se:

$$(ml^2 + J)s^2\Theta = m_1ls^2X + m_1gl\Theta - \mu_es\Theta \quad (23)$$

$$(m_1 + m_2)s^2X = U + m_1ls^2\Theta - \mu_ssX \quad (24)$$

Manipulando as equações (23) e (24) tem-se as funções de transferência que descrevem o sistema:

$$\frac{\Theta(s)}{U(s)} = \frac{m_1ls}{cs^3 + (\mu_s a + \mu_e b)s^2 + (\mu_s \mu_e - bm_1gl)s - am_1gl} \quad (25)$$

$$\frac{X(s)}{U(s)} = \frac{as^2 + \mu_es - m_1gl}{cs^4 + (\mu_s a + \mu_e b)s^3 + (\mu_s \mu_e - bm_1gl)s^2 - am_1gl} \quad (26)$$

onde:

$$\begin{cases} a = J + m_1l^2 \\ b = m_1 + m_2 \\ c = ab - (m_1l)^2. \end{cases}$$

2.4 DEFINIÇÃO DOS PARÂMETROS DO SISTEMA

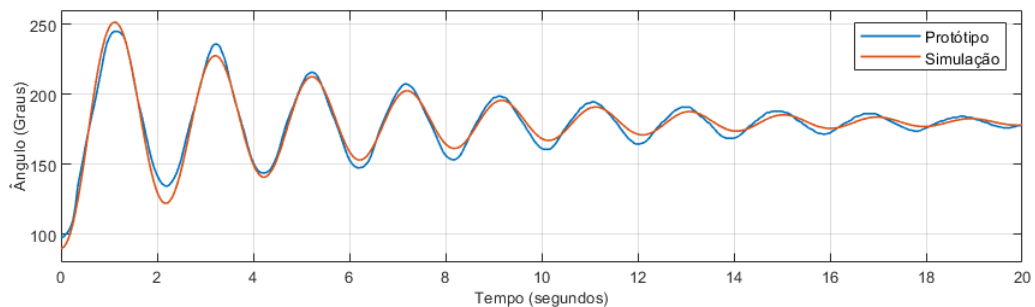
Os valores como massas e distâncias podem ser facilmente medidos na planta física, porém algumas constantes como os coeficientes de atrito e momentos de inércia apresentam uma maior dificuldade para serem definidos. Assim, com a ajuda do modelo computacional do sistema foi possível determinar os mesmos.

Na literatura podem ser encontrados diferentes maneiras para a identificação do sistema do pêndulo invertido, redes neurais são utilizadas por Gautam (2016), Naik e Cochran (2012) utiliza o movimento pendular do sistema e séries de potências, este método consegue reconstruir as equações diferenciais de forma precisa com menos amostras que o método utilizando redes neurais.

Para a identificação do sistema, tendo todos os outros coeficientes medidos e aplicados no modelo computacional, aplicou-se primeiramente valores aleatórios ao coeficiente de atrito da haste do pêndulo e seu ponto de fixação (μ_e) e seu momento de inércia (J) e então um valor alto para o coeficiente de atrito entre o carro e o trilho (μ_c), de modo com que este permanecesse imóvel e iniciou-se a simulação em malha aberta com o ângulo inicial do pêndulo em 90° até que o mesmo estabilizasse em 180° .

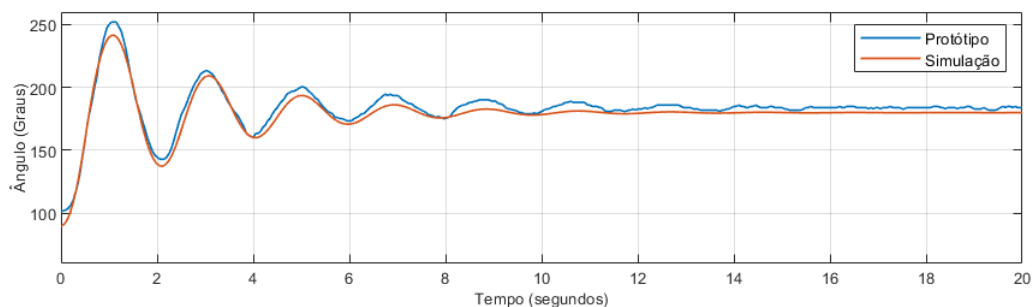
Com o carro preso sem poder se movimentar sobre o trilho, a forma de onda foi adquirida da mesma forma e os coeficientes foram sendo ajustados de forma que o coeficiente de atrito altera diretamente a quantidade de oscilações até a estabilização do ângulo e o momento de inércia altera diretamente a frequência de oscilação. A comparação do comportamento da simulação com os coeficientes já ajustados e do comportamento da planta física podem ser vistos na figura 4.

Figura 4 – Comparação entre simulação e planta física com o carro fixo.



Tendo estes coeficientes definidos, o mesmo foi feito com o carro solto, na simulação colocou-se inicialmente um valor aleatório para o coeficiente de atrito entre o trilho e o carro de forma que este pudesse se movimentar, e na planta física permitiu-se que o mesmo corresse livremente sobre o trilho. Assim o coeficiente foi sendo ajustado até que o comportamento da planta e o da simulação fossem os mesmos. A figura 5 mostra a comparação entre o comportamento do modelo computacional e planta física.

Figura 5 – Comparação entre simulação e planta física com o carro solto.



Desta forma, os valores das constantes foram definidos conforme a tabela 1.

Tabela 1 – Constantes definidas para a planta física.

Símbolo	Valor	Descrição
m_1	0.2	Massa da haste (kg)
m_2	0.35	Massa do carro (kg)
l	0.65	Distância da extremidade ao centro de massa da haste (m)
μ_e	0.045	Coefficiente de atrito entre a haste e seu eixo
μ_c	0.35	Coefficiente de atrito entre o carro e o trilho
J	0.038	Momento de inércia da haste (kg.m ²)

Tomando como base estes parâmetros, substituindo-os na equações (25) e (26), obtêm-se as funções de transferência contínuas:

$$G_{\Theta}(s) = \frac{0.13s}{0.05048s^3 + 0.06763s^2 - 0.6857s - 0.4464} \quad (27)$$

$$G_X(s) = \frac{0.1225s^2 + 0.045s - 1.275}{0.05048s^4 + 0.06763s^3 - 0.6857s^2 - 0.4464s} \quad (28)$$

2.5 FUNÇÕES DE TRANSFERÊNCIA DISCRETAS

Para a obtenção das funções de transferência discretas deve-se estipular um período de amostragem T_s . A escolha de um valor apropriado depende principalmente da dinâmica do sistema e não deve ser muito baixo, sob o risco de perda de informação devido à sub-amostragem dos sinais, nem muito alta devido à necessidade de grande precisão numérica para cômputo de valores muito pequenos (com muitas casas decimais) e ao tempo restrito de processamento de um passo de controle pelo microcontrolador, que deve ser feito dentro de uma janela de tempo entre dois instantes T_s . A discussão mais aprofundada sobre esse assunto está além do escopo deste trabalho.

Seguindo essas diretrizes e fazendo alguns testes, estipulou-se um período de amostragem $T_s = 10$ ms, este tempo não causa perda de informações da planta e disponibiliza tempo suficiente para o processamento no microcontrolador. Realizando-se a discretização das funções de transferência contínuas através do método ZOH, que utiliza um sustentador de ordem zero para manter o sinal constante entre os instantes de amostragem, obtêm-se as funções de transferência discretas do sistema:

$$G_{\Theta}(z) = \frac{1.282E - 4z^2 - 5.713E - 7z + 1.201E - 4}{z^3 - 2.988z^2 + 2.975z - 0.9867} \quad (29)$$

$$G_X(z) = \frac{1.21E - 4z^3 - 1.21E - 4z^2 - 1.203E - 4z + 1.201E - 4}{z^4 - 3.988z^3 + 5.963z^2 - 3.961z + 0.9867} \quad (30)$$

O método ZOH foi utilizado pois ao aplicar o controle por meio do microcontrolador, os sinais de entrada da posição e do ângulo, e o sinal de controle se mantêm constantes entre os instantes de amostragem da mesma forma que ocorre com o sustentador de ordem zero.

As FT discretas tem utilidade apenas para o projeto do controle discreto, haja visto que a planta é sempre contínua, pois representa um sistema físico real. Dessa forma, após a realização do projeto, normalmente a FT discreta não são mais utilizadas.

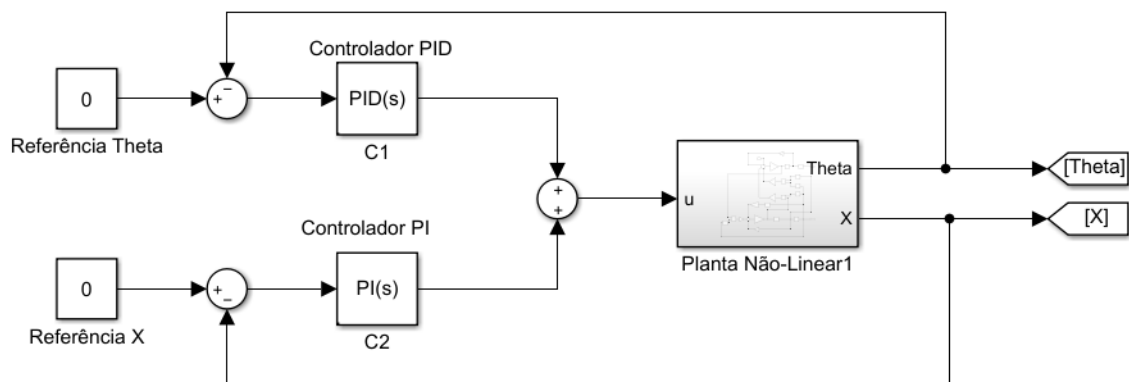
Este capítulo apresentou o diagrama do corpo livre e as equações diferenciais dele obtidas, sua linearização e discretização, bem como os ensaios que permitiram a obtenção dos parâmetros da planta. No próximo capítulo será apresentada a estratégia de controle usada visando manter o pêndulo em equilíbrio em uma dada posição longitudinal do carro pré-estabelecida.

3 ESTRATÉGIA DE CONTROLE

Este capítulo irá apresentar a estrutura do controlador adotado, o projeto das duas malhas (de equilíbrio e de posição) e os resultados obtidos do sistema em operação em malha fechada.

A estratégia de controle adotada é a mesma utilizada por Vendramini e Silva (2010) e Wang (2011), que consiste em somar as ações de controle de dois controles, um PID para o controle do ângulo e um PI para o controle da posição do pêndulo, conforme mostrado na figura 6.

Figura 6 – Esquema de controle.



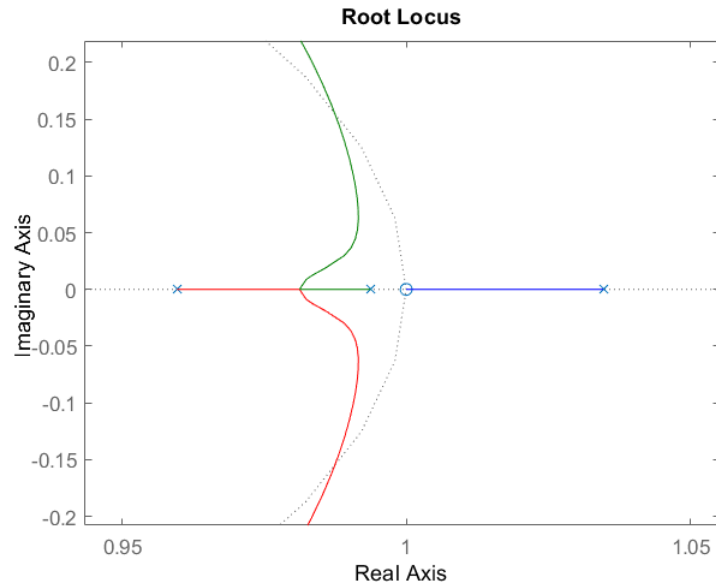
Embora não seja tão evidente, é importante citar que é fundamental que esses dois controles tenham constantes de tempo diferentes, sendo a malha de equilíbrio da haste necessariamente mais rápida que a de posição do carro. O equilíbrio só é viável se houver ações rápidas de deslocamento do carro, enquanto a posição dele pode (e deve) ser ajustada mais lentamente usando pequenos desequilíbrios na malha de equilíbrio para que o carro possa ser direcionado para a posição desejada mais lentamente.

Levando isso em consideração, o projeto dos controladores deve ser realizado em partes. Primeiramente foi realizado o projeto da malha de equilíbrio, onde para este projeto utilizou-se a FT da equação (29). Na figura 7 pode-se observar o diagrama do lugar das raízes desta função de transferência que mostra a posição dos polos e zeros assim como sua trajetória. Nota-se que a planta possui um polo fora do círculo unitário caracterizando assim sua instabilidade.

Utilizando a ferramenta interativa SISOTOOL do Matlab, onde o diagrama do lugar das raízes e a resposta ao degrau unitário são plotados simultaneamente, é possível inserir os polos e zeros do controlador e posicioná-los apropriadamente, observando o comportamento do diagrama do lugar das raízes e da resposta ao degrau em tempo real.

Assim, usando-se a estratégia de cancelamento polo-zero, buscou-se primeiramente o cancelamento de um dos polos da planta com um dos zeros do controlador, o segundo zero foi posicionado mais à esquerda para ajudar na estabilidade (trazer a trajetória do LR mais para o interior do círculo) e resposta (polos de MF mais próximos da origem resultam respostas mais rápidas), e por

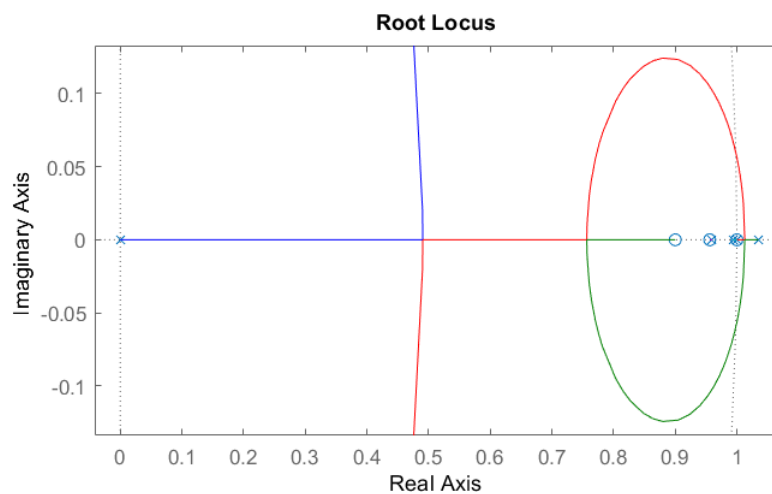
Figura 7 – Lugar das raízes da função de transferência do ângulo.



fim o ganho do controlador foi sendo ajustado até ser obtida uma resposta ao degrau satisfatória para o sistema.

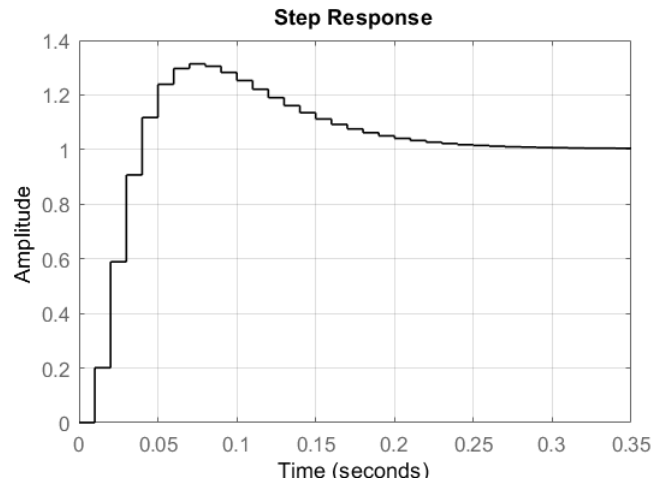
O máximo percentual de sobressinal foi escolhido conforme a sugestão em Ogata (1982), onde é sugerido que sistemas-padrão de segunda ordem não tenham o coeficiente de amortecimento $\xi < 0.4$, isto implica em um percentual de sobressinal máximo de 25%, e estipulou-se que o tempo de acomodação do sistema deveria ser menor que 250 ms. Após ajustes no controlador, para manter o tempo de acomodação abaixo do tempo estipulado, pois esta malha necessita ser rápida, aceitou-se um percentual de sobressinal de 30% que está um pouco acima do sugerido. O diagrama do lugar das raízes pode ser visto na figura 8 e a resposta ao degrau na figura 9.

Figura 8 – Lugar das raízes com controlador PID da função $P_{z\theta}$.



Como resultado, a eq. (31) representa a FT do controlador PID projetado para a malha de

Figura 9 – Resposta ao degrau com o controlador C_1 .



equilíbrio.

$$C_1(z) = \frac{1824z^2 - 3387z + 1570}{z^2 - z} \quad (31)$$

Antes do projeto da segunda malha de controle, aplicou-se primeiramente a equação a diferenças diretamente na planta não linear no modelo computacional para verificar o funcionamento correto do controlador, e posteriormente, no modelo computacional sendo aplicado na planta por meio do modelo de um microcontrolador, este modelo será mostrado no capítulo 4. Com o controle funcionando corretamente na simulação, o mesmo foi aplicado ao protótipo.

Tendo o controlador da malha de equilíbrio definido, é possível fazer o ajuste do controle da malha de posição. Como dito anteriormente, este controlador deve ter uma dinâmica mais lenta em relação ao da malha de equilíbrio, pois primeiramente deve ser garantido o equilíbrio para somente então a plataforma ser levada à posição desejada.

Aqui optou-se por utilizar um controlador PI, que foi definido de forma similar ao outro controlador. Foi utilizado um controlador PI pois a ação derivativa do controle produz uma resposta proporcional a velocidade de variação do erro, sendo bastante sensível a ruído, aumentando sua estabilidade e diminuindo o tempo de resposta do sistema, como esta malha necessita de um tempo de resposta mais lento que a malha de equilíbrio, esta ação de controle não foi utilizada. No diagrama do lugar das raízes observa-se um polo fora do círculo unitário, este independente da posição do zero do controlador e não é possível trazer a trajetória do LR para dentro do círculo. Assim, buscou-se um controlador onde este fosse o único ponto de instabilidade, com um zero posicionado bem próximo a esse polo para reduzir seu efeito de instabilidade, e verificou-se a resposta em conjunto com a malha de equilíbrio no modelo computacional. A figura 10 mostra o diagrama do lugar das raízes da função de transferência da posição e a figura 11 mostra o diagrama do lugar das raízes com o controlador.

Este controlador foi definido de forma interativa, da mesma forma que o controlador da malha de equilíbrio, porém como este sempre é instável, foram feitos testes adicionais aplicando este controlador no modelo computacional junto a malha de equilíbrio. Após o teste no modelo computacional observou-se o funcionamento correto do controlador da malha de posição, definido na

Figura 10 – Lugar das raízes da função de transferência da posição.

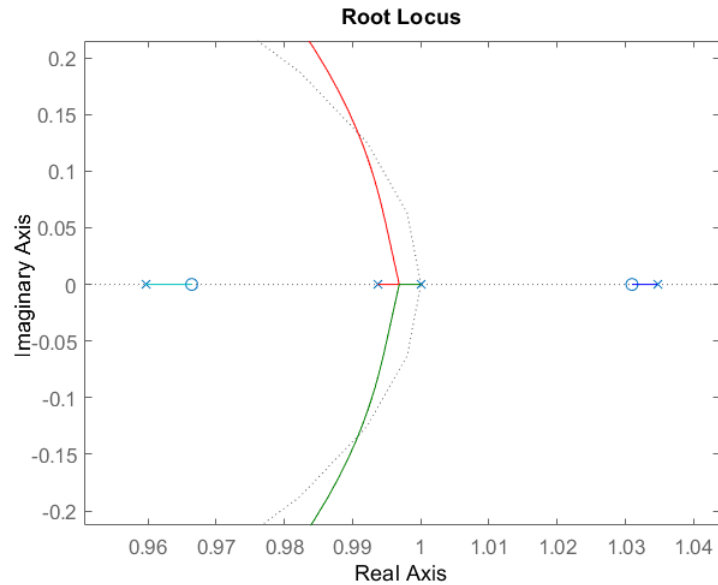
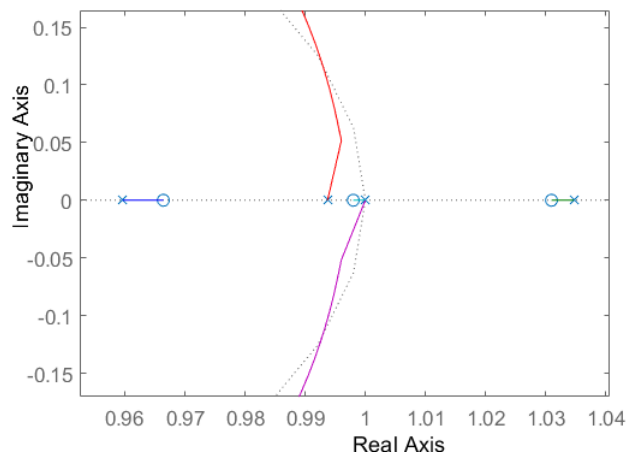


Figura 11 – Lugar das raízes com controlador da função P_{zx} .



equação (32).

$$C_2 = \frac{70z - 69.86}{z - 1} \quad (32)$$

Com o funcionamento correto no modelo computacional o controle pôde ser aplicado diretamente no protótipo e fazer um ajuste fino das malhas de controle. Inicialmente verificou-se uma grande oscilação ao tentar manter o equilíbrio. Assim, foram inseridos dois ganhos, K_1 e K_2 , multiplicando o sinal de cada controlador, que passam a ser $C'_1 = K_1 \cdot C_1$ e $C'_2 = K_2 \cdot C_2$, e os mesmos foram sendo ajustados até que o funcionamento em conjunto das malhas tivesse um bom comportamento. O ganho da malha de equilíbrio foi reduzido de modo a se ter um controle mais suave do ângulo, pois o motor estava dando trancos muito intensos no carro, causando muita vibração. Assim, por meio de testes de diferentes valores para K_1 e K_2 no protótipo, os ganhos definidos foram $K_1 = 0.5$ e $K_2 = 1$. Embutindo estes ganhos diretamente na equação a diferenças dos controladores, tem-se

as expressões das duas malhas do controlador:

$$C'_1 = \frac{912z^2 - 1693.5z + 785}{z^2 - z} \quad (33)$$

$$C'_2 = \frac{70z - 69.86}{z - 1} \quad (34)$$

Este capítulo apresentou a estratégia de controle de duas malhas em paralelo, uma para controle do equilíbrio e outra para o controle da posição do carro, os projetos dessas duas malhas usando Lugar das Raízes na ferramenta SISOTOOL e as formas de onda da resposta do sistema através de simulação e de experimentação no protótipo.

O próximo capítulo irá apresentar detalhes sobre a implementação do modelo computacional e do protótipo.

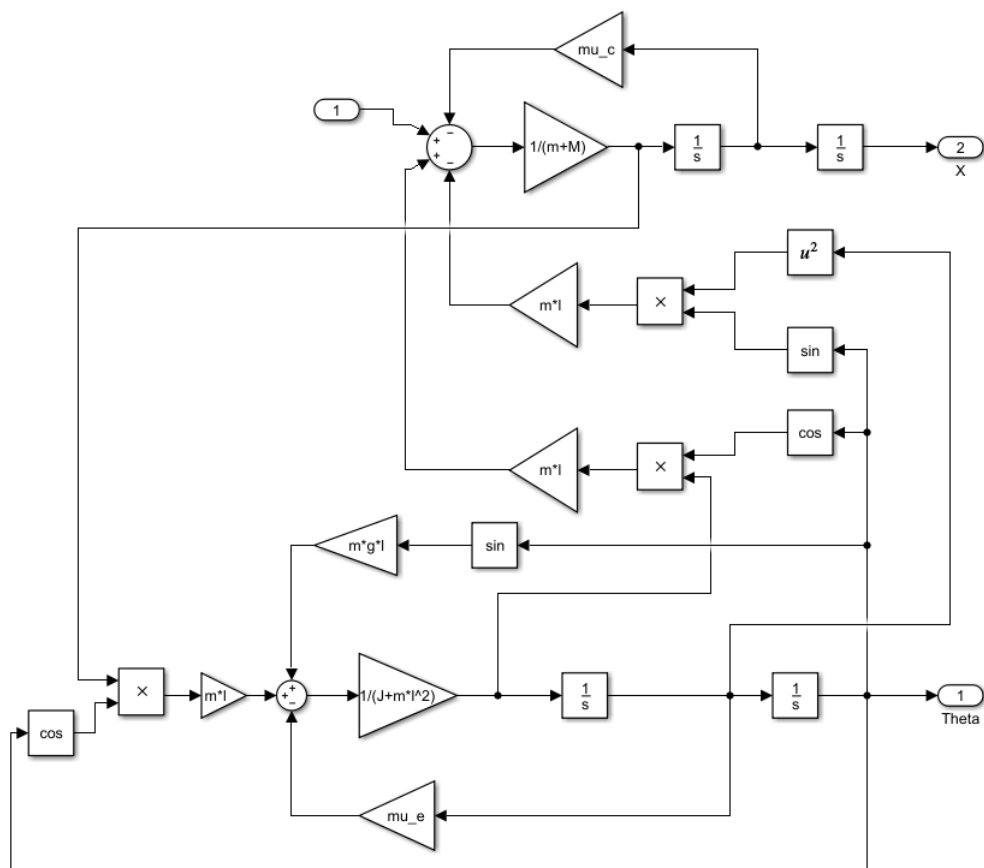
4 IMPLEMENTAÇÃO DO SISTEMA

Este capítulo irá apresentar detalhes do modelo computacional elaborado, da implementação do protótipo de laboratório e do *software* de controle embarcado em um microcontrolador *ATMega2560*.

4.1 MODELAGEM COMPUTACIONAL

Para realizar a simulação e comprovação do correto funcionamento do controle antes de aplicá-lo diretamente na planta física, foi utilizado o ambiente *SIMULINK* do MatLab para construir o modelo da planta física. Utilizando blocos de operações matemáticas construiu-se o modelo não linear real da planta utilizando as equações diferenciais (19) e (20) que descrevem o sistema.

Figura 12 – Modelo da planta não linear.



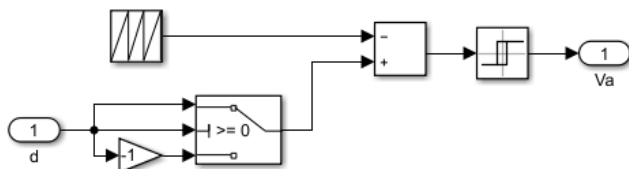
Com este modelo já é possível o teste de controladores na planta utilizando apenas as equações a diferenças. Afim de obter resultados de simulação mais condizentes com a planta física, optou-se por desenvolver também um modelo do microcontrolador, circuito de acionamento e componentes que realizam a aquisição de valores como ângulo e posição.

Para representar um microcontrolador é preciso que o modelo tenha na sua entrada o con-

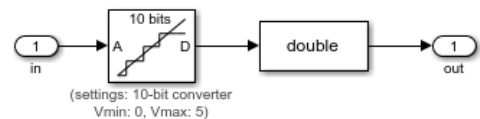
versor A/D, um bloco interno onde está a programação e na sua saída um sinal PWM. Ao se fazer isso, acaba-se tendo uma estrutura *software-in-the-loop* pois o *software* dentro do modelo é quase idêntico ao *software* embarcado no microcontrolador real. O bloco PWM e bloco de conversor A/D são mostrados na figura 13.

Figura 13 – PWM e conversor A/D.

(a) PWM.



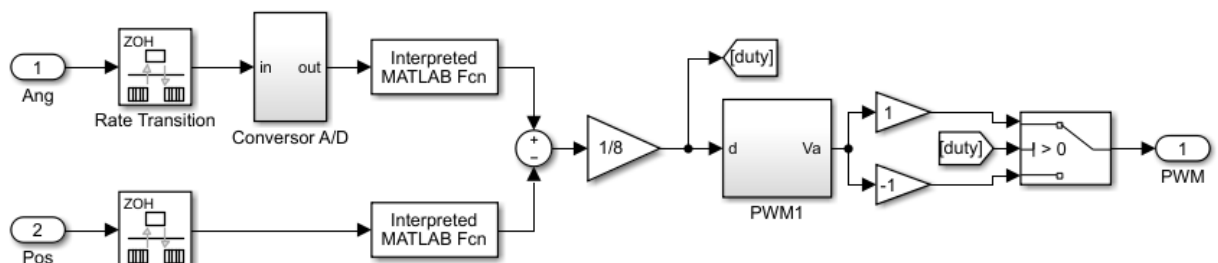
(b) Conversor A/D.



Valores de *duty cycle* negativos devem provocar rotação no sentido oposto. Porém, sabe-se que valores negativos provocam a saturação do sinal PWM em zero. O que o algoritmo no microcontrolador faz nesse caso é detectar valores negativos de *duty cycle*, gerar sinais que provocam a inversão da rotação e inverter o sinal do valor do *duty cycle*, tornando-o positivo. Com isso, consegue-se a rotação no sentido oposto.

Os blocos *Interpreted Matlab Function* permitem inserir código de programação, como em um microcontrolador real e são utilizados para o mapeamento reverso e para realizar os cálculos de controle, também faz-se necessária a aplicação de blocos chamados *rate transition*, que são usados na interface entre os domínios de tempo contínuo e de tempo discreto. O que se faz é especificar a natureza do sinal entregue na saída do bloco. Ao se especificar um período de amostragem, o sinal na entrada será convertido em sinal discreto amostrado de acordo com esse período de amostragem e colocado na saída. No caso deste trabalho, o período de amostragem adotado é $T_s = 10$ ms. Da mesma forma, se não for especificada nenhuma amostragem, a saída do bloco será um sinal contínuo. Assim o modelo completo do microcontrolador pode ser visto na figura 14.

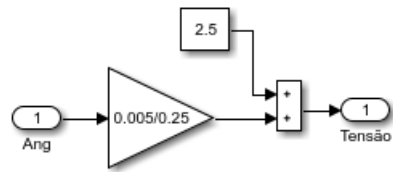
Figura 14 – Modelo do microcontrolador.



O bloco de acionamento eleva a faixa de tensão de saída do PWM de 0V a 5V para uma faixa de tensão entre 0V e 18V, esta tensão é então convertida em força que é aplicada como entrada no sistema. Os blocos dos componentes apenas fazem a conversão de ângulo para tensão (potenciômetro) e de distância para pulsos (*encoder*).

Figura 15 – Blocos do potenciômetro, encoder e acionamento.

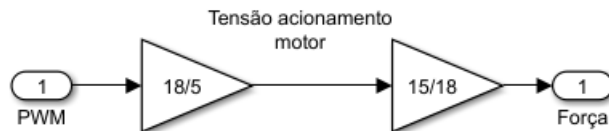
(a) Potenciômetro.



(b) Encoder.

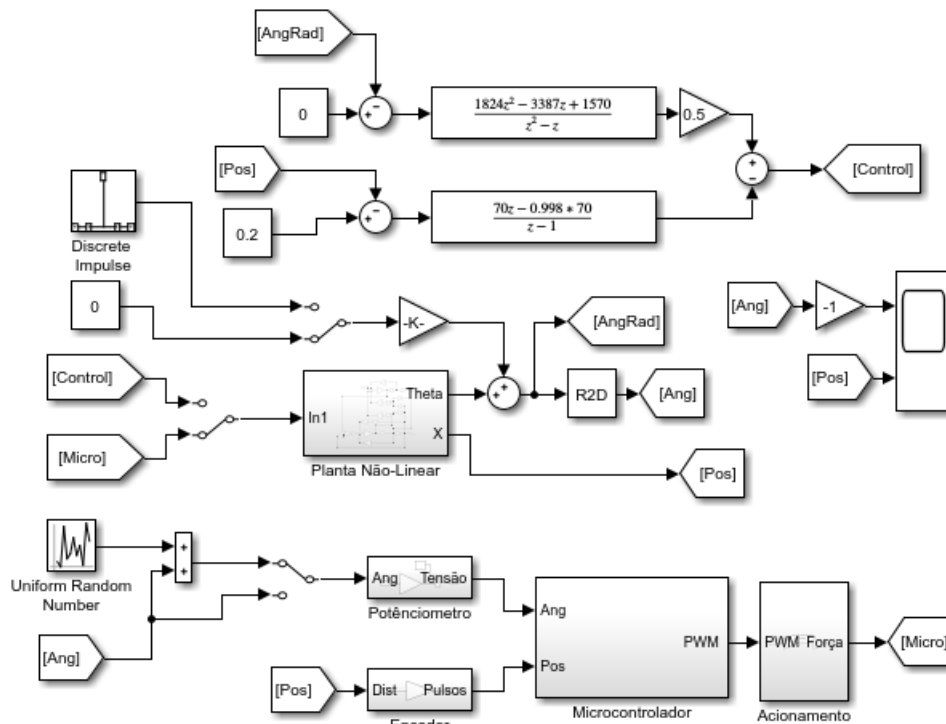


(c) Acionamento.



Assim, o modelo computacional completo da planta é vista na figura 16. O bloco *Uniform Random Number* pode ser somado ou não à referência do ângulo aplicado ao microcontrolador, este bloco soma valores randômicos de -0.5° a 0.5° ao ângulo medido reproduzindo o ruído presente na leitura do ângulo do protótipo.

Figura 16 – Modelo computacional completo.



4.2 ESTRUTURA DA PLANTA FÍSICA

A planta física consiste em uma haste presa a um carro que percorre um trilho, este é movido para frente e para trás sobre o trilho por meio de um motor de corrente contínua. Para a montagem do trilho utilizou-se perfis de alumínio 20x20cm em *v-slot*, com comprimento de um metro, sendo dois destes perfis fixados em paralelo. O carro e as peças para fixação do motor, microcontrolador, *driver* de acionamento e polias foram confeccionadas utilizando uma impressora 3D.

Para a movimentação do carro utilizou-se um motor de corrente contínua de 6V e 210rpm, acoplado a uma polia GT2 de 60 dentes que traciona uma correia GT2 de 6mm fixada ao carro de modo a poder movimentá-lo sobre o trilho. Após alguns testes verificou-se que a velocidade era demasiadamente baixa para o alcance do equilíbrio do pêndulo, basicamente porque o motor estava sendo acionado a maior parte do tempo com tensão muito abaixo da nominal, necessitando de uma velocidade maior em casos de distúrbio na haste onde o ângulo alcança maiores valores. O motor passou a ser alimentado com uma tensão de 18V, obtendo assim uma maior velocidade.

Este motor é acionado pelo *driver* L298N, que pode acionar conjuntamente até dois motores de corrente contínua ou dois motores de passo de 5V a 35V. Este *driver* possui uma queda de tensão na saturação elevada nos transistores responsáveis pelo controle da direção de rotação dos motores. Ao alimentar o *driver* com uma tensão de 5V verificou-se uma queda de tensão de 2,5V, sendo necessário considerar esta queda de tensão ao regular a fonte de alimentação.

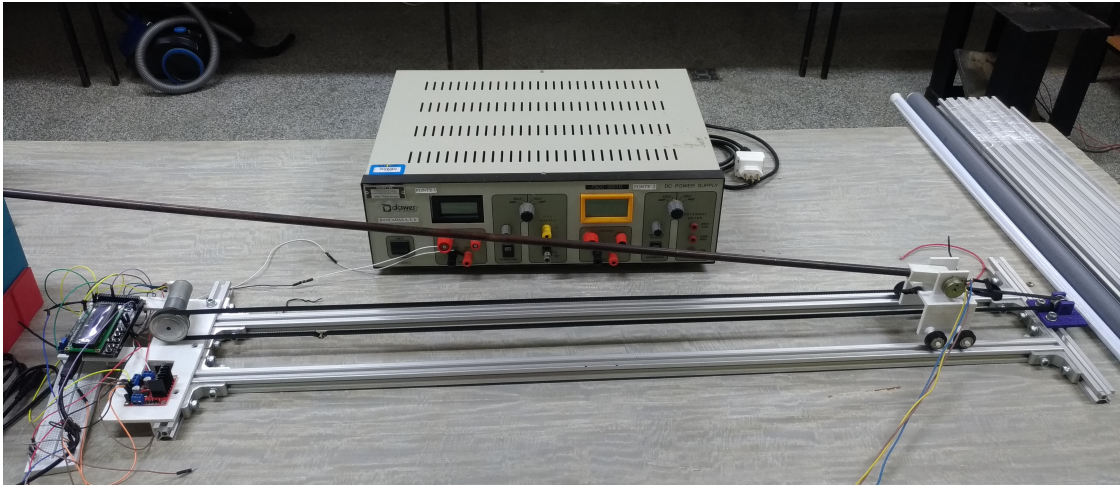
O *driver* apresenta dois pinos responsáveis pela determinação da rotação do motor e um pino *enable* que deve estar em nível lógico alto para o acionamento do motor. Neste pino é conectado o sinal PWM gerado pelo microcontrolador responsável pelo controle da velocidade.

Para a implementação do controle foi utilizada uma placa ARDUINO MEGA com o processador *ATmega2560*. Este opera com uma frequência de *clock* de 16MHz, possui 54 pinos digitais de entrada e saída, onde 15 destes podem ser utilizados com saídas PWM, e ainda 16 entradas analógicas. Juntamente com o microcontrolador, um *shield* de um *display* LCD com seis botões foi utilizado, onde foi criada uma pequena IHM (Interface Homem-Máquina) responsável por mostrar o ângulo atual ou a posição do carro em relação a posição inicial, definir a posição atual do carro como a posição atual e habilitar ou desabilitar o controle.

O ângulo do pêndulo é medido por meio de um potenciômetro linear de 100k Ω conectado a uma entrada analógica do microcontrolador, onde é realizada a conversão A/D da tensão deste potenciômetro, sendo a tensão convertida para ângulo pelo microcontrolador, onde cada nível de quantização do conversor corresponde a 0,25°. A posição do carro é adquirida por meio de um *encoder* incremental já acoplado ao motor, que possui 360 pulsos por volta. Com o número de pulsos contados e sabendo que a cada pulso o carro se movimenta 0,36mm é possível calcular a posição do carro.

O protótipo construído pode ser visto na figura 17.

Figura 17 – Protótipo do pêndulo invertido.



4.3 DESENVOLVIMENTO DO SOFTWARE DE CONTROLE

Primeiramente para o *software* de controle foi configurada uma saída PWM utilizando a função *fast PWM* do processador *ATmega2560* configurado no *Timer 0*, gerando um sinal PWM na frequência de 1kHz.

Para a configuração deste sinal é necessária a configuração de *bits* em dois registradores. No registrador *TCCR0A* devem ser setados os *bits* *COM0A1* e *COM0B1* que fazem com que o sinal vá a zero quando chega ao valor de referência e os *bits* *WGM01* e *WGM00* para que seja configurado no modo *fast PWM*. No registrador *TCCR0B* é necessário setar os *bits* *CS01* e *CS00*, esses são responsáveis por selecionar o *prescaler*, que altera a frequência do sinal PWM.

Na forma tradicional, uma conversão A/D é disparada por uma instrução *analogRead()* que precisa aguardar a conversão terminar para então fazer a leitura. Isso causa um atraso considerável. Para evitar isso, decidiu-se colocar o conversor em modo de conversão contínua, onde a próxima conversão é iniciada automaticamente após o término da anterior, continuamente. O problema é que as bibliotecas do Arduino não permitem o acesso a esse recurso. Foi necessário modificá-las criando uma forma diferente de configuração do módulo A/D. Com isso, a função *analogRead* modificada faz apenas a leitura do registrador onde está armazenado o valor da última conversão.

Para configurar a conversão analógico-digital em modo contínuo devem ser configurados dois registradores, o registrador *ADCSRA* deve ter os *bits* *ADEN* (habilita a conversão), *ADSC* (inicia a conversão), *ADATE* (habilita o *trigger* automático), *ADSP0*, *ADSP1* e *ADSP2* (*prescaler*) setados, e o registrador *ADMUX* deve ter os *bits* *REFS0* (seleciona a tensão de referência a ser utilizada, utiliza *Vcc* como referência), *MUX2* e *MUX1* (seleciona a porta analógica onde deve ser realizada a conversão) também devem ser setados.

O *shield* utilizado para a interface IHM possui 6 botões, todos conectados à mesma entrada analógica, sendo identificado qual foi o botão pressionado pela tensão aplicada a esta entrada. Desta forma para realizar a leitura dos botões é necessário alterar a porta analógica onde está sendo realizada a conversão. Para o funcionamento correto do controle deve-se garantir a alteração desta

porta, sua leitura, a alteração novamente para a porta que recebe o valor do ângulo e a conversão deste valor entre dois intervalos da ação de controle, pois o valor das conversões analógicas são armazenadas no mesmo registrador. Uma pequena biblioteca para a alteração das portas foi criada, facilitando sua alteração caso precise ser feita a alteração entre mais portas.

A pequena central IHM foi criada para poderem ser visualizados em um *display* LCD o valor atual do ângulo do pêndulo ou posição do carro, configurar a posição atual do carro, como a posição inicial, e iniciar ou pausar o controle. A cada 100ms, logo após o término da ação de controle, a porta onde é realizada a conversão analógica digital é alterada, e após garantir que o valor armazenado nos registradores seja o valor da entrada selecionada, a leitura do botão pressionado é feita e executa-se a ação referente a este botão.

Para garantir que o valor armazenado no registrador seja o valor correspondente à porta onde o valor deva ser lido, deve-se verificar após a alteração da porta o término de pelo menos duas conversões. Isto é feito monitorando a *flag* de interrupção que vai a nível lógico baixo quando uma conversão é terminada. Após duas conversões a leitura pode ser feita garantindo a leitura correta da conversão.

Para a seleção das telas que mostram ângulo da haste e posição do carro, utilizou-se uma máquina de estados onde o pressionamento de um botão é definido como transição de um estado ao outro.

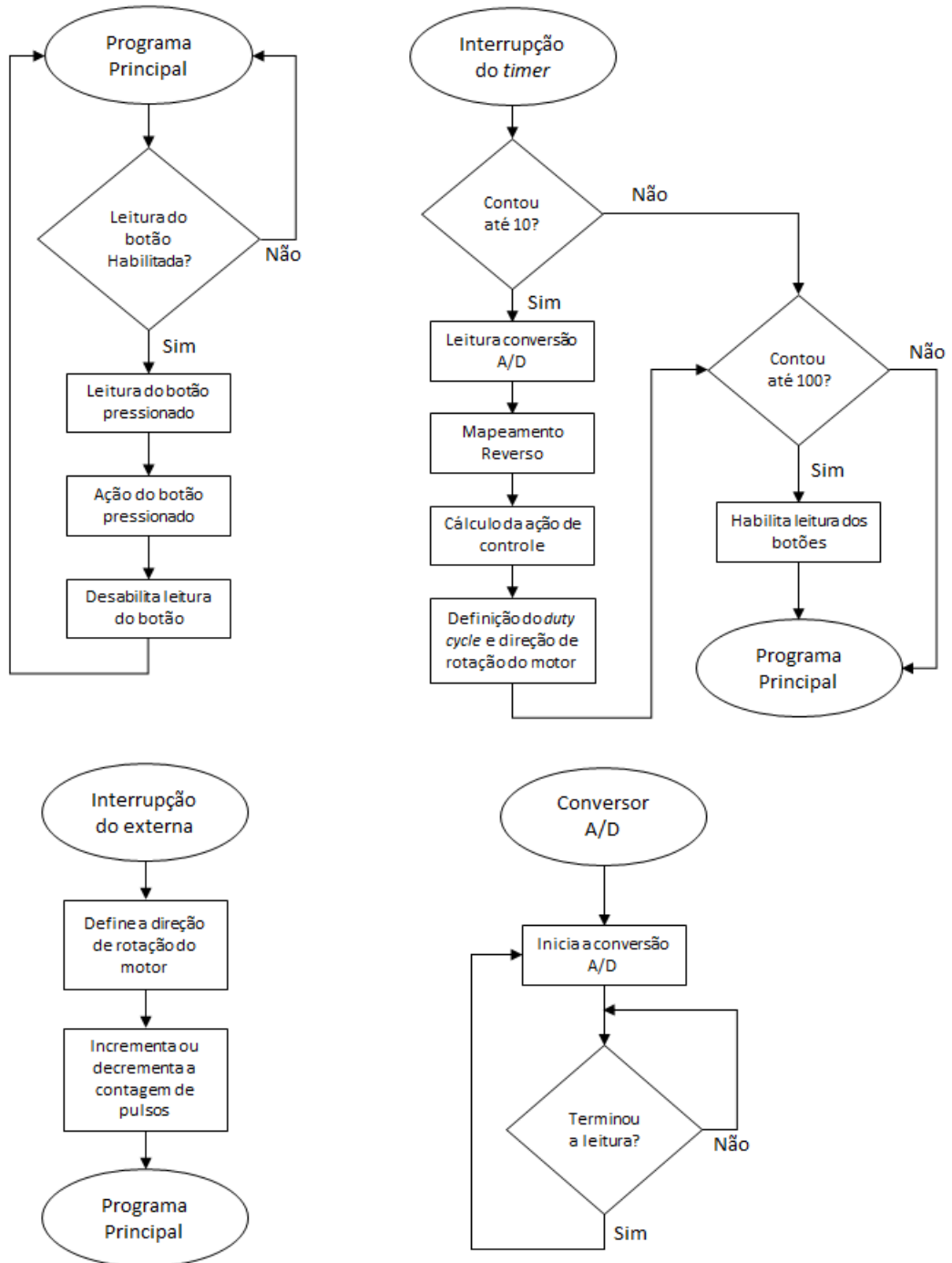
Faz-se também necessária a configuração de interrupção por estouro do *timer* para que o cálculo do controle e aquisição do ângulo e da distância do carro sejam feitos no período de amostragem definido, assim como habilitar a troca de porta da leitura do conversor analógico digital no tempo desejado. Esta interrupção foi configurada para acontecer a cada 1ms, a cada dez interrupções o cálculo de controle é realizado, e a cada 100 interrupções a leitura dos botões é habilitada.

Para a configuração da interrupção por estouro do *timer*, foi utilizado o *Timer 2*. Para isto, devem ser configurados dois registradores: no registrador TCCR2B os *bits* CS21 e CS20 devem ser setados, estes são responsáveis pela alteração do prescaler. Configurados desta forma garante-se o estouro a cada 1ms. No registrador TIMSK2 o *bit* TOIE2 deve ser setado, este habilita a interrupção por estouro do timer.

Uma interrupção por sinal externo foi também configurada para a leitura do encoder. Para isto, o sinal do encoder deve ser conectado a um pino do microcontrolador onde uma alteração possa causar uma interrupção, assim configurou-se esta interrupção para ocorrer em uma borda de descida do sinal de um dos canais do encoder, e então é verificado o nível lógico do outro canal, quando este tem o nível alto, a contagem de pulsos é incrementada, e quando este possui o nível baixo, a contagem é decrementada. O esquema do *software* é mostrado na figura 18.

Este capítulo apresentou a implementação do modelo computacional utilizado para simulações do sistema, do protótipo usado em laboratório e do *software* utilizado para o controle embarcado da planta. O próximo capítulo irá apresentar o resultados obtidos na implementação do controle digital no protótipo.

Figura 18 – Esquemático do *software*.



5 RESULTADOS EXPERIMENTAIS

Este capítulo irá apresentar os resultados obtidos pela aplicação do controle no protótipo construído, bem como sua comparação com as simulações realizadas no modelo computacional.

5.1 MALHA DE EQUILÍBRIO

A figura 19 mostra os resultados obtidos na simulação utilizando apenas a malha de equilíbrio. Um distúrbio foi aplicado ao ângulo de modo a inverter a direção em que o carro se move, isto é necessário no protótipo para que o carro não bata no final do trilho, assim o mesmo foi aplicado na simulação para comparar os comportamentos. O ruído aplicado nas simulações são provenientes do bloco *Uniform Random Number* que soma um valor aleatório entre valores definidos ao ângulo.

Nota-se que há uma diferença entre a aplicação do controle diretamente pela sua equação e a aplicação do controle com o modelo do microcontrolador e demais componentes do protótipo. Isto se deve ao fato do ângulo só poder ser medido em incrementos de 0.25° , fazendo com que uma pequena oscilação do ângulo e da posição ocorram.

Como apenas a malha de equilíbrio está sendo implementada, nota-se que o carro continua se movendo em velocidade constante para um dos lados, pois o pêndulo tende a cair e uma força é aplicada constantemente para evitar a queda.

A figura 20 mostra o resultado obtido quando o controle é aplicado ao protótipo.

Nota-se uma grande oscilação no valor do ângulo, isto se deve ao ruído do potenciômetro e a vibração que ocorre na haste devido ao seu tamanho e rigidez, este ruído e vibração influenciam a ação do controle causando a oscilação observada na posição do carro. Além disso, há outros fatores do mundo real que não foram modelados, como a folga entre engrenagens, a existência de uma zona morta, que não foi compensada, e efeitos da inversão da polaridade do sinal PWM. No fim do trilho foi colocado um obstáculo, assim quando o carro chega nesta posição a haste é empurrada para o outro lado, sendo assim invertido o sentido do movimento do carro.

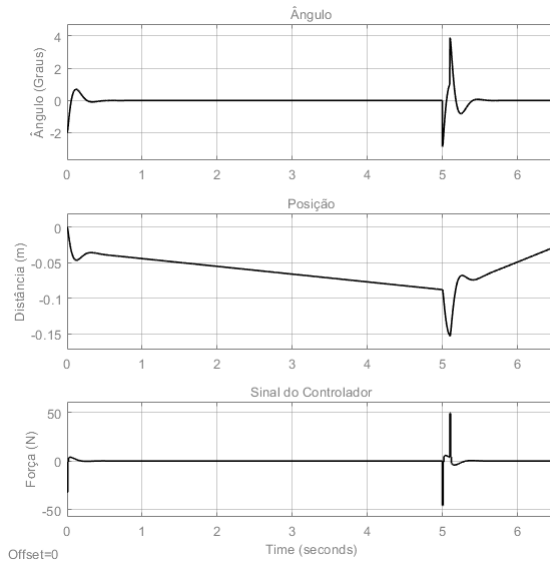
Com o ruído sendo aplicado na simulação, o comportamento se assemelha mais ao comportamento observado no protótipo, mesmo com um filtro aplicado na leitura do potenciômetro reduzindo bastante o ruído presente no mesmo, a vibração presente na haste ainda ocasiona este comportamento.

5.2 MALHA DE POSIÇÃO

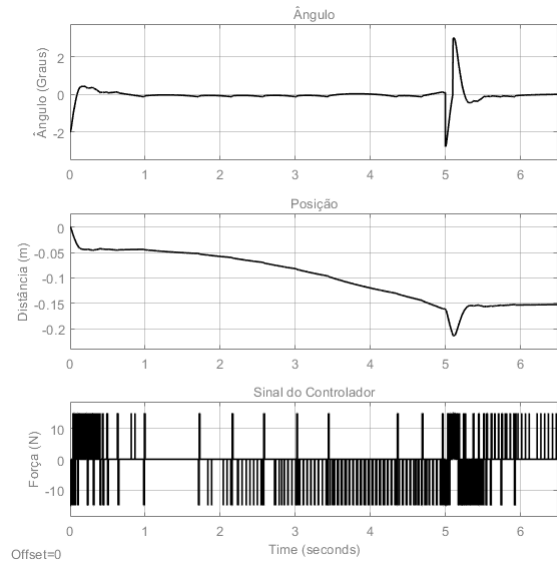
Ao aplicar a malha de posição junto à de equilíbrio ao protótipo, notou-se que o mesmo não estava buscando a posição de referência, então a malha de posição foi também testada isoladamente. A figura 21 mostra os resultados da simulação apenas da malha de posição, o carro inicia na posição zero e busca a referência em 0.2m, ficando oscilando em torno deste ponto. Nota-se que ao aplicar

Figura 19 – Comparação dos resultados de simulação da malha de equilíbrio.

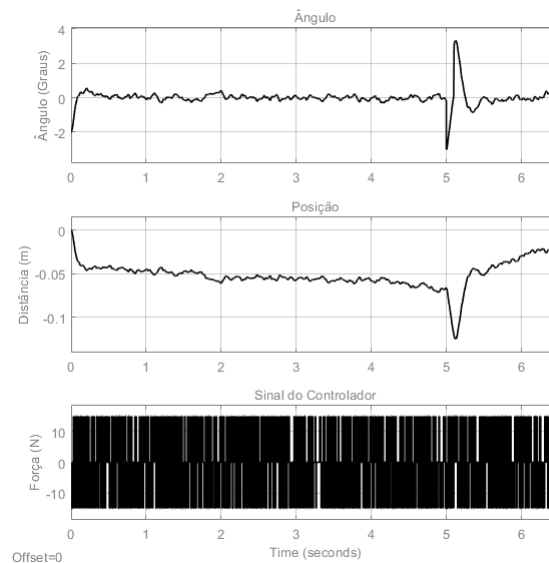
(a) Usando equação a diferenças.



(b) Usando microcontrolador (sem ruído).



(c) Usando microcontrolador (com ruído).



somente esta malha, o sistema não é estável, isto era esperado pois há um polo fora do círculo unitário.

Ao ser aplicada somente a malha de posição ao protótipo, constatou-se que o valor da posição estava invertida em relação à simulação. Assim o sinal desta malha foi somado ao da malha de equilíbrio em vez de ser subtraído como na simulação. Assim a figura 22 mostra o comportamento do protótipo apenas com a malha de posição aplicada após a correção do sinal, nota-se que, apesar da oscilação, o carro busca a posição de referência.

Pode-se perceber que no protótipo o sistema fica mais próximo da estabilidade, isto se dá pelo fato de na simulação, ao inverter o sentido da força, o carro desacelera devagar e então é acelerado na direção oposta. No protótipo quando a direção da força é invertida, o motor inverte seu sentido de giro quase que instantaneamente.

Figura 20 – Resultado da malha de equilíbrio aplicada ao protótipo.

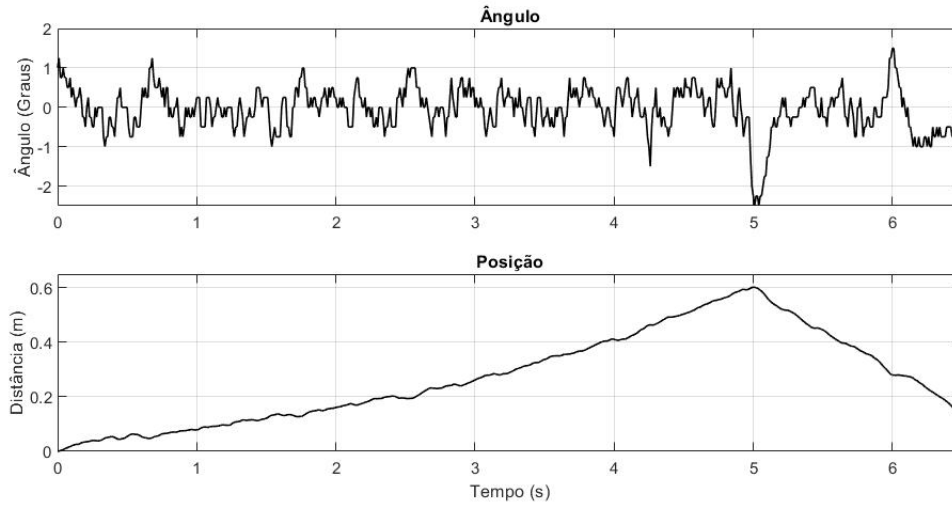


Figura 21 – Comparação dos resultados de simulação da malha de posição.

(a) Usando equação a diferenças. (b) Usando microcontrolador.

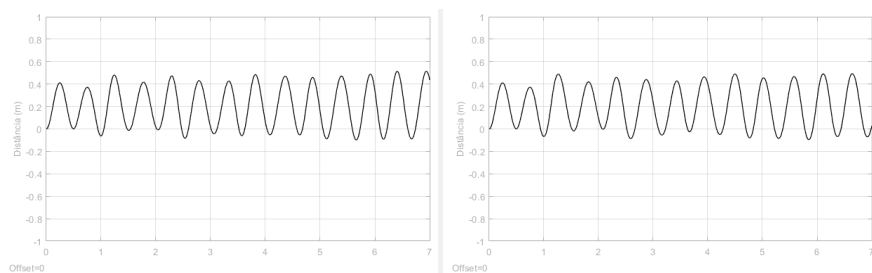
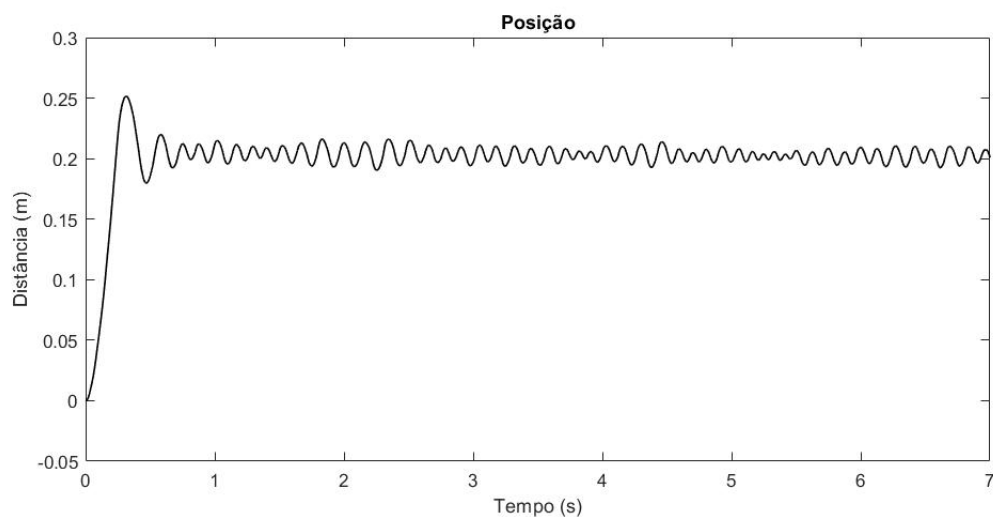


Figura 22 – Resultado da malha de posição aplicada ao protótipo.

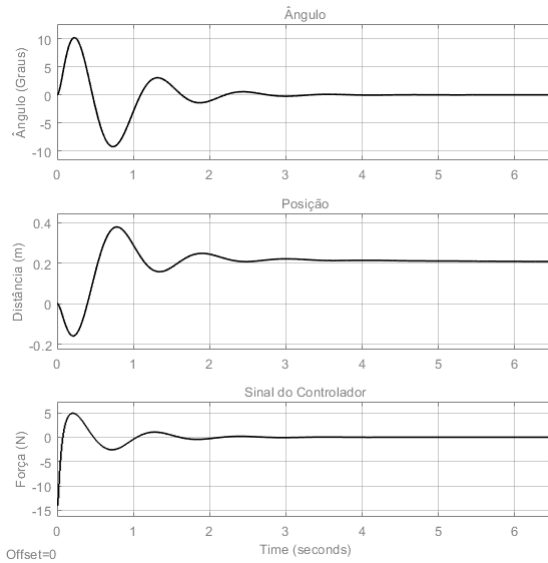


5.3 MALHA DUPLA

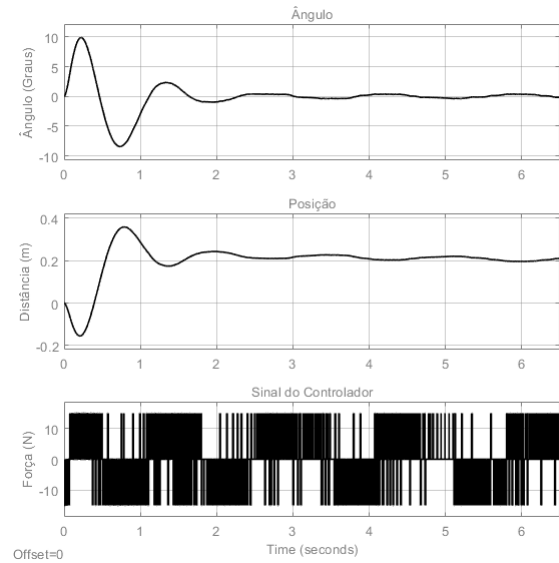
A figura 23 mostra o resultado da simulação aplicada no modelo computacional. Assim como na malha de equilíbrio, a implementação por meio do modelo do microcontrolador e componentes apresenta uma diferença na velocidade de resposta que ocorre novamente pela limitação na força de entrada aplicada ao sistema. Um degrau de 0.2 foi aplicado na referência da posição.

Figura 23 – Comparação dos resultados de simulação com dupla malha.

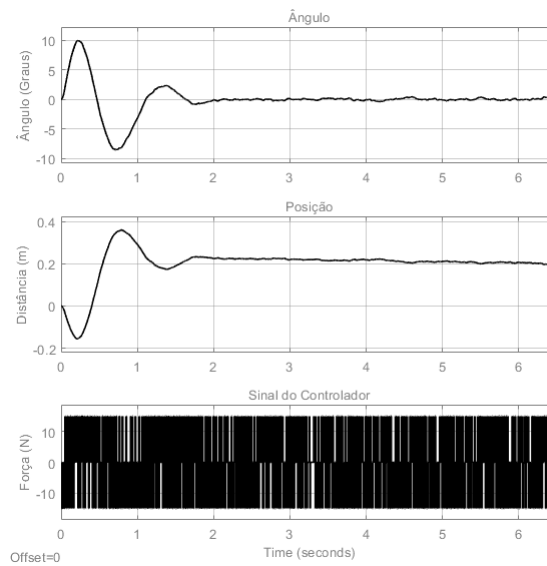
(a) Usando equações a diferenças.



(b) Usando microcontrolador (sem ruído).



(c) Usando microcontrolador (com ruído).



O comportamento obtido ao aplicar as duas malhas ao protótipo pode ser visto na figura 24. Com a malha dupla aplicada ao protótipo o mesmo se comporta de maneira parecida com a simulação, com o ruído presente na aquisição do valor do ângulo e a vibração que ocorre na haste, sendo a malha de equilíbrio mais rápida, a posição desejada nem sempre é alcançada, podendo ter um pequeno erro.

A figura 25 mostra o comportamento ao se aplicar um distúrbio na haste, tentando derrubá-la, mas pode-se notar que o carro sempre busca a posição de referência que neste caso é a posição de zero metros.

Este capítulo apresentou os resultados obtidos com a implementação do controle no protótipo e modelo computacional do pêndulo invertido. No próximo capítulo serão apresentadas as conclusões do trabalho.

Figura 24 – Resultado de dupla malha aplicada ao protótipo.

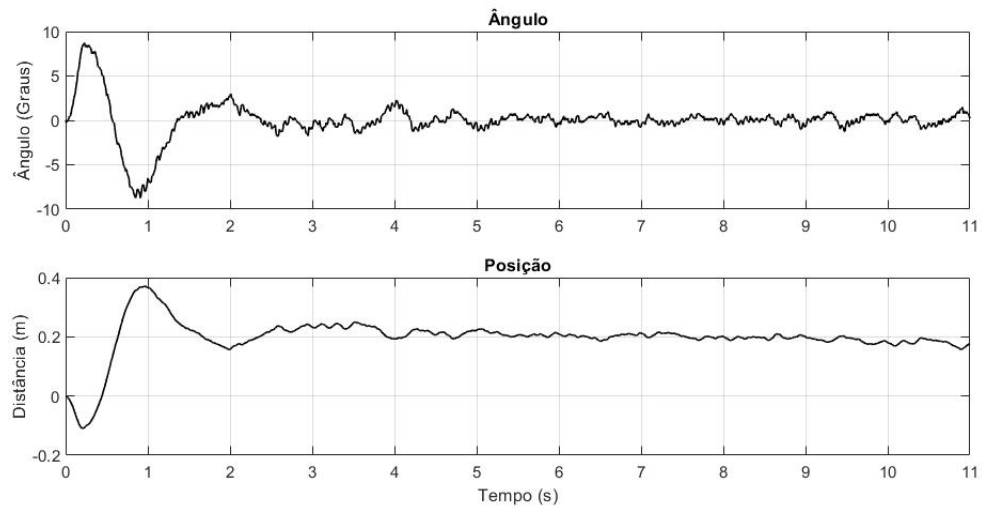
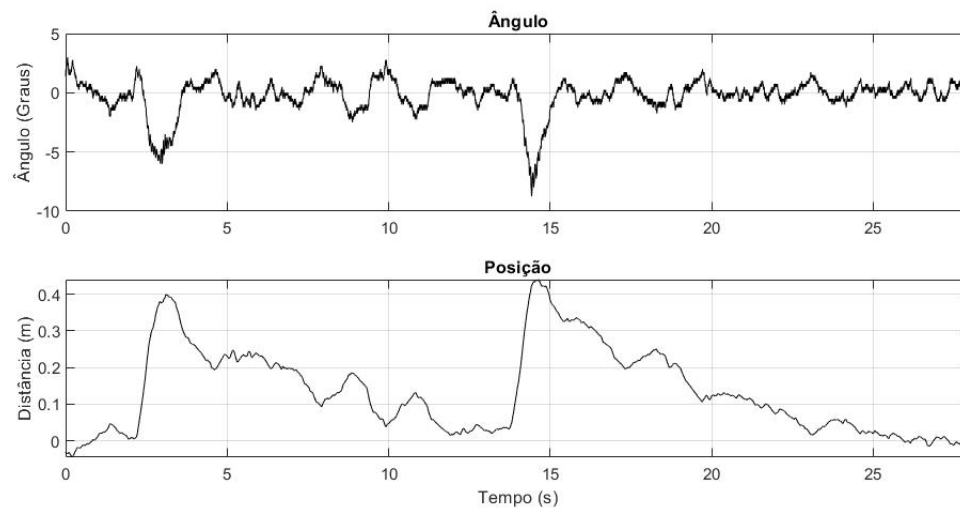


Figura 25 – Distúrbio aplicado à haste com dupla malha.



6 CONCLUSÕES

Este trabalho apresentou o desenvolvimento do modelo matemático que descreve o comportamento dinâmico de um pêndulo invertido, assim como sua linearização e funções de transferência contínuas e discretas. Posteriormente apresentou a estratégia de controle PID de duas malhas para controlar o equilíbrio e posição do pêndulo. O projeto dos controles foram realizados utilizando a ferramenta interativa *SISOTOOL* do *software MatLab*.

Um modelo computacional completo, descrevendo o modelo não linear do pêndulo, modelo do microcontrolador, circuito de acionamento e componentes utilizados para a aquisição de dados de realimentação foi desenvolvido para a simulação e validação do comportamento do controle antes da implementação no protótipo.

Um protótipo foi construído em laboratório para a aplicação do controle utilizando perfis de alumínio e peças confeccionadas em impressora 3D. As peças confeccionadas apresentaram diversos problemas estruturais, assim foi necessário rever o projeto das mesmas.

O motor utilizado para a movimentação do carro apresenta uma baixa velocidade usando sua alimentação nominal (6V) fazendo com que o protótipo não consiga trazer o ângulo novamente a zero mesmo com ângulos pequenos. Devido a dificuldade de se encontrar um motor com maior velocidade e torque necessário para movimentar a plataforma, o mesmo é alimentado por uma tensão acima da nominal, e uma polia GT2 de 60 dentes foi acoplada ao seu eixo para aumentar a velocidade.

Inicialmente apenas uma fonte de alimentação deveria ser utilizada tanto para o motor como para o microcontrolador, sendo que a tensão para o microcontrolador seria reduzida para 5V por meio do regulador de tensão presente no driver de acionamento do motor, porém quando o motor era acionado havia uma grande variação nesta tensão, fazendo com que o ângulo lido pelo potenciômetro tivesse uma grande variação. Como o motor está constantemente alterando seu sentido de giro, isto apresentava um grande problema para o controle. Desta forma o motor e o microcontrolador foram alimentados por fontes distintas.

Utilizou-se como haste um cano de cobre oco, entretanto esta apresenta uma grande vibração, fazendo com que o controle não funcione de forma ideal. Para resolver este problema, a haste deverá ser trocada para uma haste mais rígida ou de menor comprimento.

O pêndulo também apresenta um certo movimento no eixo y , isto ocorre pelo fato do carro estar sobre o trilho. Será preciso fazer um novo projeto para o carro de modo com que este também possua rodas nas laterais do trilho, fazendo com que este fique totalmente fixo eliminando o movimento no sentido indesejado.

Este trabalho trouxe um grande aprendizado, pois além de envolver um grande número de conhecimentos obtidos em disciplinas do curso, exigiu o desenvolvimento de soluções técnicas criativas aos problemas existentes no protótipo físico e no modelo computacional.

A elaboração do modelo computacional também se põe como um desafio, pois plantas do mundo real normalmente são como "caixas-pretas" que precisam ser analisadas e testadas para que

se consiga determinar um modelo que represente seu comportamento. Esse processo nem sempre é fácil, pois alguns parâmetros da planta são difíceis de se obter.

Além disso, o modelo também é importante para fins de comparação com a planta física, fazendo o elo de ligação entre a teoria e a prática aplicadas a um sistema físico real.

Espera-se que a planta construída seja útil para as disciplinas do curso envolvidas com controle, bem como para pesquisas que busquem aplicar outras técnicas de controle.

REFERÊNCIAS

- BALOH, Michael; PARENT, Michael. Modeling and model verification of an intelligent self-balancing two-wheeled vehicle for an autonomous urban transportation system. In: **The conference on computational intelligence, robotics, and autonomous systems**. [S.l.: s.n.], 2003. p. 1–7. Citado na página 10.
- CHANCHAREON, R.; SANGVERAPHUNSIRI, V.; CHANTRANUWATHANA, S. Tracking control of an inverted pendulum using computed feedback linearization technique. In: **2006 IEEE Conference on Robotics, Automation and Mechatronics**. [S.l.: s.n.], 2006. p. 1–6. ISSN 2158-2181. Citado na página 10.
- CHO, Hyun-Taek; JUNG, Seul. Neural network position tracking control of an inverted pendulum an x-y table robot. In: **Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)**. [S.l.: s.n.], 2003. v. 2, p. 1210–1215 vol.2. Citado na página 10.
- FILHO, Moisés Duarte; CORTES, Jacqueline Magalhães Rangel; CARVALHO, Adelson Siqueira; JUNIOR, Geraldo Galdino de Paula. Controle fuzzy para posicionamento de um pêndulo invertido. **X Simpósio de Excelência em Gestão e Tecnologia–SEGeT, Resende–RJ**, 2013. Citado na página 10.
- GAUTAM, Pooja. System identification of nonlinear inverted pendulum using artificial neural network. In: . [S.l.: s.n.], 2016. p. 1–5. Citado na página 17.
- HUANG, Chung-Neng. The development of self-balancing controller for one-wheeled vehicles. **Engineering**, Scientific Research Publishing, v. 2, n. 04, p. 212, 2010. Citado na página 10.
- KUMAR, E Vinodh; JEROME, Jovitha. Robust lqr controller design for stabilizing and trajectory tracking of inverted pendulum. **Procedia Engineering**, Elsevier, v. 64, p. 169–178, 2013. Citado na página 10.
- NAIK, Manjish; COCHRAN, Douglas. System identification of an inverted pendulum on a cart using compressed sensing. **report, Arizona State University**, 2012. Citado na página 17.
- NAWAWI, Sophan Wahyudi; AHMAD, Mohamad Noh; OSMAN, Johari Halim Shah. Real-time control of a two-wheeled inverted pendulum mobile robot. In: . [S.l.: s.n.], 2008. Citado na página 10.

OGATA, K. **Engenharia de controle moderno**. [S.l.]: Pearson Prentice Hall, 1982. Citado 2 vezes nas páginas 10 e 21.

ÅSTRÖM, K.J.; FURUTA, K. Swinging up a pendulum by energy control. **Automatica**, v. 36, n. 2, p. 287 – 295, 2000. ISSN 0005-1098. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0005109899001405>>. Citado na página 10.

VENDRAMINI, Gabriel; SILVA, Paulo Sérgio da. Controle de um pêndulo invertido sobre uma plataforma móvel utilizando PID e MFAC (model-free adaptive control). In: **Proceeding of the 9th Brazilian Conference on Dynamics, Control and Their Applications (DINCON'10)**. [S.l.: s.n.], 2010. p. 405–414. Citado 2 vezes nas páginas 10 e 20.

WANG, Jia-Jun. Simulation studies of inverted pendulum based on pid controllers. **Simulation Modelling Practice and Theory**, v. 19, n. 1, p. 440 – 449, 2011. ISSN 1569-190X. Modeling and Performance Analysis of Networking and Collaborative Systems. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1569190X10001796>>. Citado 2 vezes nas páginas 10 e 20.