



Movimentação de Cargas Suspensas

Mestrado em Engenharia Mecânica – Produção Industrial

Daniela Sofia Duarte Pedrosa

Leiria, março de 2024



Movimentação de Cargas Suspensas

Mestrado em Engenharia Mecânica – Produção Industrial

Daniela Sofia Duarte Pedrosa

Trabalho de Projeto realizado sob a orientação do Professor Doutor Carlos Fernando Couceiro de Sousa Neves e do Professor Doutor Diogo Pedro Ferreira Nascimento Baptista.

Leiria, março de 2024

Originalidade e Direitos de Autor

O presente relatório de projeto é original, elaborada/o unicamente para este fim, tendo sido devidamente citados todos os autores cujos estudos e publicações contribuíram para a/o elaborar.

Reproduções parciais deste documento serão autorizadas na condição de que seja mencionada a Autora e feita referência ao ciclo de estudos no âmbito do qual o mesmo foi realizado, a saber, Curso de Mestrado em Engenharia Mecânica, no ano letivo 2023/2024, da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria, Portugal, e, bem assim, à data das provas públicas que visaram a avaliação destes trabalhos.

Dedicatória

Para os meus pais.

Agradecimentos

Gostaria de expressar a minha sincera gratidão a todas as pessoas que me apoiaram para a realização deste projeto.

Primeiramente, gostaria de agradecer aos meus orientadores, Professor Doutor Carlos Neves e Professor Doutor Diogo Baptista, pelo conhecimento partilhado, disponibilidade, e pelas suas contribuições e orientações que foram importantes para o desenvolvimento deste projeto.

À Engenheira Fernanda Dinis Ludovino Carvalho e ao Engenheiro Jorge Manuel Pedrosa Domingues pela sua ajuda no progresso deste projeto.

Agradeço à minha família pelo apoio que sempre me deram.

Agradeço aos meus amigos pelos incentivos durante a realização deste projeto.

Por fim, agradeço ao meu namorado, por me acompanhar nesta etapa e por todo o seu apoio.

Resumo

Os equipamentos de transporte e elevação de cargas são amplamente utilizados nos mais diversos setores industriais. Podem ser classificados em vários tipos de acordo com a sua configuração de movimento, nomeadamente ponte rolante, sendo este o mais comum em instalações produtivas, e que será o principal em estudo no presente trabalho. Estes equipamentos requerem um controlo manual rigoroso, para evitar oscilações nas cargas. Tais oscilações que, para além de diminuírem a eficiência e produtividade da tarefa, também representam um risco de segurança. Portanto, surge a motivação do estudo de movimentação de cargas suspensas, com o objetivo de demonstrar a viabilidade de controlar este tipo de equipamento de forma a reduzir o ângulo de oscilação da carga e a posicionar a carga com precisão no local desejado, horizontalmente e verticalmente

Para o presente trabalho foi essencial rever a literatura relacionada com o tema, e estudar os conceitos teóricos relacionados com mecânica e sistemas de controlo. Através do projeto, foram realizadas modificações ao equipamento já existente no Laboratório de Robótica Avançada e Fábricas Inteligentes no Instituto Politécnico de Leiria, para possibilitar a elevação de cargas. Um modelo simplificado do equipamento foi criado e modelado, sendo definido em espaço de estados para implementação de um sistema de controlo, neste caso o controlo de realimentação de estados, aplicando o método de colocação de pólos, e adição de um observador. Este sistema de controlo foi implementado em *software* MATLAB para executar simulações, de modo a prever o comportamento do sistema.

Através das simulações realizadas selecionou-se uma matriz de ganho adequada, conforme prioridades para o controlo do sistema, de modo a realizar testes experimentais. Analisando os resultados obtidos, de forma geral, o tempo de acomodação das três variáveis de saída, foi semelhante entre a simulação e a realidade, apesar das diferenças observadas nos gráficos. Comparando o movimento não controlado com o movimento controlado verificou-se que há uma melhoria no tempo de acomodação do ângulo de oscilação da carga: Enquanto a oscilação converge para 0° nos primeiros 10 segundos no movimento controlado, no movimento sem controlo a oscilação é aproximadamente 8° . De seguida, foi pré-definida uma trajetória para a carga suspensa, dividindo os seus troços numa equação paramétrica, e obtiveram-se resultados semelhantes entre o comportamento simulado e o experimental. Além disso, foi possível limitar o ângulo de oscilação a valores inferiores $0,6^\circ$. No último teste, foi considerado que o modelo do sistema era linear por troços, ou seja, o modelo do sistema varia conforme a medição do comprimento do cabo. Concluindo-se que esse modelo está mais próximo do modelo não linear do sistema, e verificou-se que, quando há variação do comprimento do cabo, o controlo com este modelo possui melhores resultados, sendo que o ângulo de oscilação teve uma amplitude entre $[-1,1^\circ; 1^\circ]$.

Pelos resultados obtidos, observou-se a diminuição do ângulo de oscilação da carga, e controlo da posição do carro e comprimento do cabo. Assim, confirma-se a possibilidade de controlo de equipamentos de transporte e elevação de cargas.

Palavras-chave: Ponte rolante, Carga suspensa, Sistema de controlo, Realimentação de estado, Colocação de pólos, Observador

Abstract

Hoisting and load transportation equipment is widely used across various industrial sectors. According to their movement configuration, they can be classified into various types, such as overhead cranes, which are the most common in production facilities and will be the focus of this study. However, this equipment requires meticulous manual control to prevent load oscillations. Such oscillations not only reduce the efficiency and productivity of tasks but also represent a safety risk. Hence, investigating suspended load handling becomes imperative, aiming to demonstrate the feasibility of controlling this equipment to minimize load oscillation angles and accurately position loads at desired locations, both horizontally and vertically.

For this project, a comprehensive review was conducted, delving into theoretical concepts encompassing mechanics and control systems. Through the project, modifications were made to existing equipment at the Advanced Robotics and Intelligent Factories Laboratory at the Polytechnic Institute of Leiria, to enable load lifting capabilities. A simplified model of the equipment was developed and formulated, with its dynamics defined in state-space form, to implement a control system using state feedback control and the pole placement method, with the addition of an observer. This control system was implemented using MATLAB software, enabling simulation runs to predict the system's behavior accurately.

Through the simulations, an appropriate gain matrix was selected based on system control priorities to conduct experimental tests. Overall, analyzing the results obtained, the settling time of the three output variables was generally similar between simulation and reality, despite observed differences in the graphs. There was an improvement in the settling time for the load's oscillation angle when comparing uncontrolled movement to controlled movement. While the oscillation converges to 0° within the first 10 seconds in controlled movement, it remains approximately 8° in uncontrolled movement. Subsequently, a predefined trajectory was set for the suspended load, dividing its sections into a parametric equation, yielding similar results between simulation and experiment. Additionally, the oscillation angle was successfully limited to values below 0.6° . In the final test, the system model was considered linear by sections, meaning it varies as the cable length changes. It was concluded that this model closely resembles the nonlinear system model, and control with this model yielded better results when cable length varied, with the oscillation angle ranging between $[-1.1^\circ; 1^\circ]$.

Based on the results obtained, a decrease in the load oscillation angle and control of the trolley position and cable length was observed. Thus, the possibility of controlling transport and load-lifting equipment is confirmed.

Keywords: Overhead crane, Suspended load, Control system, State feedback, Pole placement, Observer

Índice

Originalidade e Direitos de Autor	i
Dedicatória	ii
Agradecimentos.....	iii
Resumo.....	v
<i>Abstract</i>	vii
Índice.....	viii
Lista de Figuras	x
Lista de Tabelas.....	xiii
Lista de Siglas e Acrónimos.....	xiv
1. Introdução	1
1.1. Objetivos.....	1
1.2. Estrutura do Documento	2
2. Estado de Arte de Trabalhos Relacionados.....	3
2.1. Modelos Matemáticos.....	3
2.2. Sistemas de Controlo	4
3. Enquadramento Teórico	8
3.1. Modelação do Sistema.....	8
3.2. Modelação em Espaço de Estados	9
3.3. Sistemas Lineares	10
3.3.1. Controlabilidade e Observabilidade do Sistema.....	12
3.3.2. Pólos e Resposta do Sistema	12
3.4. Sistemas de Controlo	13
3.4.1. Controlador PID	14
3.4.2. Colocação de Pólos.....	15
3.5. Discretização do Sistema	23
3.6. Sistema Linear por Troços	25
4. Desenvolvimento do Projeto	26
4.1. Modelo Matemático do Sistema	26
4.1.1. Função de Lagrange do Sistema.....	27
4.1.2. Equações de Movimento	29
4.2. Modelo em Espaço de Estados	30
4.2.1. Linearização do Modelo.....	31
4.2.2. Comparação dos Modelos Não Linear e Linear	33
4.3. Projeto Mecânico	36
4.3.1. Estudo do Sistema de Elevação de Carga.....	36

4.3.2.	Desenvolvimento do Sistema de Elevação de Carga.....	37
4.4.	<i>Hardware</i> de Controlo	43
4.5.1.	Atuadores e Sensores.....	43
4.5.2.	Placa de Controlo	46
4.5.	<i>Software</i> de Controlo	47
4.5.1.	Controlo do Equipamento e Transmissão de Dados.....	47
4.5.2.	Aquisição e Tratamento de Dados no Computador.....	50
4.6.	Sistemas de Controlo	51
4.6.1.	Arquitetura do Sistema de Controlo	51
4.6.2.	Adaptação do Modelo com Entrada em Velocidade	51
4.6.3.	Controlabilidade do Sistema.....	54
4.6.4.	Observabilidade do Sistema	54
4.6.5.	Pólos do Sistema	55
4.6.6.	Simulações de Sistemas de Controlo.....	56
4.6.7.	Simulações com Trajetória.....	64
5.	Testes Experimentais e Resultados	66
5.1.	Condições de Realização dos Testes.....	66
5.2.	Movimento Controlado.....	68
5.3.	Controlo de Trajetória.....	78
5.4.	Controlo com Modelo por Troços	82
5.5.	Discussão de Resultados	83
6.	Conclusão.....	85
6.1.	Propostas de Trabalho Futuro	86
7.	Referências Bibliográficas	87
	Apêndices.....	90
	Apêndice A – Programa em MATLAB para comparação dos modelos não linear, linear e linear por troços.....	91
	Apêndice B – Programa em MATLAB para simulação dos sistemas de controlo.....	94
	Apêndice C – Diagrama de blocos no <i>Simulink</i> do sistema de controlo para um movimento com uma trajetória pré-definida da carga suspensa	98
	Apêndice D – Programa em MATLAB para determinar as matrizes para comprimento do cabo dividido em troços.....	99
	Apêndice E – Esquema de ligações elétricas	100
	Apêndice F – <i>Grafset</i> de nível 1 do funcionamento do <i>software</i>	101
	Apêndice G – Programa <i>Arduino</i>	102

Lista de Figuras

Figura 2.1 - Sistema de dois pêndulos.....	3
Figura 2.2 - Sistema com cabo flexível.....	4
Figura 2.3 - Sistema com cabo folgado.....	4
Figura 3.1 - Linearização em torno do ponto A. Adaptado de [40]	11
Figura 3.2 - Diagrama de blocos de um sistema de controlo industrial automático. Adaptado de [38].....	13
Figura 3.3 - Controlador PID. Adaptado de [38]	14
Figura 3.4 - Sistema de controlo com realimentação de estados.....	15
Figura 3.5 - Introdução da referência de entrada, <i>xref</i> , no sistema de controlo de realimentação de estados para melhor desempenho do controlo.....	16
Figura 3.6 - Sistema de realimentação de estados com observador de ordem completa e método de controlo colocação de pólos	17
Figura 3.7 - Controlador de realimentação de estados com observador de ordem reduzida e com o método de colocação de pólos.....	22
Figura 3.8 - Digitalização de um sistema contínuo	23
Figura 3.9 - Diferença das entradas e saídas entre sistema contínuo e sistema discreto	23
Figura 3.10 - Relação entre os pólos do plano s e do plano z	24
Figura 4.1 - Modelo simplificado do sistema em estudo	26
Figura 4.2 - Decomposição da velocidade tangencial da carga suspensa	27
Figura 4.3 - Energia potencial do pêndulo	28
Figura 4.4 - Modelo Linear versus Modelo Não Linear para ângulo de oscilação e comprimento do cabo, com carro parado e comprimento do cabo constante.....	34
Figura 4.5 - Modelo Linear versus Modelo Não Linear para ângulo de oscilação e comprimento do cabo, com carro parado e comprimento do cabo varia perto em torno de 1 metro	34
Figura 4.6 - Modelo Linear versus Modelo Não Linear versus Modelo Linear com l dividido em troços considerando que comprimento inicial do cabo é igual a 1	35
Figura 4.7 - Projeto do carro na equipamento pré-existente. Fonte: [42].....	36
Figura 4.8 - Carro.....	37
Figura 4.9 - Desenho 3D do projeto mecânico em SolidWorks.....	37
Figura 4.10 - a) Desenho 3D do tambor b) Impressão em plástico do tambor.....	38
Figura 4.11 - a) Desenho 3D suporte do tambor b), c) Resultado do suporte do tambor.....	38
Figura 4.12 - a) Desenho 3D engrenagem 30 dentes b) Impressão da engrenagem 30 dentes ...	39
Figura 4.13 - a) Desenho 3D engrenagem 15 dentes b) Impressão da engrenagem 15 dentes ...	40
Figura 4.14 - a) Desenho 3D conjunto b) Resultado da vista de cima do conjunto	40
Figura 4.15 - Montagem do acoplador	41
Figura 4.16 - Desenho 3D da peça para leitura do ângulo do cabo.....	41
Figura 4.17 - Desenho 3D da peça de fixação 1.....	41
Figura 4.18 - Desenho 3D da peça fixação 2	42
Figura 4.19 - Montagem do encoder	42
Figura 4.20 - Resultado do projeto mecânico desenvolvido	42
Figura 4.21 - Arquitetura do sistema de controlo do equipamento	43
Figura 4.22 - Motor EMG30 para controlo do comprimento do cabo	43
Figura 4.23 - L298.....	44
Figura 4.24 - Montagem do driver L298.....	45
Figura 4.25 - Implementação do sensor de fim de curso para o comprimento do cabo	45
Figura 4.26 - Encoder absoluto AEAT-6012-A06	45
Figura 4.27 - Arduino Mega 2560.....	46
Figura 4.28 - Fluxograma do funcionamento geral do sistema para recolha dados	47

Figura 4.29 - Tipos de movimento programados no Arduino	48
Figura 4.30 - Relação entre o valor medido do ângulo de oscilação de carga pelo encoder (°) e a posição do carro no carril (m)	49
Figura 4.31 - Comparação da velocidade em m/s e a velocidade PWM do motor, na subida e descida da carga	50
Figura 4.32 - Esquema do sistema de controlo	51
Figura 4.33 - Diagrama de Pólos do Sistema	56
Figura 4.34 - Diagrama de blocos do sistema em malha fechada com o controlador PID no Simulink	57
Figura 4.35 - Controlo de realimentação de estados com colocação de pólos, dada uma referência ao sistema	57
Figura 4.36 - Diagrama de blocos do sistema em malha fechada com realimentação de estados e colocação de pólos, considerando sistema regulador com dada referência no Simulink	58
Figura 4.37 - Aplicação do observador a três estados do sistema	58
Figura 4.38 - Esquema do controlo de realimentação de estados com colocação de pólos e observador de ordem completa	59
Figura 4.39 - Diagrama de blocos do sistema em malha fechada com realimentação de estados e colocação de pólos, com observador de ordem completa no Simulink.....	59
Figura 4.40 - Simulações obtidas entre diferentes controladores.....	60
Figura 4.41 - Diagrama de blocos do sistema em malha fechada com realimentação de estados e colocação de pólos, com observador de ordem reduzida no Simulink.....	61
Figura 4.42 - Resultado entre variável estimada e não estimada	61
Figura 4.43 - Comparação e influência de valores do observador	63
Figura 4.44 - Trajetória da carga desejada dividida em troços.....	64
Figura 4.45 - Comparação entre trajetória desejada versus trajetória simulação	65
Figura 5.1 - Simulação e comparação do controlo com observador de ordem completa e com observador de ordem reduzida	66
Figura 5.2 - Resultados experimentais do ângulo de oscilação de carga para controlador com observador de ordem reduzida versus observador de ordem completa.....	67
Figura 5.3 - Resultados experimentais da posição do carro para controlador com observador de ordem reduzida versus observador de ordem completa	67
Figura 5.4 - Resultados experimentais do comprimento do cabo para controlador com observador de ordem reduzida versus observador de ordem completa.....	67
Figura 5.5 - Comparação de simulações do ângulo de oscilação para diferentes pólos.....	69
Figura 5.6 - Comparação de simulações da posição do carro para diferentes pólos	69
Figura 5.7 - Comparação de simulações do comprimento do cabo para diferentes pólos	70
Figura 5.8 - Comparação das variáveis de saída entre simulação e teste experimental para os valores de ganho da simulação 1	71
Figura 5.9 - Comparação dos valores do teste 6 do ângulo de oscilação, entre resultado simulado e experimental	72
Figura 5.10 - Comparação dos valores do teste 6 da posição do carro, entre resultado simulado e experimental.....	72
Figura 5.11 - Comparação dos valores do teste 6 do comprimento do cabo, entre resultado simulado e experimental.....	73
Figura 5.12 - Comparação dos valores do teste 4 do ângulo de oscilação, entre resultado simulado e experimental	73
Figura 5.13 - Comparação dos valores do teste 4 da posição do carro, entre resultado simulado e experimental.....	74
Figura 5.14 - Comparação dos valores do teste 4 do comprimento do cabo, entre resultado simulado e experimental.....	74

Figura 5.15 - Comparação dos valores do teste 9 do ângulo de oscilação, entre resultado simulado e experimental	75
Figura 5.16 - Comparação entre simulação e experimental sem controlo do sistema.....	75
Figura 5.17 - Resultado do ângulo de oscilação de carga, movimento com controlo versus sem controlo	76
Figura 5.18 - Movimento sem controlo.....	76
Figura 5.19 - Resultado da posição do carro, movimento com controlo versus sem controlo....	77
Figura 5.20 - Resultado do comprimento do cabo, movimento com controlo versus sem controlo	77
Figura 5.21 - Comparação de simulações do ângulo experimental para diferentes pólos para a trajetória	78
Figura 5.22 - Comparação de simulações da posição do carro para diferentes pólos para a trajetória	78
Figura 5.23 - Comparação de simulações do comprimento do cabo para diferentes pólos para a trajetória	78
Figura 5.24 - Comparação do teste 1 da trajetória da carga, entre resultado simulado e experimental.....	79
Figura 5.25 - Comparação do teste 1 do ângulo de oscilação, entre resultado simulado e experimental.....	79
Figura 5.26 - Comparação do teste 1 da posição do carro, entre resultado simulado e experimental	80
Figura 5.27 - Comparação do teste 1 do comprimento do cabo, entre resultado simulado e experimental.....	80
Figura 5.28 - Comparação do teste 2 da trajetória da carga, entre resultado simulado e experimental.....	80
Figura 5.29 - Comparação do teste 2 do ângulo de oscilação, entre resultado simulado e experimental.....	81
Figura 5.30 - Comparação do teste 2 da posição do carro, entre resultado simulado e experimental	81
Figura 5.31 - Comparação do teste 2 do comprimento do cabo, entre resultado simulado e experimental.....	81
Figura 5.32 - Resultado experimental da aplicação do controlo considerado modelo linear por troços, aplicando várias referências ao comprimento do cabo	82
Figura 5.33 - Resultado experimental da aplicação do controlo considerado modelo linear em torno do comprimento do cabo de 1 metro, aplicando várias referências ao comprimento do cabo	83
Figura C.1 - Diagrama de blocos do sistema de controlo com realimentação de estados e observador de ordem completa, utilizando o método de colocação de pólos, com referência de uma trajetória pré-definida no Simulink	98
Figura E.1 - Esquema de ligações elétricas.....	100
Figura F.1 - Grafset de nível 1 simplificado do funcionamento do software.....	101

Lista de Tabelas

Tabela 4.1 - Comparação dos modelos através do coeficiente de Correlação de Pearson e valor Euclidiano	35
Tabela 4.2 - Características da engrenagem do veio M8 do tambor	39
Tabela 4.3 - Características da engrenagem do veio roscado M8.....	40
Tabela 4.4 - Especificações do motor EMG30.....	43
Tabela 4.5 - Conexões do motor EMG30.....	44
Tabela 4.6 - Especificações do L298	44
Tabela 4.7 - Especificações do encoder AEAT-6012-A06	45
Tabela 4.8 - Características do Arduino Mega 2560.....	46
Tabela 4.9 - Equações do troço horizontal i.....	64
Tabela 4.10 - Equações do troço vertical i.....	64
Tabela 4.11 - Equações do troço circular i.....	64
Tabela 5.1 - Parâmetros considerados na realização de testes experimentais	66
Tabela 5.2 - Valores dos ganhos para cada variável, para diferentes simulações	68
Tabela 5.3 - Resultados dos valores das variáveis obtidos através de simulações	70
Tabela 5.4 - Teste de simulação escolhido de acordo com a prioridade de controlo	71

Lista de Siglas e Acrónimos

2D	Bidimensional
3D	Tridimensional
EC	Energia Cinética
EP	Energia Potencial
ESTG	Escola Superior de Tecnologia e Gestão
LIT	Linear Invariante no Tempo
LQR	Regulador Linear Quadrático
LRAFI	Laboratório de Robótica Avançada e Fábricas Inteligentes
PID	Proporcional Integral Derivativo
PWM	<i>Pulse Width Modulation</i>
SMC	<i>Sliding Mode Control</i>
ZOH	<i>Zero Order Holder</i>

1. Introdução

Os equipamentos de transporte e elevação são amplamente empregues para transporte de cargas, de diversos pesos e volumes. Atualmente, é um equipamento indispensável em atividades produtivas, sendo vastamente utilizados em vários setores, tais como fábricas, construções, indústrias marítimas, transporte ferroviário, entre outros. Podem ser classificados em diferentes tipos, com base nos graus de liberdade, tais como: pontes/pórticos rolantes, gruas de lança e gruas torre rotativas. [1], [2].

De forma geral, as pontes rolantes são constituídas por um carro que se movimenta horizontalmente sobre um carril e um cabo que prende a carga útil, para a transportar. O cabo pode ser acionado para levantar ou baixar a carga e o carro, ao deslocar-se, move a carga até à posição pretendida. Assim, a principal função de uma ponte rolante é a movimentação de cargas suspensas.

Quando manuseada gera inevitavelmente movimentos pendulares nas cargas, tornando a tarefa de transporte lenta e difícil de controlar manualmente. Uma vez que a carga é suspensa por cabos, a carga pode sofrer oscilações excessivas durante o percurso e, principalmente, no arranque e na paragem. Naturalmente, quanto maior a aceleração do movimento, maior o ângulo de oscilação da carga no cabo. Isto pode levar à diminuição da produtividade por redução do volume de trabalho realizado, visto que o tempo total necessário para o posicionamento da carga se torna maior. Além disso, afeta a precisão e qualidade do posicionamento da carga, podendo tornar-se um risco de segurança ou provocar danos noutros equipamentos ou materiais em zonas adjacentes à trajetória.

Têm surgido publicações, pelo menos desde o ano de 1961, sendo relatadas diversas abordagens de modelação deste equipamento e técnicas de controlo utilizadas, assim como o seu desempenho [1].

Assim, surge o desafio deste projeto: conceber um sistema de controlo para controlar um equipamento de movimentação de cargas, de modo a transportar a carga com reduzida oscilação e de a posicionar no exato local desejado, no mais curto período possível.

Com o estudo e desenvolvimento deste projeto, será possível verificar as vantagens e potencialidades da implementação de métodos de controlo nestes equipamentos.

1.1. Objetivos

O principal objetivo do presente projeto consiste na alteração mecânica do sistema existente no Laboratório de Robótica Avançada e Fábricas Inteligentes da Escola Superior de Tecnologia e Gestão (ESTG) do Politécnico de Leiria, para estudar a movimentação de cargas suspensas, em duas dimensões, testando várias estratégias de controlo para melhorar o desempenho do sistema. Este divide-se em objetivos específicos:

- Investigar e analisar os estudos existentes na área de controlo de movimentação de cargas suspensas, de modo a identificar a metodologia a estudar;
- Alterar o projeto mecânico para possibilitar a movimentação vertical da carga;
- Modelar matematicamente o equipamento desenvolvido;
- Desenvolver um programa de simulação do modelo do equipamento, para estudar e prever o respetivo comportamento;

- Projetar e implementar um sistema de controlo e verificar o efeito, através do programa de simulação, tendo em conta a diminuição da oscilação da carga e posicionamento, horizontal e vertical, da carga no local desejado;
- Conceber um *software* e *hardware* de modo a aplicar o sistema de controlo.
- Realizar testes experimentais e comparar com as simulações realizadas, para validar as previsões da dinâmica do sistema controlado.

1.2. Estrutura do Documento

O presente relatório encontra-se estruturado em 6 capítulos, expostos de forma resumida de seguida.

No capítulo 1 está descrita a introdução do projeto em estudo e seus objetivos.

O capítulo 2 apresenta a literatura científica relacionada com o tema em estudo,

O capítulo 3 diz respeito ao enquadramento teórico dos principais conceitos utilizados para realizar este projeto.

No capítulo 4 está exposto o desenvolvimento do projeto, iniciando pelo desenvolvimento do modelo matemático do sistema, seguindo-se a conceção do projeto mecânico, o desenvolvimento do seu *software* e *hardware* e o estudo e implementação do sistema de controlo.

O capítulo 5 pretende demonstrar os dados obtidos a partir de testes realizados e verificar a influência do sistema de controlo implementado.

No capítulo 6 são apresentadas as conclusões e propostas de trabalhos futuros.

2. Estado de Arte de Trabalhos Relacionados

2.1. Modelos Matemáticos

Através da pesquisa e da análise de diversos artigos sobre sistemas de pontes rolantes, constata-se a existência de uma variedade de modelos matemáticos propostos.

De um modo geral, é possível dividir os esquemas matemáticos em espaço 2D e espaço 3D. A. Khatamianfar e A. V. Savkin [3], H. Le Xuan *et al.* [4] e M. J. Maghsoudi *et al.* [5] são exemplos da utilização de modelos de pontes rolante em 3D. No entanto, são mais comuns na literatura os modelos em espaço 2D pela sua simplicidade. A representação da ponte rolante no espaço 2D é composta, de forma geral, por um carro com movimento horizontal que está sujeito a uma força F , unido a um pêndulo e pelo cabo que suporta a carga. Em operação, o cabo forma um ângulo de oscilação, θ em relação à posição (vertical) de repouso.

Dependendo do tipo de mecanismo, o pêndulo pode variar em configuração e forma. V. Caneiro [6] e L. Tuan *et al.* [7] consideram o cabo como um pêndulo simples, e assumem que o cabo é rígido. Em vez de considerarmos um pêndulo simples, este pode ser visto como um pêndulo duplo. Este é um sistema caracterizado por dois pêndulos (Figura 2.1), do qual um deles está no extremo do outro, e, conseqüentemente, existirá outro ângulo de oscilação. Um exemplo desse tipo é um cabo combinado com um gancho para mover a carga, no entanto, isso não se adequa ao projeto em questão. Estes tipos são utilizados no trabalho de Q. Wu *et al.* [8] e K. Alhazza *et al.* [9].

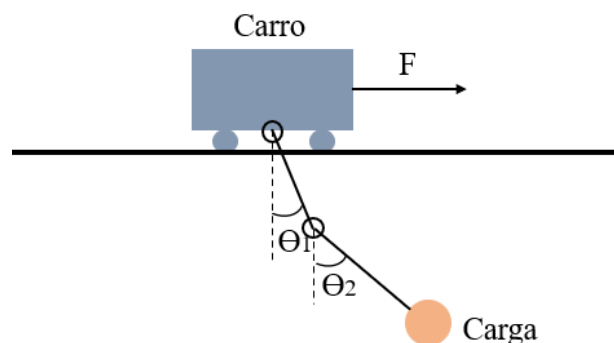


Figura 2.1 - Sistema de dois pêndulos

Entre os vários estudos existentes, há diferenças nas variáveis consideradas no sistema.

S. Zhang *et al.* [10], Q. Zhou *et al.* [11] e H. Lee [12] abordam o comprimento do cabo como constante, considerando duas variáveis: o deslocamento do carro, $x(t)$, e o ângulo de oscilação, $\theta(t)$. No entanto, devido à presença de mecanismos de elevação em pontes rolantes, surgiram estudos que também levam em conta a dinâmica desse mecanismo. L. Tuan *et al.* [7] e L. Ramli *et al.* [13] consideram a variação do comprimento do cabo, $l(t)$, que eleva a carga. Este modelo, é o que mais interessa no presente projeto. Assim, surge os modelos que têm em conta a existência de duas forças a atuar no sistema, fx e fl , que correspondem à força aplicada no carro e à força aplicada no cabo, respetivamente. Por exemplo, os modelos propostos por H. Lee [12], L. Anh Tuan *et al.* [14] e R. Miranda-Colorado [15]. Em contrapartida há estudos, como Q. H. Ngo e K. S. Hong [16] que apenas consideram fx . No caso do estudo de T.W. Yang *et al.* [17] para além de considerarem as variações no comprimento do cabo, também desenvolveram um modelo considerando a flexibilidade do cabo (Figura 2.2), empregando a equação dinâmica de onda para representar o cabo.

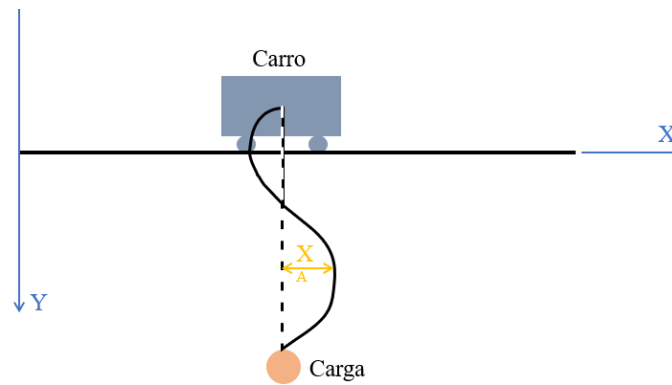


Figura 2.2 - Sistema com cabo flexível

H. V. Pham *et al.* [18] consideraram a elasticidade do cabo. Nestes modelos, os cabos são modelados com dimensões infinitas e deformáveis, sendo denominados modelos de massa distribuída. Já, os modelos de cabo rígido são chamados modelos de massa concentrada.

M. Rasool Mojallizadeh *et al.* [2] descreve o método de cabo folgado (Figura 2.3), o qual aborda situações em que o cabo é excessivamente longo, a massa da carga útil é comparável à do cabo e o carro apresenta grandes oscilações.

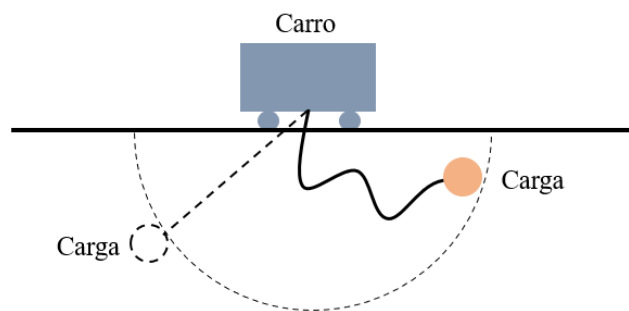


Figura 2.3 - Sistema com cabo folgado

Esse trabalho apresenta ainda modelos de múltiplos pêndulos, que podem ser modelados como cabo único, realizando aproximações.

Na literatura, o modelo matemático de um sistema de ponte rolante, é determinado através da energia cinética e da energia potencial, seguindo-se a aplicação do Princípio de Hamilton, para determinar as equações dinâmicas que representam o sistema, como é demonstrado por T. L. Nguyen *et al.* [19] e K. T. Mohamed [20]. Tendo em conta o número de graus de liberdade, determinam-se as coordenadas e forças generalizadas. Após isto, é desenvolvida a equação de Euler-Lagrange para determinar as equações do movimento para cada coordenada generalizada.

2.2. Sistemas de Controle

Nos últimos anos, vários esquemas de controle para modelos de pontes rolantes têm sido propostos por diferentes autores. Na literatura analisada é utilizado o controle em malha fechada, que recorre a sensores para medir as variáveis de saída do sistema e, de acordo com os objetivos de controle, é determinada a ação de controle mais indicada para os atingir.

Um método utilizado em sistemas de controle é a realimentação de estado, método que M. H. Fatehi *et al.* [21] teve como base para desenvolver um esquema de controle para controlar um sistema de ponte rolante com cabo flexível. O objetivo era a movimentação de uma carga para uma posição final desejada, enquanto simultaneamente reduzir a oscilação da carga útil. Foram conduzidas simulações do comportamento do sistema sem a implementação do controlador proposto, e com o controlador proposto, para observar as diferenças. Os resultados dessas simulações demonstraram que o proposto possibilita a movimentação da carga para a posição desejada, num curto período de tempo (cerca de 8 segundos), e reduzindo significativamente a oscilação da carga (em $\pm 7^\circ$).

S. Zhang *et al.* [10] introduziram um método de controle baseado no controlador Proporcional Integral Derivativo, PID, de modo a atingir uma alta precisão do posicionamento do carro e eliminar as oscilações. Os autores afirmam que o PID, um controlador simples, permite a redução de oscilação e posicionamento da carga com precisão, visível nos resultados experimentais, comprovando as vantagens da utilização deste controle. Também no estudo de V. Caneiro [6] são abordados os controladores PID para o controle da posição do carro e da oscilação da carga. Sendo que, através dos resultados publicados, foi possível verificar a redução da oscilação máxima da carga de $6,9^\circ$ para $3,6^\circ$. Alguns autores propõem a junção do controlador PID com outros tipos de controle. Segundo Q. Zhou *et al.* [11], com o controlador PID tradicional, é “difícil alcançar o efeito de controle ideal”, apresentando um controlador constituído pelo controlador PID e um controlador difuso. Comparando as curvas de resposta do controlador PID com o controlador PID + *difuso*, observou-se que o tempo de ajuste de oscilação da carga e o ângulo máximo são maiores com o controlador PID tradicional. Afirma que o controle proposto é mais estável, tendo melhor precisão de posicionamento da carga e maior capacidade de adaptação a mudanças nas condições de operação.

Outro controlador mencionado na literatura revista é o Regulador Linear Quadrático, LQR. Segundo M. Ali Mohammed *et al.* [22], o LQR permite controlar sistemas complexos que exigem requisitos de desempenho rigorosos, e procura determinar o controlador ótimo que visa minimizar a função de custo (tipicamente definida em termos de diferenças entre o estado atual do sistema e o estado desejado, ou em entradas de controle aplicadas ao sistema). Neste artigo são ainda considerados aspetos importantes, tais como a minimização do *overshoot*, do tempo de crescimento e do tempo de estabelecimento, porque têm impacto no desempenho de operação do sistema, afetando a rapidez da resposta e a proximidade da referência desejada. Como na realidade as pontes rolantes são usadas para transportar vários pesos, com o objetivo de abordar a questão das mudanças de massa da carga, Y. Sun *et al.* [23] propõem um controlador de realimentação de estado LQR robusto para sistemas de pontes rolantes de carga variável. São apresentados vários testes de simulação, variando a massa da carga, para três tipos de controladores: PID, LQR e LQR robusto. Os resultados da simulação mostraram que quanto maior a massa da carga, menor a velocidade de resposta dos três controladores. No entanto, para cargas de massas maiores, o tempo de ajuste e a amplitude do ângulo de oscilação é menor com o LQR robusto, comparando com o controlador PID e o controlador LQR tradicional.

O artigo de J. Smoczek [24] apresenta um sistema de controle baseado na lógica difusa, com a finalidade de redução ou eliminação do ângulo de oscilação de carga, para controlar um modelo dinâmico de ponte rolante em 2D com comprimento variável do cabo. Este autor afirma que através dos resultados experimentais provaram que um sistema de controle baseado na lógica difusa garante a robustez do desempenho do sistema. Também H.-H. Lee e S.-K. Cho [25] apresentam um esquema de controle difuso para eliminar a oscilação da carga para uma ponte rolante 3D. Através dos testes experimentais, os autores verificaram a robustez e a eficácia do sistema de controle proposto, que garantiu o controle preciso da posição da carga,

horizontalmente e verticalmente, e a oscilação da carga é eliminada totalmente 2 segundos depois de atingir as posições desejadas.

O trabalho de A. Mary *et al.* [26] visou projetar um controle adaptativo para um sistema de ponte rolante de duplo pêndulo. O controlador adaptativo foi sugerido para mover o carro para a posição desejada o mais rápido possível e com um ângulo mínimo de oscilação da carga útil. O controle apresentado no estudo leva em consideração a incerteza do sistema e a perturbação externa. Através da simulação, foram comparados o controlador proposto com outros controladores, tal como o PID. Os resultados demonstraram que o proposto é mais robusto que os outros.

H. Le Xuan *et al.* [4] abordam o problema de controlar de forma eficaz e robusta uma ponte rolante 3D sob a incerteza da massa da carga útil, e propõem empregar uma rede neuronal para desenvolver um mecanismo adaptativo para estimar os parâmetros desconhecidos e incertos. Segundo o artigo, os resultados obtidos de simulações da implementação do algoritmo proposto mostraram que o controle é eficaz e consistente, obtendo um ângulo máximo de oscilação de aproximadamente 8° a 4° .

G. Bartolini *et al.* [27] propõem um esquema de controle simples, baseado no controle de Modo Deslizante, do inglês *Sliding Mode Control*, SMC, para garantir o posicionamento da carga de forma precisa e no menor tempo possível, além da supressão de oscilação da carga durante o seu movimento. A confirmação do bom desempenho do método proposto foi realizada através de testes experimentais. Também H.-H. Le *et al.* [28] contribuíram com um estudo sobre o controle SMC para pontes rolantes. Os resultados indicaram que o desempenho do controle proposto não é significativamente afetado pela velocidade ou pela distância de subida/descida da carga. Além disso, foi observada a redução da oscilação da carga ao longo da sua trajetória. Q. Hussein *et al.* [29] também recorreram à utilização deste controlador, e apresentaram estudos e testes comparativos com o controlador LQR. Os resultados mostraram que ambos possuem um desempenho satisfatório, porém o SMC demonstrou maior eficácia e robustez, porque mesmo com alterações de parâmetros, foi possível a movimentação da carga com reduzida oscilação e para o local desejado, com um tempo de acomodação menor.

No estudo de L. Tuan *et al.* [7], é introduzido um controlador adaptativo do SMC para controlar o movimento do carro que suspende uma carga, posicionando-a na posição desejada, realizando operações de elevação e descida da carga, com a mínima oscilação da mesma. Esta versão adaptativa do SMC foi proposta para permitir ao sistema que se ajuste às variações nos parâmetros da ponte rolante, ou seja, sem ter informação da massa da carga útil e dos fatores de atrito. Após as simulações verificaram que o ângulo de oscilação da carga, durante o seu transporte, foi mantido dentro de uma pequena faixa de $0,3^\circ$, sendo eliminado quando a carga útil chega à posição desejada, em 5,2 segundos, e a o movimento do comprimento do carro atingiu a referência em 5,6 segundos. Na prática, o tempo de acomodação da posição do carro e do comprimento foi inferior à simulação, mas o mesmo não se verificou para o ângulo de oscilação.

Outro sistema de controle apresentado na literatura é o modelo de controle preditivo, que calcula as sequências de controle com base no modelo dinâmico para minimizar uma função objetivo. O artigo de F. Debrouwere *et al.* [30] recorre a este modelo, e através dos testes descritos, a abordagem demonstrou-se eficaz porque manteve o desempenho do sistema mesmo quando sujeito influências externas não previstas. Além disso, verificou-se que o controle tem precisão em seguir a referência desejada, possibilitando controlar de forma otimizada uma ponte rolante.

G. O. Tysse *et al.* [31] propõem um sistema de controle constituído por um controlador sustentado em Lyapunov para estabilizar a dinâmica do pêndulo de uma grua articulada, e por um controlador preditivo de modelo não linear, para controlar a posição da carga da grua e o comprimento do cabo. Através das simulações e testes experimentais realizados constatou-se que tanto a posição

da ponta da grua como o comprimento do cabo atingiram a referência desejada. Nos ensaios práticos verificou-se que o ângulo de oscilação estava nos limites de $\pm 2^\circ$ (na simulação estava em $\pm 1,5^\circ$) após 15 segundos do movimento, convergindo para 0° quando a ponta da grua chegou à referência. Na maioria da literatura, a teoria de Lyapunov é abordada, sendo utilizada para analisar a estabilidade de sistemas dinâmicos.

No artigo de M. Rasool Mojallizadeh *et al.* [2] foi ainda abordada a teoria de *flatness*, que pode ser combinada com técnicas de malha fechada, e tem como objetivo seguir com precisão uma trajetória de referência. Em Z. Zhang *et al.* [32] é realizada uma abordagem baseada neste controlo, que pelos resultados apresentados observou-se que o sistema consegue ajustar o movimento do carro quando sujeito a uma perturbação externa, de modo que o carro atinja a sua posição de referência, com reduzida oscilação de carga.

M. Karkoub e M. Zribi [33] projetaram um controlador μ -*synthesis* para controlar a posição do carro, o comprimento do cabo e o ângulo de oscilação da carga da ponte rolante. Os autores responsáveis por este estudo afirmam que o controlador é robusto para dinâmicas que não são completamente consideradas, como variações imprevistas ou efeitos não lineares, tais como o atrito ou ruídos de medição. O controlador proposto conseguiu orientar a ponte rolante para a posição desejada, mantendo o ângulo de oscilação muito pequeno ($\approx 1^\circ$).

L. A. Tuan *et al.* [34], apresentam um modelo de uma ponte rolante 3D e propõem o uso de um sistema de controlo combinado com a linearização por realimentação parcial, para eliminar a oscilação da carga, e o método do modo deslizante, para controlar a elevação da carga e a posição do carro, mesmo com a imprecisão e incerteza do modelo. De acordo com os resultados de simulação e experimentos o controlador proposto estabiliza todos os estados do sistema. Relativamente à posição do carro, o melhor resultado foi do teste experimental, que estabilizou em 4,7 segundos. Já o tempo de acomodação do comprimento do cabo foi de 4 segundos. Por último, o ângulo de oscilação ficou limitado a um intervalo máximo de $\pm 4,3^\circ$, mas atinge valores muito próximos de 0° após 3,6 segundos.

Segundo o artigo de revisão de M. Rasool Mojallizadeh *et al.* [2] existe outro método de controlo, que consiste em controlar a oscilação do ângulo do cabo pela manipulação do cabo, ou seja, o comprimento do cabo é considerado uma entrada de controlo, em vez de uma saída de controlo.

Ainda Y. Shi *et al.* [35] recorrem a um método diferente de controlo que utiliza a visão artificial para detetar a carga através de um sensor visual, e depois planear a trajetória. Este controlo tem como finalidade melhorar a eficiência de operação da ponte rolante. Os resultados de implementação deste método mostraram que o sistema de controlo permite o movimento da carga para o destino desejado com precisão e suprime o ângulo de oscilação da carga.

Na literatura também é referido a utilização do controlo híbrido, que consiste numa combinação de malha aberta e malha fechada. A malha aberta não recorre a sensores, ao contrário do que acontece na técnica de malha fechada. O estudo de H. Yavuz e S. Beller [36] é um exemplo da utilização de métodos híbridos num sistema de pêndulo simples, que utilizaram o método de *input shaping* (método de malha aberta) em conjunto com a rede neural (método de malha fechada).

3. Enquadramento Teórico

3.1. Modelação do Sistema

Na Física, afirma-se que o espaço é homogéneo (invariante à translação) e isotrópico (invariante à rotação) e que o tempo é homogéneo, quando os resultados de uma experiência não dependem expressamente do local no espaço ou do instante no tempo em que esta decorre. Considera-se que existem 3 tipos de invariâncias, sendo estas, a invariância do espaço que se divide em translação e rotação e a invariância do tempo no que respeita a translações temporais.— Em resultado, estabelecem-se as leis de conservação do momento linear, momento angular e energia, respetivamente [37].

As leis de conservação ligadas às invariâncias espaço-temporais são importantes para descrever os movimentos dos objetos no espaço ao longo do tempo. A Mecânica é a parte da Física que trata desses movimentos e das interações que lhes dão origem. Estas leis podem ser reunidas na função de Lagrange $L(r, \dot{r}, t)$, que caracteriza o sistema mecânico e contém toda a informação sobre a evolução mecânica de um objeto ou sistema, onde r representa a coordenada de posição, $\dot{r} \equiv dr/dt$ a velocidade e t o tempo [37].

Tendo em conta que, de entre todas as trajetórias possíveis no espaço e no tempo, o sistema se movimenta segundo a de ação menor, S , definida como:

$$S = \int_{t_i}^{t_f} L(r, \dot{r}, t) dt \quad (3.1)$$

em que t_i é o instante inicial e t_f o instante final, o sistema seguirá a trajetória que torna mínimo o integral da função L ao longo desse caminho (princípio de Hamilton), que corresponde à variação infinitesimal nula $\delta S = 0$. A partir desta condição, para sistemas isolados, é possível obter as respetivas equações de movimento, denominadas por equações de Euler-Lagrange [37].

$$\frac{\partial L}{\partial r} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{r}} \right) = 0 \quad (3.2)$$

Tipicamente, um sistema terá tantas equações de Euler-Lagrange quantos os seus graus de liberdade, ou seja, o número mínimo de variáveis para descrever o estado do sistema [37].

Na Mecânica Clássica, a função de Lagrange para um sistema mecânico e isolado de partículas, em que r_1, r_2, \dots, r_n são as posições das n partículas, e v_1, v_2, \dots, v_n as respetivas velocidades, é dada por

$$L(r_1, r_2, \dots, r_n; v_1, v_2, \dots, v_n; t) \equiv T - U \quad (3.3)$$

em que T é a energia cinética, que mede a energia do movimento, estando relacionada com a velocidade e é definida por:

$$T = \frac{1}{2} m_1 v_1^2 + \frac{1}{2} m_2 v_2^2 + \dots + \frac{1}{2} m_n v_n^2 = \sum_{\alpha} \frac{1}{2} m_{\alpha} v_{\alpha}^2 \quad (3.4)$$

onde m_1, m_2, \dots, m_n são as massas, e U é a energia potencial que mede a energia de interação interna, ligada às distâncias relativas, que pode ser obtida por:

$$U \equiv U(|r_i - r_j|) \quad i = 1, 2, \dots, j = 1, 2, \dots (i \neq j) \quad (3.5)$$

Desta forma, a função de Lagrange, representada pela expressão seguinte, permite caracterizar o comportamento de um sistema, que depende de coordenadas generalizadas q_i e respectivas velocidades \dot{q}_i . Em qualquer instante, estas devem determinar completamente o estado do sistema [37].

$$L(q_i, \dot{q}_i, t) \quad (3.6)$$

De referir que até aqui foi considerado um sistema isolado, no entanto, se existirem forças exteriores, F^{ext} , estas têm de ser adicionadas na equação de Euler-Lagrange, agora exposta:

$$\frac{\partial L}{\partial r} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{r}} \right) = -F^{ext} \quad (3.7)$$

Rescrevendo a equação anterior em coordenadas generalizadas obtém-se:

$$\frac{\partial L}{\partial q_i} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) = -Q_i^{ext} \Leftrightarrow \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = Q_i^{ext} \quad (3.8)$$

onde Q_i^{ext} representa a força exterior generalizada [37].

3.2. Modelação em Espaço de Estados

A teoria de controlo moderno surgiu da necessidade de atender às rigorosas exigências de desempenho e aumento da complexidade dos sistemas de controlo. Divergindo da teoria de controlo convencional, a teoria de controlo moderno pode ser aplicada a sistemas de entradas e saídas múltiplas, que podem ser lineares ou não lineares, variantes ou invariantes no tempo [38].

A teoria de controlo moderno tem como base o conceito de estado e espaço de estados, sendo mais apropriada para analisar um sistema complexo [38].

O estado de um sistema dinâmico é o menor conjunto de variáveis, denominadas variáveis de estado, que determina completamente o seu funcionamento. O conhecimento dessas variáveis, em conjunto com o conhecimento da evolução da entrada e do respetivo estado no instante inicial de análise, permite determinar o comportamento do sistema dinâmico para qualquer instante posterior. Se são necessárias pelo menos n variáveis (x_1, x_2, \dots, x_n) para descrever toda a dinâmica do sistema, então as n variáveis formam um conjunto de variáveis de estado. É de notar que estas variáveis não precisam de ser quantidades físicas, nem mensuráveis ou observáveis. Contudo, em termos práticos, é essencial e benéfico que as variáveis de estado escolhidas sejam de fácil medição. A flexibilidade na escolha das variáveis de estado, uma vez que o conjunto de variáveis de estado não é único, é uma vantagem adicional na teoria de controlo moderno [38]. Ainda, a utilização de um modelo em espaço de estados permite recorrer a ferramentas de álgebra linear para desenvolver um projeto de sistemas de controlo digital [39], tornando mais simples a simulação computacional da sua operação [40].

Ao selecionar as variáveis de estado, é importante ter em conta que o número mínimo de variáveis deve ser adequado para descrever completamente o estado do sistema e essas variáveis têm de ser linearmente independentes entre si. Geralmente, o número mínimo necessário é equivalente à ordem da equação diferencial que descreve o sistema (ou, o que é equivalente, ao número de equações de primeira ordem que descreve) [40].

As n variáveis de estado formam o vetor de estado x . E o espaço n -dimensional, cujos eixos coordenados são formados pelos eixos das variáveis de estado (x_1, x_2, \dots, x_n) é denominado espaço de estados, em que qualquer estado pode ser representado por um ponto nesse espaço.

A análise no espaço de estados envolve três tipos de variáveis que devem ser definidas na modelação do sistema dinâmico: variáveis de entrada ($u_1(t), u_2(t), \dots, u_r(t)$ com r entradas), variáveis de estado ($x_1(t), x_2(t), \dots, x_n(t)$ com n variáveis de estado - ou, simplesmente, estados), e variáveis de saída ($y_1(t), y_2(t), \dots, y_m(t)$ com m saídas).

Considerando f e g funções vetoriais, o modelo em espaço de estados compreende as seguintes equações denominadas equação de estado e equação de saída respetivamente, que modelam: a primeira a evolução (dinâmica) do sistema e a segunda a relação entre o estado (interno) e as saídas (manifestação externa – mensurável – do comportamento do sistema modelado)

$$\dot{x}(t) = f(x, u, t) \quad (3.9)$$

$$y = g(x, u, t) \quad (3.10)$$

em que \dot{x} é a derivada do vetor de estado em relação ao tempo. Se as funções f e/ou g envolverem explicitamente o tempo t , então o sistema é considerado variante no tempo [38].

A função f é constituída por n funções [39],

$$f(x, u, t) = \begin{bmatrix} f_1(x, u, t) \\ \dots \\ f_n(x, u, t) \end{bmatrix} \quad (3.11)$$

Assim, uma representação no espaço de estado consiste nas equações diferenciais de primeira ordem a partir das quais pode ser obtida a solução para as variáveis de estado, e na equação algébrica de saída a partir da qual todas as variáveis do sistema com manifestação externa podem ser obtidas. De salientar, que não há uma única representação no espaço de estados, uma vez que depende da escolha das variáveis de estado e uma escolha diferente das variáveis de estado leva a uma representação diferente do mesmo sistema [40].

3.3. Sistemas Lineares

Considera-se um sistema linear quando este obedece a dois princípios: de sobreposição e de homogeneidade.

O princípio de homogeneidade descreve a resposta do sistema para uma multiplicação da entrada por um escalar. Especificamente, num sistema linear, a propriedade de homogeneidade é demonstrada se, para uma entrada $u_1(t)$ que produz uma saída $y_1(t)$, uma entrada $au_1(t)$ produz uma saída $ay_1(t)$. Isto é, a multiplicação de uma entrada por um escalar produz uma resposta que é multiplicada pelo mesmo escalar [40].

O princípio de sobreposição afirma que a resposta produzida pela aplicação simultânea de duas entradas diversas é a soma de duas respostas individuais. Ou seja, se uma entrada $u_1(t)$ produz uma saída $y_1(t)$ e uma entrada $u_2(t)$ produz uma saída $y_2(t)$, então uma entrada $u_1(t) + u_2(t)$ produz uma saída $y_1(t) + y_2(t)$. Então, para um sistema linear, a resposta a diversas entradas pode ser determinada tratando uma entrada individualmente e somando os resultados [38], [40].

Nos sistemas lineares, o sistema em espaço de estado pode ser representado através das duas equações seguintes,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (3.12)$$

$$y(t) = C(t)x(t) + D(t)u(t) \quad (3.13)$$

em que $A(t)$ é denominada matriz de estado, $B(t)$ matriz de entrada, $C(t)$ a matriz de saída, e $D(t)$ a matriz de transmissão direta. Quando um sistema linear não depende do tempo, este é considerado um sistema linear invariante no tempo, LIT, em que as matrizes A, B, C e D não dependem do tempo. Os sistemas representados por equações diferenciais, cujos coeficientes são funções de tempo, são chamados sistemas lineares variantes no tempo [38].

Por vezes, pretende-se controlar o comportamento de um sistema não linear próximo a um determinado ponto de trabalho, que consiste num vetor de estado constante, x_0 , e uma entrada constante u_0 . No caso em o ponto de trabalho tenha valor de entrada $u_0 = 0$, esse denomina-se ponto de equilíbrio [39].

Se o sistema funcionar em torno de um ponto de equilíbrio e os sinais envolvidos forem pequenos, então é possível aproximar o sistema não linear a um sistema linear, e utilizar o modelo linearizado (Figura 3.1) para o projeto de sistema de controlo [38].

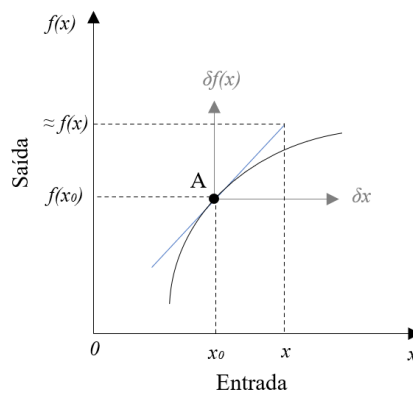


Figura 3.1 - Linearização em torno do ponto A. Adaptado de [40]

Por exemplo, quando um pêndulo está em repouso, ele está em equilíbrio. O seu deslocamento angular é descrito por uma equação diferencial não linear, porém ele pode ser expresso por uma equação diferencial linear para pequenas variações em torno desse ponto de equilíbrio. Ao realizar uma aproximação linear de um sistema não linear é possível simplificar a análise e o projeto de um sistema [40].

A linearização pode ser realizada aplicando a série de Taylor em torno do ponto de trabalho, supondo que as variáveis apresentam pequenos desvios das condições de funcionamento do sistema. Denotando \bar{x} como o ponto de trabalho do sistema, a série de Taylor pode ser expressa pela equação (3.14), em torno desse ponto para realizar uma aproximação linear do comportamento não linear do sistema [38]:

$$\delta x = \frac{\partial f}{\partial x} \Big|_{(x,u)=(\bar{x},0)} \cdot (x - \bar{x}) + \frac{\partial f}{\partial u} \Big|_{(x,u)=(\bar{x},0)} \cdot (u - \bar{u}) \quad (3.14)$$

A matriz de estado A é dada pela matriz jacobiana $\frac{\partial f}{\partial x} \Big|_{(x,u)=(\bar{x},0)}$, e a matriz de estado B pela matriz jacobiana $\frac{\partial f}{\partial u} \Big|_{(x,u)=(\bar{x},0)}$.

No entanto, se as condições operacionais variarem muito, o resultado não será preciso, as equações linearizadas não serão adequadas e devem ser utilizadas equações não lineares [38].

3.3.1. Controlabilidade e Observabilidade do Sistema

Os conceitos de controlabilidade e observabilidade foram introduzidos por Kalman e possuem um papel importante no projeto de sistemas de controlo em espaço de estados. De facto, as condições de controlabilidade e observabilidade podem ditar a existência de uma solução completa para o problema de projeto do sistema de controlo.

Um sistema linear é controlável se for possível transferir o sistema de qualquer estado inicial desejado para qualquer estado final desejado, através da aplicação de controlo num intervalo de tempo finito [38].

Para definir a posição dos pólos do sistema em malha fechada, o sinal de controlo u , pode controlar o comportamento de cada variável de estado em x . Se qualquer uma das variáveis de estado não puder ser controlada pelo controlo u , então não é possível alocar os pólos do sistema nos locais desejados.

Um sistema de ordem n é controlável, se e só se, a matriz de controlabilidade W_c tiver característica n , onde

$$W_c = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B] \quad (3.15)$$

Assim, a partir da matriz de controlabilidade é possível identificar se um sistema é controlável ou não [40], [38].

Ainda, um sistema é dito observável se for possível obter o vetor de estado inicial $x(t_0)$, a partir de $u(t)$ e $y(t)$ medidos durante um intervalo de tempo finito a partir de t_0 . Ou seja, a observabilidade é a capacidade de estimar as variáveis de estado a partir do conhecimento da entrada $u(t)$ e da saída $y(t)$.

Um sistema de matriz de estado de dimensão n é observável se matriz de observabilidade O_m tiver característica n [40], onde

$$O_m = \begin{bmatrix} C \\ CA \\ \dots \\ CA^{n-1} \end{bmatrix} \quad (3.16)$$

3.3.2. Pólos e Resposta do Sistema

O conceito de pólos e raízes é fundamental para a análise e projeto de sistemas de controlo, pois simplifica a determinação da resposta de um sistema.

A resposta e a estabilidade de um sistema de malha fechada estão relacionadas com a localização dos pólos, que correspondem às raízes da equação característica da matriz de estado, e podem ser determinadas pela seguinte equação:

$$\det(sI - A) = 0 \quad (3.17)$$

Caso os pólos estejam localizados apenas no semiplano esquerdo do plano complexo, o sistema é considerado estável. No entanto, se algum pólo estiver no semiplano direito do plano complexo, o sistema é instável. Ainda, se os pólos se localizam sobre o eixo imaginário $j\omega$ e no semiplano da esquerda, o sistema é marginalmente estável, desde que os pólos sobre o eixo imaginário $j\omega$ sejam de multiplicidade unitária, caso contrário o sistema torna-se instável [40].

3.4. Sistemas de Controlo

Um sistema de controlo pode ser constituído por diversos componentes e, tipicamente, pode ser representado por um diagrama de blocos. O diagrama de blocos de um sistema é uma representação gráfica que apresenta as funções executadas por cada componente e o fluxo de sinais entre os diferentes componentes.

Um controlador automático (Figura 3.2) de um sistema compara o valor real de saída com o valor de entrada de referência, determinando o desvio (erro) entre eles, e produz um sinal de controlo que aproxima o desvio para zero ou valor muito próximo de zero [38].

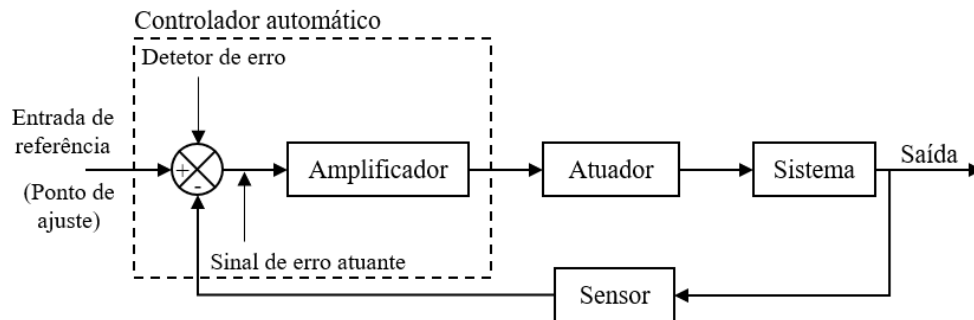


Figura 3.2 - Diagrama de blocos de um sistema de controlo industrial automático. Adaptado de [38]

A saída do controlador automático alimenta um atuador, como por exemplo um motor elétrico. O sensor é um dispositivo de medição que converte a variável de saída numa variável conveniente, como por exemplo em deslocamento, para que possa ser comparada com o sinal de referência de entrada. Este elemento faz parte do ramo de realimentação do sistema de malha fechada. Por sua vez, o ponto de ajuste do controlador deve ser convertido num sinal de referência com as mesmas unidades do sinal de realimentação, que vem do sensor ou do elemento de medição [38].

Sistemas de controlo de malha aberta

Nos sistemas de controlo de malha aberta o sinal de saída não exerce nenhuma ação de controlo no sistema. Isto é, o sinal de saída não é medido, nem realimentado para ser comparado com a entrada de referência. Assim, a cada entrada de referência corresponde uma condição fixa de operação. A precisão do sistema depende de calibração, sendo que na presença de distúrbios, um sistema deste tipo não irá executar a tarefa desejada. Desta forma, o sistema de controlo de malha aberta só poderá ser utilizado se a relação entre a entrada e a saída for conhecida, e sem existir qualquer distúrbio interno ou externo. Um exemplo prático é o da máquina de lavar a roupa [38].

Sistemas de controlo de malha fechada

Um sistema de controlo com realimentação é um sistema que compara a saída com a entrada de referência, e utiliza a diferença como meio para controlar o sistema. Num sistema de controlo de malha fechada, o sinal de erro atuante, que corresponde à diferença entre o sinal de entrada e o sinal de realimentação, realimenta o controlador, de modo a minimizar o erro e atingir ou aproximar a saída do sistema ao valor desejado. Assim, o controlo em malha fechada implica sempre a utilização do controlo com realimentação para reduzir o erro do sistema [38].

Sistemas de controlo de malha fechada versus de malha aberta

Um sistema de controlo de malha fechada, ao utilizar realimentação, faz com que a sua resposta seja relativamente insensível a distúrbios externos e a variações internas nos parâmetros do sistema. No caso de um sistema em malha aberta, é aconselhável o seu uso para sistemas dos quais as entradas são conhecidas antecipadamente, e que estejam isentos ou pouco sujeitos a perturbações pois, caso contrário, irão aparecer erros, e a saída pode apresentar diferenças em relação ao desejado. Do ponto de vista da estabilidade, o sistema de controlo de malha aberta tem uma construção mais simples, porque a estabilidade é um problema menos significativo. Por outro lado, a estabilidade constitui um problema importante nos sistemas de controlo de malha fechada, que podem apresentar uma tendência de correção de erros além do necessário, causando oscilações de amplitude constante ou variável. Além disso, o número de componentes utilizados num sistema de controlo de malha fechada é maior comparativamente com o sistema de malha aberta, o que implica que em geral o sistema de controlo de malha fechada é mais dispendioso [38].

3.4.1. Controlador PID

O controlador PID, esquematizado na Figura 3.3, é muito utilizado para fins industriais. É um tipo de controlo linear em malha fechada, que assume que o comportamento geral do sistema é linear. As variáveis de saída são medidas para determinar a ação de controlo mais indicada para manipular a dinâmica do sistema de acordo com o objetivo de controlo. Este controlador é composto por ações derivativas, integrais e proporcionais.

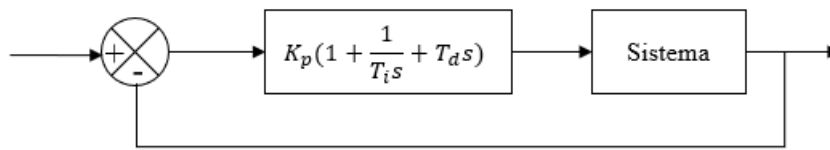


Figura 3.3 - Controlador PID. Adaptado de [38]

A combinação das ações de controlo proporcional, de controlo integral e de controlo derivativo é conhecida como ação de controlo proporcional-integral-derivativo. Essa combinação aproveita as vantagens individuais de cada uma das três ações de controlo. A equação de um controlador com essas ações combinadas é dada por:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (3.18)$$

onde K_p é o ganho proporcional, T_i é denominado tempo integrativo, T_d é denominado tempo derivativo, K_i a constante integral e K_d a constante derivativa, onde K_i e K_d podem ser determinados da seguinte forma, respetivamente:

$$K_i = \frac{K_p}{T} \quad (3.19)$$

$$K_d = K_p T_d \quad (3.20)$$

A parte proporcional é responsável por aumentar ou diminuir a velocidade de resposta, e minimizar o erro estacionário. Para um controlador com ação de controlo proporcional, a relação entre a saída do controlador $u(t)$ e o sinal de erro atuante $e(t)$ é representada pela seguinte expressão.

$$u(t) = K_p e(t) \quad (3.21)$$

O controlo proporcional de um sistema sem integrador pode resultar num erro estacionário, que pode ser eliminado se for incluída no controlador uma ação de controlo integral.

A parte integral suaviza as correções da parte proporcional, e pode eliminar o erro. Num controlador com ação de controlo integral, o valor da saída $u(t)$ do controlador é modificado a uma taxa de variação proporcional ao sinal de erro atuante $e(t)$,

$$\frac{du(t)}{dt} = K_i e(t) \text{ ou } u(t) = K_i \int_0^t e(t) dt \quad (3.22)$$

No controlo integral de um sistema, o sinal de saída do controlador em qualquer instante é a área sob a curva do sinal de erro atuante até aquele momento. De observar que a ação de controlo integral, embora elimine o erro residual ou o erro estacionário, pode conduzir a uma resposta oscilatória com uma amplitude que decresce lentamente ou mesmo uma amplitude sempre crescente, ambas, em geral, indesejáveis.

A parte derivativa prevê o erro atuante, iniciando uma ação corretiva, o que contribui para aumentar a sua estabilidade. Uma ação de controlo derivativo, quando acrescentada a um controlador proporcional permite que se obtenha um controlador de alta sensibilidade a perturbações. Embora o controlo derivativo não afete diretamente o erro estacionário, ele aumenta o amortecimento do sistema, permitindo, assim, o uso de um valor mais elevado do ganho, K_d , o que resultará em maior precisão no regime permanente. Ao atuar em relação à taxa de variação do erro atuante e não sobre o próprio erro, o controlo derivativo nunca é utilizada de forma isolada, encontrando-se sempre combinado com ações de controlo proporcional ou proporcional-integral [38].

3.4.2. Colocação de Pólos

Na Figura 3.4 é apresentado um diagrama de blocos do sistema de controlo de realimentação de estados, em que o sinal de controlo u é determinado por um estado instantâneo.

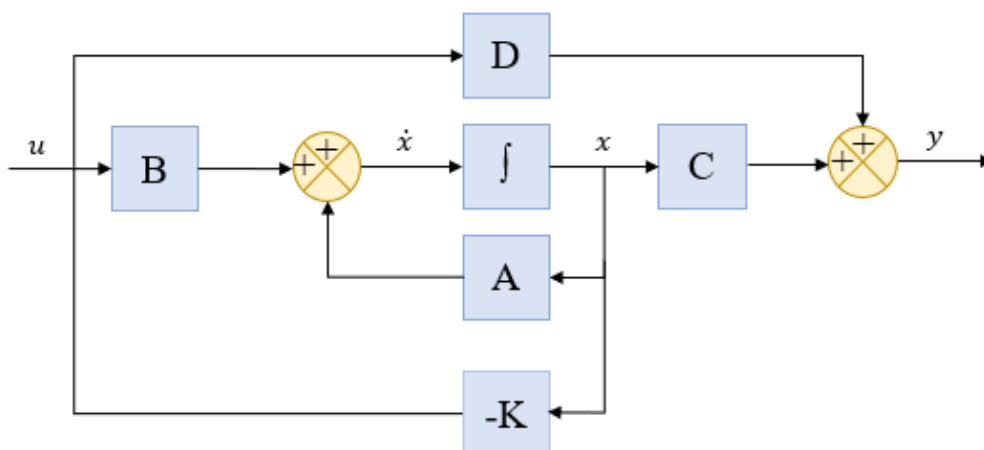


Figura 3.4 - Sistema de controlo com realimentação de estados

Para realimentar todos os estados do sistema, surge a técnica de colocação de pólos, na qual pressupõe que todas as variáveis de estado são mensuráveis e estão disponíveis para realimentação. Só é possível aplicar esse método, se o sistema considerado for completamente controlável, para que através de uma matriz de ganho seja possível colocar os pólos do sistema, em malha fechada, em localizações desejadas [38].

O método de controlo de colocação de pólos requer um modelo em espaço de estados do sistema, de acordo com seguintes equações [38]:

$$\dot{x} = Ax + Bu \quad (3.23)$$

$$y = Cx + D \quad (3.24)$$

Cada variável de estado é realimentada para o sinal de controlo u , através da matriz de ganho, K , constituído por n valores de ganhos, K_i , $i = 1, 2, \dots, n$. Assim, o sinal de controlo é expresso da forma $u = -Kx$, ou seja [39], [40],

$$u = -Kx = [-K_1 \quad -K_2 \quad \dots \quad -K_n] \begin{bmatrix} x_1 \\ \dots \\ x_n \end{bmatrix} \quad (3.25)$$

No caso de o sinal de referência ser nulo, o sistema considera-se um sistema regulador. O objetivo de um regulador é manter as variáveis de estado do sistema no seu valor nominal face às perturbações que atuam no sistema. O sistema regulador pode aplicar-se também quando a entrada de referência é um valor constante, e o objetivo do controlo irá ser manter a saída igual ou próximo do valor de referência [38], [39].

No entanto, há situações em que é necessário ou que se pretende dar uma referência final desejada, não em termos da entrada, mas em termos do estado, e que o sistema seja regulador, para que a diferença entre o estado e a referência de entrada tenda para zero. Para isto, é possível modificar o esquema da Figura 3.4, como representado na Figura 3.5. Onde é adicionando a diferença $x_{ref} - x$, de modo que esta diferença seja regulada para zero, onde x_{ref} , é a referência de entrada que se quer impor ao sistema, e x o estado do sistema num determinado tempo.

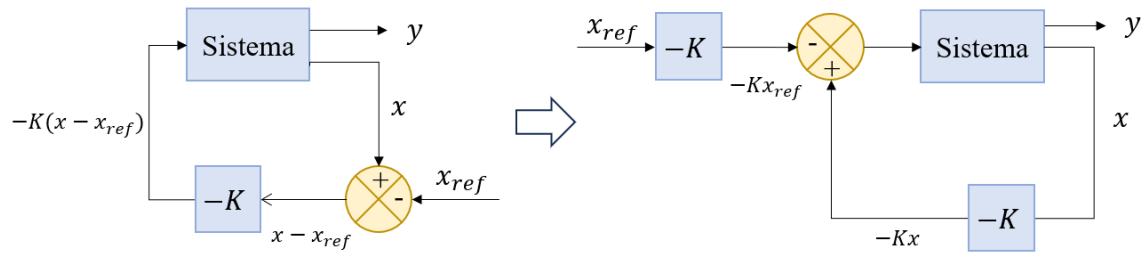


Figura 3.5 - Introdução da referência de entrada, x_{ref} , no sistema de controlo de realimentação de estados para melhor desempenho do controlo

Desta forma, a entrada do sistema é dada por:

$$u = Kx_{ref} - Kx = K(x_{ref} - x) \quad (3.26)$$

E as equações que representam o sistema em espaço de estados ficam [38], [39], [40]:

$$\dot{x} = Ax + B(Kx_{ref} - Kx) \quad (3.27)$$

$$\Leftrightarrow \dot{x} = Ax + BKx_{ref} - BKx \quad (3.28)$$

$$\Leftrightarrow \dot{x} = (A - BK)x + BKx_{ref} \quad (3.29)$$

$$y = Cx \quad (3.30)$$

Os pólos do sistema em malha fechada equivalem aos valores próprios da matriz $A - BK$, que correspondem às raízes do polinómio característico [39].

A estabilidade e a característica da resposta podem ser determinadas através desses valores próprios. O sistema será assintoticamente estável, se ao selecionar a matriz K , os valores próprios da matriz $A - BK$ sejam posicionados no lado esquerdo do plano complexo s [38]. O plano s é uma representação gráfica dos pólos de um sistema dinâmico, no qual o eixo horizontal representa a parte real e o eixo vertical representa a parte imaginária dos números complexos. A localização dos pólos de um sistema neste plano permite analisar a estabilidade, o comportamento e o desempenho do sistema.

É importante ter em consideração que a seleção dos pólos desejados de malha fechada é um compromisso entre a velocidade de resposta do erro e a sensibilidade aos distúrbios e ruídos, o que significa que aumentando a velocidade de resposta do erro, em geral os efeitos nos distúrbios e nos ruídos aumentarão. Além disso, quanto mais afastados da origem, e localizados no semiplano esquerdo do plano complexo s , maior será a amplitude do sinal de controlo. Consequentemente, será desejável analisar, através de simulações computacionais, a resposta característica do sistema para várias matrizes K distintas e escolher aquela que confere o melhor desempenho global do sistema [38].

Caso o sinal de referência se altere com o tempo, já não se estará na presença de um regulador, mas de um sistema seguidor (também chamado frequentemente servomecanismo ou servossistema) [38].

Observador de ordem completa

Na abordagem de realimentação de estados supõe-se que todas as variáveis de estado podem ser realimentadas (ou seja, medidas). No entanto, na prática, nem sempre todas as variáveis estão disponíveis para realimentação, ou porque representam grandezas não mensuráveis ou porque não é economicamente viável medi-las. Caso estejamos perante um sistema em que não são medidos nem conhecidos alguns estados, é necessário estimar essas variáveis de estado através de um observador de estado [39].

O mais conhecido observador de ordem completa (ou seja, que observa todas as variáveis de estado), o observador de Luenberger, estima todos os estados a partir dos valores da saída e da entrada. O conceito de observabilidade tem um papel importante, porque os observadores de estado podem ser projetados se, e somente se, a condição de observabilidade for satisfeita, e implica que a matriz de observabilidade seja invertível [39].

A partir do diagrama de blocos da Figura 3.4 foi adicionado um observador de ordem completa que fornece valores estimados de todas as variáveis de estado, sendo possível observar a sua configuração na Figura 3.6 [38], [39].

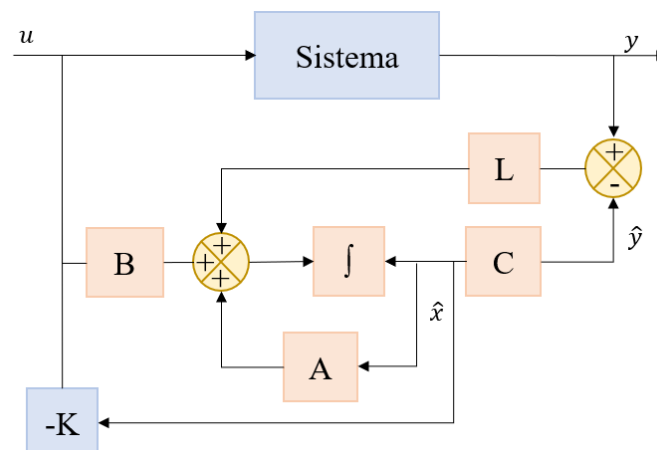


Figura 3.6 - Sistema de realimentação de estados com observador de ordem completa e método de controlo colocação de pólos

O observador é um subsistema e o seu modelo matemático é semelhante ao do sistema, mas tem um termo adicional que incorpora o erro de estimação, em realimentação, para compensar as incertezas nas matrizes A e B, e a ausência do erro inicial [38].

Assim, o vetor de estado observado \hat{x} é utilizado na realimentação de estados para gerar o vetor de controlo desejado. Perante um sistema regulador, com referência nula, a entrada do sistema é dada por [38]:

$$u = -K\hat{x} \quad (3.31)$$

No observador de Luenberger pretende-se que o erro de estimação do estado tenha uma dinâmica convergente, ou seja, que tenda para zero ao longo do tempo.

O erro de estimação ou erro de observação é a diferença entre o vetor de estado real e o vetor de estado medido.

$$e = x - \hat{x} \quad (3.32)$$

No entanto, a própria natureza do problema implica que o estado não é conhecido, não podendo usar-se o erro como sinal de realimentação para o observador. Assim sendo, a dinâmica do estimador vai ser corrigida recorrendo à melhor aproximação possível para esse erro, que é o sinal de erro de estimação da saída, através da inclusão de uma parcela proporcional a este mesmo erro, L . Escrevendo as equações de estado do observador tem-se [39], [40]:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y}) \quad (3.33)$$

em que $(y - \hat{y})$ é o erro entre a saída real e a saída estimada.

Substituindo na equação de estado do sistema obtém-se,

$$\dot{x} = Ax - BK\hat{x} = (A - BK)x + BK(x - \hat{x}) \quad (3.34)$$

Assim, a equação de estado pode ser obtida por [38]:

$$\dot{x} = (A - BK)x + BKe \quad (3.35)$$

É importante verificar qual a influência que a adição desta dinâmica tem no sistema como um todo. Para representar a dinâmica do estado e a dinâmica do estado estimado, são subtraídas as duas equações, a do observador e a do estado, obtendo [38], [39]:

$$(\dot{x} - \dot{\hat{x}}) = A(x - \hat{x}) + Bu - Bu - L(y - \hat{y}) = (A - LC)(x - \hat{x}) \quad (3.36)$$

$$\dot{\varepsilon} = (A - LC)\varepsilon \quad (3.37)$$

$$(y - \hat{y}) = C(x - \hat{x}) \quad (3.38)$$

Incluindo a dinâmica do sistema e da estimação, o sistema composto fica [38], [39]:

$$\begin{bmatrix} \dot{x} \\ \dot{\varepsilon} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x \\ \varepsilon \end{bmatrix} \quad (3.39)$$

$$\hat{y} = C\hat{x} \quad (3.40)$$

onde os pólos do sistema como um todo, resultam da agregação dos pólos do sistema e do observador, sem alteração de nenhum deles, uma vez que os pólos são dados pelo determinante de

$$\left[sI - \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \right] \quad (3.41)$$

ou seja,

$$|sI - A + BK||sI - A + LC| \quad (3.42)$$

Os valores próprios de $(A-LC)$ são os pólos do observador escolhidos, e os valores próprios de $(A - BK)$ são os pólos de malha fechada desejados, que foram especificados quando foi definido o vetor K , para realimentação de estados total [39], [40].

Independente e semelhante do projeto do controlador, o projeto do observador consiste em determinar o vetor L . Para determinar os pólos do observador arbitrariamente, sabe-se que uma matriz tem os mesmos valores próprios da sua transposta, então, determinar os valores próprios de $(A - LC)$ é o mesmo de determinar os valores próprios da matriz $(A - LC)^T$.

Devem ser escolhidos pólos para o observador que estejam suficientemente à esquerda dos pólos do regulador para que a resposta do observador seja projetada para ser mais rápida que a do controlador [39], [40].

O ganho de realimentação K pode ser determinado através da seguinte expressão:

$$K = \bar{K}T_c \quad (3.43)$$

em que $\bar{K} = [\alpha_1 - a_1 \quad \alpha_2 - a_2 \quad \dots \quad \alpha_n - a_n]$ e T_c , que, corresponde à matriz de transformação, é dada pela multiplicação entre a inversa da matriz de controlabilidade do sistema $W_c^{-1} = [B \quad AB \quad \dots \quad A^{n-1}B]^{-1}$ e a matriz de controlabilidade da forma canónica do controlador \bar{W}_c :

$$T_c = \bar{W}_c W_c^{-1} \quad (3.44)$$

onde \bar{W}_c pode ser calculada a partir da matriz de estado e da matriz de entrada do sistema na forma canónica, através dos coeficientes, a , do polinómio caraterístico da matriz original.

Como o sistema é controlável, é possível selecionar os pólos desejados, que dão origem aos coeficientes da equação caraterística, α . Desta forma, é determinado o ganho K .

No caso do observador, pode construir-se um problema dual do anterior, tomando em consideração que $K = L^T$ e $W_c = [C^T \quad A^T C^T \quad (A^T)^{n-1} C^T]$. Portanto, pode usar-se a mesma solução para determinar (e colocar) os pólos do observador.

$$L^T = [\alpha_1 - a_1 \quad \dots \quad \alpha_n - a_n] \bar{W}_c [C^T \quad A^T C^T \quad (A^T)^{n-1} C^T]^{-1} \quad (3.45)$$

Aplicando as propriedades das matrizes transpostas, $(ABC)^T = C^T B^T A^T$ e $(A^{-1})^T = (A^T)^{-1}$, resulta em

$$L = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}^{-1} \bar{W}_c^T \begin{bmatrix} \alpha_1 - a_1 \\ \alpha_n - a_n \end{bmatrix} \quad (3.46)$$

Só é possível determinar L se a matriz $\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}^{-1}$, a matriz de observabilidade do sistema inicial, for invertível, ou seja, se tiver caraterística n .

Observador de ordem reduzida

Em certos casos algumas variáveis de estado podem ser medidas diretamente e com precisão, eliminando a necessidade de estimativas. Como resultado, surge o observador de ordem reduzida, que estima apenas os estados não medidos.

Supondo que o vetor de estado x é um vetor de dimensão n , e que a saída y é um vetor de dimensão m que pode ser medido e que reflete m estados, então m variáveis de estado não precisam de ser estimadas. Desta forma, é apenas necessário estimar $n - m$ variáveis de estado [38].

No entanto, é importante notar que, se a medida das variáveis de saída envolve ruídos significativos e é relativamente imprecisa, então o uso de observadores de ordem completa poderá resultar num desempenho melhor que o uso de observadores de ordem reduzida [38].

Considerando o sistema em espaço de estados

$$\dot{x} = Ax + Bu \quad (3.47)$$

$$y = Cx + D \quad (3.48)$$

onde o vetor de estado x pode ser dividido em x_m e em x_o . O vetor x_m é igual à saída y , sendo possível ser medido, e o vetor x_o é a parte que não é possível medir, então é estimado. Assim, a equação de estado o observador de ordem reduzida determina-se da seguinte forma [38].

$$\begin{bmatrix} \dot{x}_m \\ \dot{x}_o \end{bmatrix} = \begin{bmatrix} A_{mm} & A_{mo} \\ A_{om} & A_{oo} \end{bmatrix} \begin{bmatrix} x_m \\ x_o \end{bmatrix} + \begin{bmatrix} B_m \\ B_o \end{bmatrix} u \quad (3.49)$$

$$y = [I_m \quad 0] \begin{bmatrix} x_m \\ x_o \end{bmatrix} \quad (3.50)$$

em que

A_{mm} = matriz $(n - m) \times (n - m)$

A_{mo} = matriz $(n - m) \times 1$

A_{om} = matriz $1 \times (m - 1)$

A_{oo} = escalar

B_m = matriz $(n - m) \times 1$

B_o = escalar

e \dot{x}_m representa a parte do estado que pode ser diretamente medida, e \dot{x}_o a parte não mensurável diretamente.

Considerando apenas a equação correspondente à partição mensurável do estado obtém-se a equação seguinte.

$$\dot{x}_m = A_{mm}x_m + A_{mo}x_o + B_mu \quad (3.51)$$

Como não conhecemos x_o , isolando as parcelas desconhecidas no lado direito, a equação fica reescrita da seguinte forma:

$$\dot{x}_m - A_{mm}x_m - B_mu = A_{mo}x_o \quad (3.52)$$

Considerando que a equação está na forma de equação de saída do tipo $y' = C'x'$, e tomando as seguintes mudanças de variável:

$$y' = \dot{x}_m - A_{mm}x_m - B_m u \quad (3.53)$$

$$C' = A_{mo} \quad (3.54)$$

$$x' = x_o \quad (3.55)$$

e tendo em conta a partição desconhecida, ou seja, a estimar, obtém se a seguinte equação:

$$\dot{x}_o = A_{om}x_m + A_{oo}x_o + B_o u = A_{oo}x_o + (A_{om}x_m + B_o u) \quad (3.56)$$

É possível identificar esta equação como uma equação de estado do tipo $\dot{x}' = A'x' + Bu$. Sendo que a contribuição da entrada para a evolução do estado é uma função Bu mais complexa e não linear da entrada. Assim, definem-se as seguintes mudanças de variáveis:

$$x' = x_o \quad (3.57)$$

$$A' = A_{oo} \quad (3.58)$$

$$Bu = A_{om}x_m + B_o u \quad (3.59)$$

Consequentemente, é possível construir um sistema particionado em duas partes, para a componente desconhecida [38].

Considerando a equação de estado observado no observador de ordem completa,

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y}) = A\hat{x} + Bu + LC(x - \hat{x}) \quad (3.60)$$

é possível escrever uma equação de estimação da seguinte forma:

$$\dot{\hat{x}}' = A'\hat{x}' + Bu + L(y' - \hat{y}') = A'\hat{x}' + Bu + LC'(x' - \hat{x}') \quad (3.61)$$

$$\dot{\hat{x}}_o = A_{oo}\hat{x}_o + A_{om}x_m + B_o u + LA_{mo}(x_o - \hat{x}_o) \quad (3.62)$$

Assim, a equação fica,

$$\dot{\hat{x}}_o = [A_{om} \quad A_{oo}] \begin{bmatrix} x_m \\ \hat{x}_o \end{bmatrix} + B_o u + L(A_{mo}x_o - A_{mo}\hat{x}_o) \quad (3.63)$$

De seguida, escreve-se a equação dinâmica do erro de estimação da parte do estado não medida,

$$\dot{e}_o = \dot{x}_o - \dot{\hat{x}}_o \quad (3.64)$$

ou seja,

$$\dot{e}_o = A_{oo}x_o + (A_{om}x_m + B_o u) - [A_{oo}\hat{x}_o + A_{om}x_m + B_o u + LA_{mo}(x_o - \hat{x}_o)] \quad (3.65)$$

$$\dot{e}_o = (A_{oo} - LA_{mo})e_o \quad (3.66)$$

Deste modo, a equação característica do observador de ordem reduzida é obtida a partir de:

$$|sI - A_{oo} - LA_{mo}| \quad (3.67)$$

A matriz de ganho L , de dimensão $1 \times (n - m)$, do observador de ordem reduzida pode ser determinada escolhendo os pólos desejados [38].

Para resolver o problema com dados conhecidos, relembrado que

3.5. Discretização do Sistema

Os sistemas de controlo digital podem controlar sistemas em malha fechada a custo reduzido. Por exemplo, caso haja a necessidade de realizar modificações no sistema, estas podem ser implementadas alterando o programa de computador, ao invés de alterações em equipamentos.

As conversões digital-analógica e analógica-digital (Figura 3.8) são requeridas no sistema para assegurar a compatibilidade dos sinais analógicos e digitais ao longo do sistema [40].



Figura 3.8 - Digitalização de um sistema contínuo

Num sistema digital só possuem significado os sinais no instante de amostragem, mas o sistema contínuo que os recebe tem funcionamento entre os instantes de amostragem e, por isso, entre esses instantes usam-se dispositivos retentores. O mais comum é o retentor que mantém constante o último valor de entrada em cada segmento do tempo discreto, denominado retentor de ordem zero, do inglês *zero-order holder*, ZOH.

O modelo discreto relaciona-se com o modelo contínuo da seguinte forma:

$$u(kT) = u[k], \quad kT \leq t < kT+T \quad (3.77)$$

$$y[k] = y(kT) \quad (3.78)$$

em que T é o tempo de amostragem definido, u a entrada e y a saída.

Sendo $G(s)$ a função de transferência no domínio contínuo do sistema e $G(z)$ a correspondente função de transferência no domínio discreto, na Figura 3.9 está representado a diferença entre as entradas e saídas de um sistema contínuo e um sistema discreto.

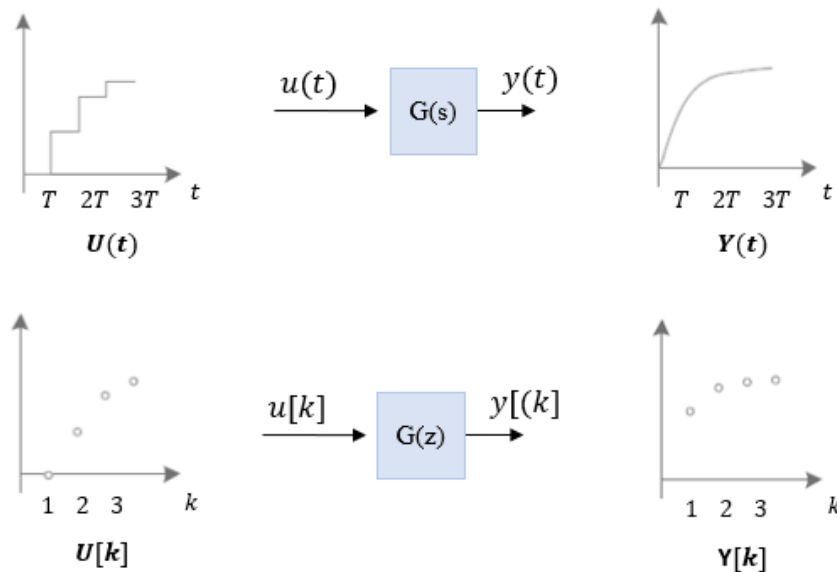


Figura 3.9 - Diferença das entradas e saídas entre sistema contínuo e sistema discreto

O modelo em espaço de estados em tempo discreto é o seguinte:

$$x(k+1) = \Phi x(k) + \Gamma u(k) \quad (3.79)$$

$$y(k) = Cx(k) + Du(k) \quad (3.80)$$

Os termos Φ e Γ podem calcular-se conjuntamente construindo uma matriz MT e aplicando-lhe a exponencial matricial [39]:

$$M = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} \Leftrightarrow e^{MT} = \begin{bmatrix} \Phi & \Gamma \\ 0 & 1 \end{bmatrix} \quad (3.81)$$

Como $\Phi = e^{AT}$, em que A corresponde à matriz de estado do sistema contínuo, é possível determinar os pólos em regime discreto, aplicando a fórmula da relação entre os pólos em s e os pólos em z , com o ZOH:

$$p_z = e^{p_s T} \quad (3.82)$$

A relação entre o plano s o plano z está representada no esquema da Figura 3.10 [39].

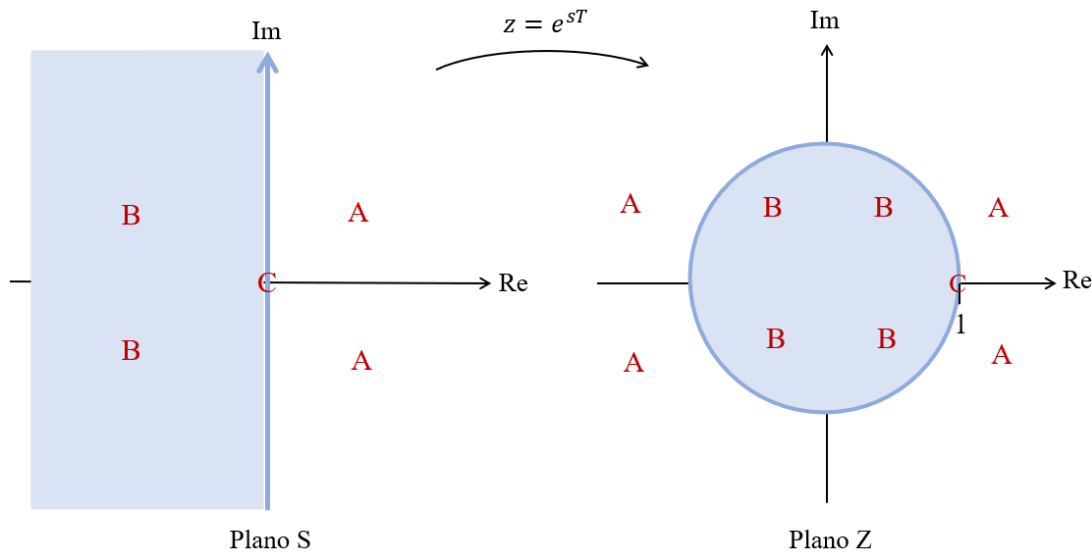


Figura 3.10 - Relação entre os pólos do plano s e do plano z

Cada região do plano s tem uma região correspondente no plano z . Os pontos que possuem parte real positiva estão no semiplano da direita do plano complexo s , ou seja, na região A . Portanto, pontos na metade direita do plano s correspondem a pontos fora do círculo unitário no plano z . Os pontos sobre o eixo jw , região C , possuem valores nulos da parte real e resultam em pontos no plano z com módulo igual 1. Assim, os pontos sobre o eixo jw no plano s estão localizados pontos sobre o círculo unitário no plano z . Já os pontos do plano s que possuem parte real negativa que estão no semiplano da esquerda, na região B , correspondem a pontos no interior do círculo unitário no plano z [40].

Afirma-se que um sistema de controlo digital é [40]:

- estável se todos os pólos estão dentro do círculo unitário no plano z ;
- instável se algum pólo está fora do círculo unitário e/ou se existem pólos de multiplicidade maior que um sobre o círculo unitário;
- marginalmente estável se pólos de multiplicidade um estão sobre o círculo unitário e todos os demais pólos estão dentro do círculo unitário.

3.6. Sistema Linear por Troços

Os sistemas não lineares são mais difíceis de analisar comparativamente aos sistemas lineares, pois num sistema linear as propriedades num determinado ponto são válidas globalmente. Ao contrário dos sistemas não lineares, em que uma análise local pode não dizer nada sobre o comportamento global do sistema. Para chegar a melhores resultados em sistemas não lineares, por vezes, é necessário restringir a classe do sistema, ou fazer aproximações do mesmo.

Assim, surge o sistema linear por troços ou sistema *piecewise linear*, o qual é dividido em múltiplas regiões. Cada região deste sistema possui um modelo linear que é válido apenas dentro dessa região específica. A passagem entre as regiões ocorre quando há alterações nas variáveis de estado ou quando as variáveis de estado atingem um valor pré-definido.

O sistema linear por troços pode surgir da linearização de um sistema não linear em torno de diferentes pontos de trabalho. Assim, o espaço de estados pode ser subdividido num conjunto de regiões X_i , em que cada uma é linear em x , e também cada uma possui as equações que descrevem a sua dinâmica [41].

4. Desenvolvimento do Projeto

4.1. Modelo Matemático do Sistema

Na modelação do sistema, foram consideradas simplificações e feitas considerações, que deu origem ao modelo simplificado do sistema em estudo que está representado na Figura 4.1.

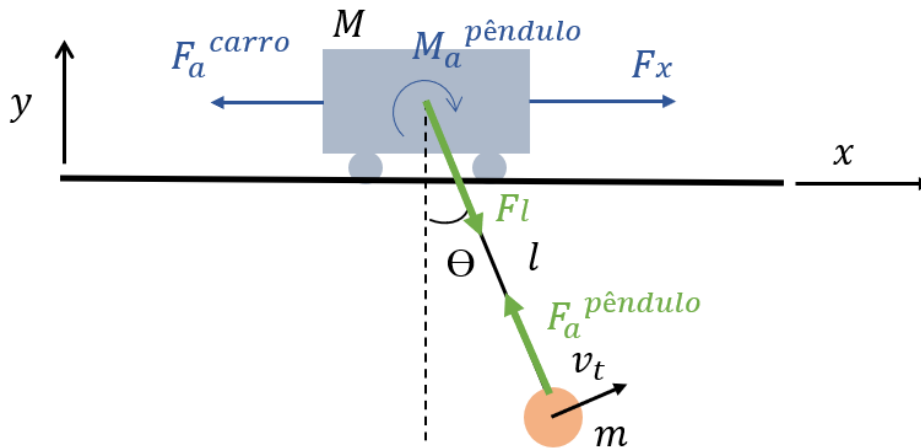


Figura 4.1 - Modelo simplificado do sistema em estudo

O sistema é composto por um carro com deslocamento linear, que se move quando lhe é aplicada uma força, e um pêndulo (ou cabo) com ligação ao carro, que se pode mover verticalmente, quando lhe é aplicada uma força. Na extremidade do cabo encontra-se a carga suspensa.

No modelo exposto salientam-se os seguintes parâmetros:

- M é a massa do carro, em quilograma (kg),
- m é a massa da carga suspensa, em quilograma (kg);
- θ é o ângulo formado entre a posição vertical e a posição instantânea do pêndulo, em radianos (rad);
- l é o comprimento do pêndulo, em metro (m);
- F_x é a força exterior aplicado ao carro, em Newton (N);
- F_a^{carro} é a força de atrito do carro, em Newton (N);
- F_l é a força exterior aplicado ao pêndulo, em Newton (N);
- $F_a^{pêndulo}$ é a força de atrito do pêndulo, em Newton (N);
- $M_a^{pêndulo}$ é o momento de atrito do pêndulo, que tem origem no ponto de suspensão da carga em Newton \times metro ($N \cdot m$);
- v_t é a velocidade tangencial na extremidade do cabo, em metro/segundo (m/s).

Para simplificação do modelo, não foi considerada a massa do pêndulo, conseqüentemente o seu momento de inércia é nulo, e este foi tratado como um corpo rígido. Foi assumido que a massa do carro e da carga suspensa estão concentradas nos respectivos centros de massa. Além disso, o efeito de alongamento entre o carro e a carga, devido à força de tração no pêndulo, foi ignorado.

4.1.1. Função de Lagrange do Sistema

Para deduzir a função de Lagrange do sistema, L , é necessário obter a energia cinética e a energia potencial do sistema. Considerando que o modelo do sistema é constituído por dois corpos ligados, a expressão para a energia cinética do carro, EC_{carro} é:

$$EC_{carro} = \frac{1}{2} M \dot{x}^2 \quad (4.1)$$

A energia cinética do pêndulo, $EC_{pêndulo}$ tem em consideração a energia cinética de translação e de rotação,

$$EC_{pêndulo} = EC_{translação} + EC_{rotação} \quad (4.2)$$

$$EC_{pêndulo} = \frac{1}{2} m \dot{x}^2 + \frac{1}{2} I_{cabo} \dot{\theta}^2 \quad (4.3)$$

Na velocidade do centro de massa do pêndulo considera-se tanto o movimento do carro como o movimento do pêndulo. Além disso, a velocidade tangencial do pêndulo, v_t , pode ser decomposta em v_x , eixo de abcissas e em v_y , eixo das ordenadas. Assim, a velocidade é dada por:

$$v = (\dot{x} + v_x)^2 + v_y^2 \quad (4.4)$$

determinado as expressões de v_x e v_y da seguinte forma (Figura 4.2):

$$v_x = l \dot{\theta} \cos \theta \quad (4.5)$$

$$v_y = l \dot{\theta} \sin \theta \quad (4.6)$$

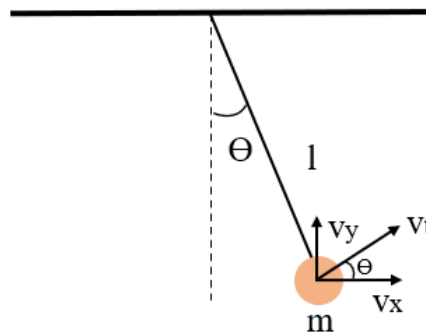


Figura 4.2 - Decomposição da velocidade tangencial da carga suspensa

Obtendo a energia cinética por,

$$EC_{pêndulo} = \frac{1}{2} m [(\dot{x} + v_x)^2 + v_y^2] + \frac{1}{2} I_{cabo} \dot{\theta}^2 \quad (4.7)$$

Dado que o comprimento do pêndulo varia, considerou-se a taxa de variação do comprimento ao longo do tempo, e tendo em conta o movimento nos dois eixos, foram tidos em conta dois componentes, sendo eles $l \sin \theta$ do eixo das abcissas, e $-l \cos \theta$ do eixo das ordenadas.

Considerado que o momento de inércia do cabo é nulo, $I_{cabo} = 0$, deduziu-se a expressão para a energia cinética do pêndulo por,

$$EC_{p\grave{e}ndulo} = \frac{1}{2}m[(\dot{x} + l\dot{\theta}\cos\theta + \dot{l}\sin\theta)^2 + (l\dot{\theta}\sin\theta - \dot{l}\cos\theta)^2] \quad (4.8)$$

Depois de determinadas as expressões de energia cinética dos dois corpos, a energia cinética do sistema é descrita pela seguinte expressão:

$$EC_{sistema} = EC_{carro} + EC_{p\grave{e}ndulo} \quad (4.9)$$

onde

$$EC_{sistema} = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m[(\dot{x} + l\dot{\theta}\cos\theta + \dot{l}\sin\theta)^2 + (l\dot{\theta}\sin\theta - \dot{l}\cos\theta)^2] \quad (4.10)$$

$$EC_{sistema} = \frac{1}{2}\dot{x}^2(M + m) + m\dot{x}(l\dot{\theta}\cos\theta + \dot{l}\sin\theta) + \frac{1}{2}m(l\dot{\theta})^2 + \frac{1}{2}m\dot{l}^2 \quad (4.11)$$

Uma vez que o carro não tem variação de deslocamento vertical, a energia potencial do carro é

$$EP_{carro} = 0 \quad (4.12)$$

Por outro lado, a energia potencial do pêndulo pode ser determinada, considerando como referência o eixo onde o carro se movimenta, como representado na Figura 4.3.

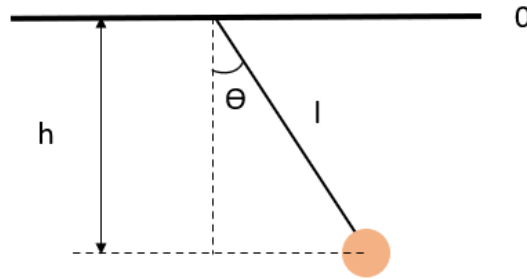


Figura 4.3 - Energia potencial do pêndulo

Assim, a energia potencial é dada por:

$$h = l\cos\theta \quad (4.13)$$

$$EP_{p\grave{e}ndulo} = -mgl\cos\theta \quad (4.14)$$

Do mesmo modo que a energia cinética, a expressão da energia potencial do sistema é dada pela soma da energia potencial do carro e do pêndulo,

$$EP_{sistema} = EP_{carro} + EP_{p\grave{e}ndulo} = -mgl\cos\theta \quad (4.15)$$

Como base na equação (3.3), a função de Lagrange do sistema pode ser determinada a partir da diferença entre as expressões de energia cinética e energia potencial, resultando na seguinte equação:

$$L = \frac{1}{2}[\dot{x}^2(M + m) + 2m\dot{x}(l\dot{\theta}\cos\theta + \dot{l}\sin\theta) + m(l\dot{\theta})^2 + m\dot{l}^2] + mgl\cos\theta \quad (4.16)$$

4.1.2. Equações de Movimento

A equação de movimento segundo a coordenada generalizada da posição linear do carro, x , pode ser obtida através da equação de Euler-Lagrange:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_x} \right) - \frac{\partial L}{\partial q_x} = Q_x^{ext} \quad (4.17)$$

As coordenadas generalizadas do sistema, q , correspondem aos três graus de liberdade (x , θ e l) do sistema em estudo,

$$q = \begin{bmatrix} x \\ \theta \\ l \end{bmatrix} \quad (4.18)$$

Na equação de Euler-Lagrange foram adicionadas as forças exteriores generalizadas, que correspondem a forças exteriores não conservativas. Segundo cada coordenada generalizada, Q^{ext} ,

$$Q^{ext} = \begin{bmatrix} F_{carro} - F_a^{carro} \\ -M_a^{p\grave{e}ndulo} \\ F_{p\grave{e}ndulo} - F_a^{p\grave{e}ndulo} \end{bmatrix} = \begin{bmatrix} F_x - D_x \dot{x} \\ -D_\theta \dot{\theta} \\ F_l - D_l \dot{l} \end{bmatrix} \quad (4.19)$$

Na expressão (4.19), as forças de atrito foram definidas como o produto entre uma constante de amortecimento equivalente e a velocidade (linear ou angular) do corpo num determinado instante, onde D_x , D_θ e D_l são os coeficientes de amortecimento equivalentes ao deslizamento do carro, à rotação do carro, e à subida/descida do pêndulo, respetivamente.

Desta forma, determinando para cada termo:

$$\frac{\partial L}{\partial \dot{x}} = \dot{x}(M + m) + m(l\dot{\theta}\cos\theta + \dot{l}\sin\theta) \quad (4.20)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{x}} \right) = \ddot{x}(M + m) + m(\dot{l}\sin\theta + 2\dot{l}\dot{\theta}\cos\theta + l\ddot{\theta}\cos\theta - l\dot{\theta}^2\sin\theta) \quad (4.21)$$

$$\frac{\partial L}{\partial x} = 0 \quad (4.22)$$

$$Q_x^{ext} = F_x - D_x \dot{x} \quad (4.23)$$

obtém-se a equação de movimento segundo x :

$$\ddot{x}(M + m) + m(\dot{l}\sin\theta + 2\dot{l}\dot{\theta}\cos\theta + l\ddot{\theta}\cos\theta - l\dot{\theta}^2\sin\theta) = F_c - D_x \dot{x} \quad (4.24)$$

Da mesma forma, a equação de movimento segundo a coordenada generalizada θ pode ser determinada por:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_\theta} \right) - \frac{\partial L}{\partial q_\theta} = Q_\theta^{ext} \quad (4.25)$$

$$\frac{\partial L}{\partial \dot{\theta}} = m\dot{x}l\cos\theta + ml^2\dot{\theta} \quad (4.26)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) = m\ddot{x}l\cos\theta + m\dot{x}l\cos\theta - m\dot{x}l\dot{\theta}\sin\theta + 2ml\ddot{\theta} + ml^2\ddot{\theta} \quad (4.27)$$

$$\frac{\partial L}{\partial \theta} = m\dot{x}(-l\dot{\theta}\sin\theta + l\cos\theta) - mgl\sin\theta \quad (4.28)$$

$$Q_{\theta}^{ext} = -D_{\theta}\dot{\theta} \quad (4.29)$$

A expressão da equação de movimento segundo o ângulo de oscilação da carga θ fica:

$$m\ddot{x}l\cos\theta + 2ml\ddot{\theta} + ml^2\ddot{\theta} + mgl\sin\theta = -D_{\theta}\dot{\theta} \quad (4.30)$$

A terceira equação determinada corresponde à equação de movimento segundo a coordenada generalizada l ,

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_l} \right) - \frac{\partial L}{\partial q_l} = Q_l^{ext} \quad (4.31)$$

$$\frac{\partial L}{\partial \dot{l}} = m\dot{x}\sin\theta + ml \quad (4.32)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{l}} \right) = m\ddot{x}\sin\theta + m\dot{x}\dot{\theta}\cos\theta + m\ddot{l} \quad (4.33)$$

$$\frac{\partial L}{\partial l} = m\dot{x}\dot{\theta}\cos\theta + ml\dot{\theta}^2 + mg\cos\theta \quad (4.34)$$

$$Q_l^{ext} = F_l - D_l\dot{l} \quad (4.35)$$

Resultando na equação de movimento segundo o comprimento do pêndulo, l ,

$$m\ddot{x}\sin\theta + m\ddot{l} - ml\dot{\theta}^2 - mg\cos\theta = F_l - D_l\dot{l} \quad (4.36)$$

Estas três equações deduzidas (4.24), (4.30) e (4.36) caracterizam o movimento dinâmico do sistema em estudo.

4.2. Modelo em Espaço de Estados

De acordo com as características e vantagens da representação em espaço de estados mencionadas, optou-se pela representação do modelo matemático determinado em espaço de estados.

Uma vez que foi deduzido um sistema de três equações diferenciais de segunda ordem foram selecionadas 6 variáveis de estado: x , \dot{x} , θ , $\dot{\theta}$, l e \dot{l} , que formam o vetor de estado x :

$$x = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6]^T = [x \ \dot{x} \ \theta \ \dot{\theta} \ l \ \dot{l}]^T \quad (4.37)$$

Na escolha deste vetor foi tido em conta que as variáveis teriam de ser facilmente quantificadas através de sensores, e que correspondessem a derivadas de ordem tão baixa quanto possível.

De modo a determinar a equação de estado foram reescritas as equações de movimento da seguinte forma.

$$\dot{x}_1 = \dot{x} = x_2 \quad (4.38)$$

$$\dot{x}_3 = \dot{\theta} = x_4 \quad (4.39)$$

$$\dot{x}_5 = \dot{l} = x_6 \quad (4.40)$$

Assim, foi possível determinar o vetor de estado \dot{x} :

$$\dot{x} = [\dot{x}_1 \ \dot{x}_2 \ \dot{x}_3 \ \dot{x}_4 \ \dot{x}_5 \ \dot{x}_6]^T = \begin{bmatrix} \frac{F_c x_5 - D_x x_2 x_5 - F_l \sin x_3 x_5 + D_l x_6 \sin x_3 x_5 + D_\theta x_4 \cos x_3}{M x_5} \\ \frac{D_\theta x_4 M - 2 m x_4 x_5 x_6 M - m g x_5 \sin x_3 M - m x_5 \cos x_3 F_c + m x_2 x_5 \cos x_3 D_x + m x_5 \cos x_3 \sin x_3 F_l - m x_5 x_6 \cos x_3 \sin x_3 D_l - m \cos^2 x_3 D_\theta x_4}{M x_5^2} \\ \frac{F_l M x_5 - D_l x_5 x_6 M + m x_5^2 x_4^2 M + g m \cos x_3 M x_5 - m \sin x_3 F_c x_5 + m \sin x_3 D_x x_2 x_5 + m \sin^2 x_3 F_l x_5 - m \sin^2 x_3 D_x x_5 x_6 - m \sin x_3 \cos x_3 D_\theta x_4}{m M x_5} \end{bmatrix} \quad (4.41)$$

Analisando a expressão (4.41) verifica-se que a variável x_1 não aparece na equação de estado do sistema. Consequentemente, esta não apresenta influência direta no estado do sistema, ou seja, a dinâmica deste não é diretamente influenciada pela posição do carro.

4.2.1. Linearização do Modelo

Uma vez que o sistema em estudo se assemelha a um sistema de pêndulo que irá atuar em torno de um ponto de equilíbrio, é possível aproximar o sistema não linear a um sistema linear.

Desta forma, o ponto de equilíbrio do sistema, \bar{x} , é a posição no qual o pêndulo permanece em repouso quando não está a sofrer nenhuma força externa (apenas a força do peso), ou seja, $\bar{u} = 0$:

$$\bar{u} = [F_c \ F_l]^T = [0 \ m g]^T \quad (4.42)$$

Assim, definiu-se que o ponto de equilíbrio do sistema é:

$$\bar{x} = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6]^T = [0 \ 0 \ 0 \ 0 \ l_0 \ 0]^T \quad (4.43)$$

em que l_0 corresponde ao comprimento do pêndulo no ponto de equilíbrio.

Para realizar a linearização do modelo não linear, recorreu-se à expansão da série de Taylor em torno de um ponto de trabalho, que neste caso corresponde ao ponto de equilíbrio referido, a partir da equação (4.44).

$$\dot{x} = \frac{\partial f}{\partial x} \Big|_{(x,u)=(\bar{x},0)} \bullet x + \frac{\partial f}{\partial u} \Big|_{(x,u)=(\bar{x},0)} \bullet u \quad (4.44)$$

A matriz jacobiana $\frac{\partial f}{\partial x} \Big|_{(x,u)=(\bar{x},0)}$ corresponde à matriz de estado, A , enquanto a matriz jacobiana $\frac{\partial f}{\partial u} \Big|_{(x,u)=(\bar{x},0)}$ corresponde à matriz de entrada B .

Desta forma, foi determinada a matriz de estado,

$$A = \frac{\partial f}{\partial x} \Big|_{(x,u)=(\bar{x},0)} = \begin{bmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} & \frac{\partial \dot{x}_1}{\partial x_3} & \frac{\partial \dot{x}_1}{\partial x_4} & \frac{\partial \dot{x}_1}{\partial x_5} & \frac{\partial \dot{x}_1}{\partial x_6} \\ \frac{\partial \dot{x}_2}{\partial x_1} & \frac{\partial \dot{x}_2}{\partial x_2} & \frac{\partial \dot{x}_2}{\partial x_3} & \frac{\partial \dot{x}_2}{\partial x_4} & \frac{\partial \dot{x}_2}{\partial x_5} & \frac{\partial \dot{x}_2}{\partial x_6} \\ \frac{\partial \dot{x}_3}{\partial x_1} & \frac{\partial \dot{x}_3}{\partial x_2} & \frac{\partial \dot{x}_3}{\partial x_3} & \frac{\partial \dot{x}_3}{\partial x_4} & \frac{\partial \dot{x}_3}{\partial x_5} & \frac{\partial \dot{x}_3}{\partial x_6} \\ \frac{\partial \dot{x}_4}{\partial x_1} & \frac{\partial \dot{x}_4}{\partial x_2} & \frac{\partial \dot{x}_4}{\partial x_3} & \frac{\partial \dot{x}_4}{\partial x_4} & \frac{\partial \dot{x}_4}{\partial x_5} & \frac{\partial \dot{x}_4}{\partial x_6} \\ \frac{\partial \dot{x}_5}{\partial x_1} & \frac{\partial \dot{x}_5}{\partial x_2} & \frac{\partial \dot{x}_5}{\partial x_3} & \frac{\partial \dot{x}_5}{\partial x_4} & \frac{\partial \dot{x}_5}{\partial x_5} & \frac{\partial \dot{x}_5}{\partial x_6} \\ \frac{\partial \dot{x}_6}{\partial x_1} & \frac{\partial \dot{x}_6}{\partial x_2} & \frac{\partial \dot{x}_6}{\partial x_3} & \frac{\partial \dot{x}_6}{\partial x_4} & \frac{\partial \dot{x}_6}{\partial x_5} & \frac{\partial \dot{x}_6}{\partial x_6} \end{bmatrix} \quad (4.45)$$

$$A = \frac{\partial f}{\partial x} \Big|_{(x,u)=(\bar{x},0)} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -D_x & -mg & D_\theta & 0 & 0 \\ 0 & M & M & Ml_0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & D_x & -g & -D_\theta(M+m) & 0 & 0 \\ 0 & Ml_0 & l_0 & \frac{-D_\theta(M+m)}{mMl_0^2} & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -D_l \\ 0 & 0 & 0 & 0 & 0 & m \end{bmatrix} \quad (4.46)$$

Do mesmo modo, obteve-se a matriz de entrada,

$$B = \frac{\partial f}{\partial u} \Big|_{(x,u)=(\bar{x},0)} = \begin{bmatrix} \frac{\partial \dot{x}_1}{\partial f_c} & \frac{\partial \dot{x}_1}{\partial f_l} \\ \frac{\partial \dot{x}_2}{\partial f_c} & \frac{\partial \dot{x}_2}{\partial f_l} \\ \frac{\partial \dot{x}_3}{\partial f_c} & \frac{\partial \dot{x}_3}{\partial f_l} \\ \frac{\partial \dot{x}_4}{\partial f_c} & \frac{\partial \dot{x}_4}{\partial f_l} \\ \frac{\partial \dot{x}_5}{\partial f_c} & \frac{\partial \dot{x}_5}{\partial f_l} \\ \frac{\partial \dot{x}_6}{\partial f_c} & \frac{\partial \dot{x}_6}{\partial f_l} \end{bmatrix} \quad (4.47)$$

$$B = \frac{\partial f}{\partial u} \Big|_{(x,u)=(\bar{x},0)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ M & 0 \\ 0 & 0 \\ -1 & 0 \\ \frac{Ml_0}{m} & 1 \\ 0 & \frac{1}{m} \\ 0 & 0 \end{bmatrix} \quad (4.48)$$

A partir das matrizes A e B , é possível obter a equação de estado do modelo linearizado:

$$\dot{x} = Ax + Bu \quad (4.49)$$

No sistema determinado a matriz de transmissão direta D é nula porque a variável de entrada é uma força que não apresenta relação direta com as variáveis de saída. Desta forma, a equação de saída do modelo fica:

$$y = Cx \quad (4.50)$$

onde, uma vez que a instrumentação existente apenas permite medir as variáveis x, θ e l , a matriz C foi definida como

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (4.51)$$

4.2.2. Comparação dos Modelos Não Linear e Linear

Após a obtenção do modelo não linear e modelo linear procedeu-se à comparação dos dois modelos em malha aberta, quando sujeitos às mesmas condições iniciais. Para esta análise desenvolveu-se um programa através do *software* MATLAB, exposto no Apêndice A. Foram introduzidos os valores das constantes físicas do sistema, as condições iniciais e tempo total da simulação, para obter gráficos da simulação correspondentes ao ângulo de oscilação e ao comprimento do cabo, de ambos os modelos, para perceber os seus comportamentos ao longo do tempo definido.

Numa simulação inicial considerou-se um pêndulo simples de comprimento fixo, e sem deslocamento do carro (ou seja $F_x = 0 \text{ N}$ e $F_l = 0 \text{ N}$). Assim, as condições tomadas foram:

$$\bar{x} = [0 \ 0 \ 0 \ 0 \ l_0 \ 0]^T \text{ e } \bar{u} = [0 \ 0]^T \quad (4.52)$$

De notar que foi considerado que no ponto de equilíbrio o comprimento do cabo corresponde ao ponto médio do seu comprimento, ou seja, a linearização do modelo foi feita em torno de $l_0 = 1 \text{ m}$.

Colocou-se $F_l = 0$ no modelo linear, para que se garantisse que não havia aceleração em l , ou seja, não haveria movimento em l . Depois, considerou-se expressão $\dot{x}_6 = \ddot{l} = 0$ de modo a determinar tal F_l no modelo não linear que não houvesse aceleração em l .

Na Figura 4.4 está representado o resultado obtido. Verifica-se que os modelo linear e não linear se sobrepõem, validando as opções tomadas.

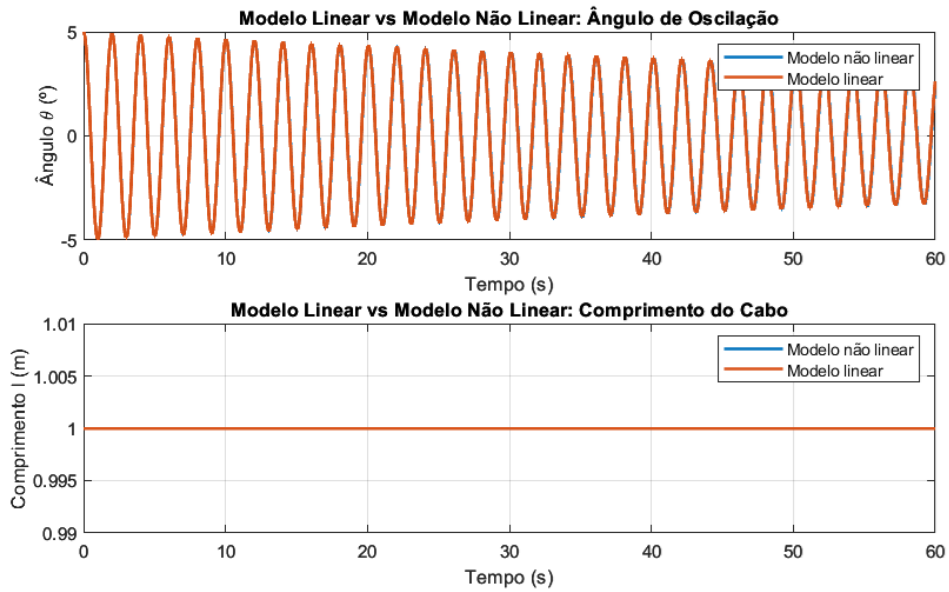


Figura 4.4 - Modelo Linear versus Modelo Não Linear para ângulo de oscilação e comprimento do cabo, com carro parado e comprimento do cabo constante

Numa situação seguinte, considerou-se um pêndulo simples que varia o comprimento, sem deslocamento do carro (ou seja $F_x = 0 N$), sendo necessário impor uma função à força F_l , para realizar movimento no cabo. Num primeiro caso definiu-se F_l não linear e F_l linear, com uma função sinusoidal do tipo $y = A \sin(\omega t)$. Tendo em conta $l_{inicial} = 1 m$ e $l_0 = 1 m$ obteve-se o gráfico da Figura 4.5.

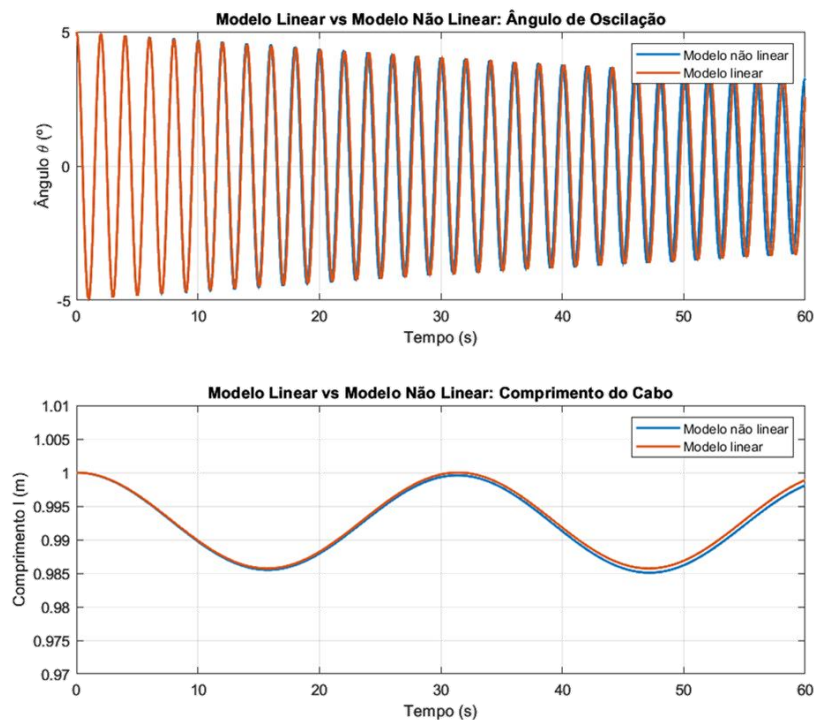


Figura 4.5 - Modelo Linear versus Modelo Não Linear para ângulo de oscilação e comprimento do cabo, com carro parado e comprimento do cabo varia perto em torno de 1 metro

Numa terceira simulação foi considerado que o valor de ponto de equilíbrio de l variava de acordo com classes definidas previamente. Ou seja, o comprimento do cabo foi dividido em troços, e conforme a leitura do comprimento do cabo num determinado instante, é selecionado um valor l (de acordo com as classes), que seria tomado como o ponto do qual seria realizada a linearização do sistema, l_0 .

Desta forma, foi desenvolvida uma função no programa MATLAB, que ao longo do tempo, o l tomava determinado valor. De seguida, foram comparados os gráficos do modelo não linear, modelo linear e o modelo linear com l dividido em troços. Na Figura 4.6 estão apresentados os resultados.

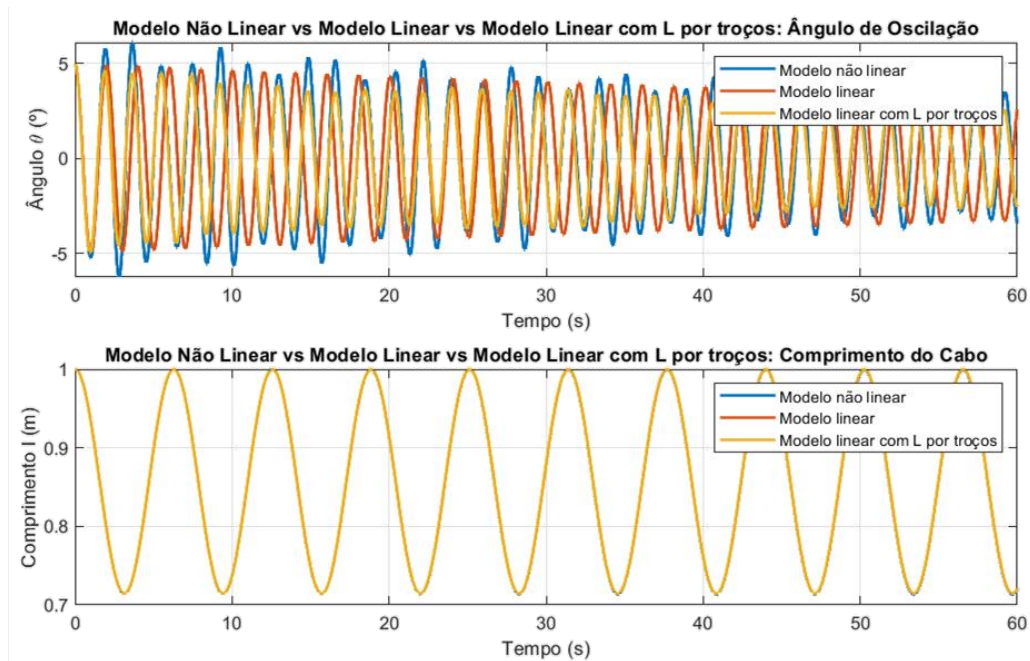


Figura 4.6 - Modelo Linear versus Modelo Não Linear versus Modelo Linear com l dividido em troços considerando que comprimento inicial do cabo é igual a 1

Graficamente, observa-se que o modelo não linear se assemelha mais ao modelo linear em troços, do que o modelo linear ao modelo não linear.

Na Tabela 4.1 estão apresentados valores que permitem comparar os modelos.

Tabela 4.1 - Comparação dos modelos através do coeficiente de Correlação de Pearson e valor Euclidiano

	Euclidiano	Correlação
Não Linear vs. Linear por Troços	1,09	0,98
Não Linear vs. Linear	5,74	$-2,08 \times 10^{-3}$
Linear vs. Linear por Troços	5,18	$-9,66 \times 10^{-3}$

Para a correlação de Pearson, quando mais perto de 1 for o resultado melhor a correlação entre os modelos. No caso do valor Euclidiano quanto mais perto de zero, mais próximos são os resultados.

Assim, confirma-se que entre o modelo não linear e o modelo linear por troços existe uma maior relação e semelhança. A partir de simulações, verificou-se que quanto menores as oscilações em torno do valor do ponto médio do comprimento, 1 metro, mais próximos são os modelos.

4.3. Projeto Mecânico

O equipamento pré-existente no Laboratório de Robótica Avançada e de Fábricas Inteligentes, consistia num carril de 18 metros de comprimento, onde se movimentava um carro de 1,5 kg ao qual estava ligada um tubo rígido com comprimento de 1,045 metros. Na sua extremidade encontrava-se a carga suspensa (Figura 4.7). O movimento do carro era possível através de um motor de passo conectado ao carro, por uma correia. Neste sistema, o objetivo era a redução do ângulo de oscilação de carga, e o posicionamento desejado do carro. Para realizar as medições do ângulo de oscilação e da posição do carro, utilizaram um *encoder* absoluto afixado na estrutura do carro, e o *encoder* incremental do motor do passo, respetivamente.



Figura 4.7 - Projeto do carro na equipamento pré-existente. Fonte: [42]

Um dos objetivos deste projeto foi dimensionar um equipamento capaz de elevar e descer uma carga suspensa, e para isso o sistema foi adaptado mecanicamente e eletricamente. Para possibilitar o movimento vertical da carga suspensa, foi necessário realizar o dimensionamento de um sistema de elevação de carga, e a substituição do tubo utilizado por um cabo de linho de diâmetro 2,5 mm. Este cabo possibilita o seu enrolamento e desenrolamento, e ao mesmo tempo é resistente para suportar a massa da carga. Esta escolha foi realizada, a partir de testes com cabos de natureza e diâmetros diferentes, para avaliar a resistência à massa da carga suspensa, aproveitando o sistema já existente. Uma vez que foram adicionados elementos, foi acrescentado um novo cabo para fazer ligações elétricas dos mesmos, e exigiu realizar alterações no controlo do sistema. Esses elementos incluem um motor DC, a driver do respetivo motor, e um sensor de fim de curso.

As alterações realizadas no equipamento pré-existente serão apresentadas de seguida.

4.3.1. Estudo do Sistema de Elevação de Carga

Inicialmente, foi importante realizar uma análise ao espaço disponível que tínhamos. O local mais apropriado seria utilizar o carro já existente (Figura 4.8), de modo que a subida e descida do cabo acompanhasse o seu movimento.

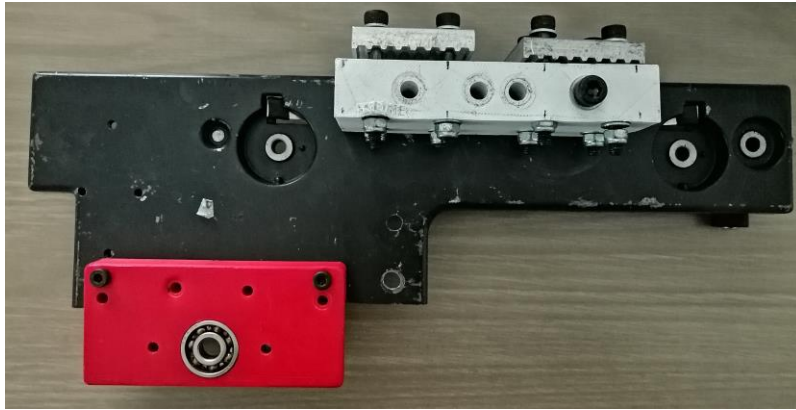


Figura 4.8 - Carro

O projeto mecânico da estrutura de elevação de carga foi desenvolvido a partir do *software SolidWorks*, e o resultado está apresentado na Figura 4.9.

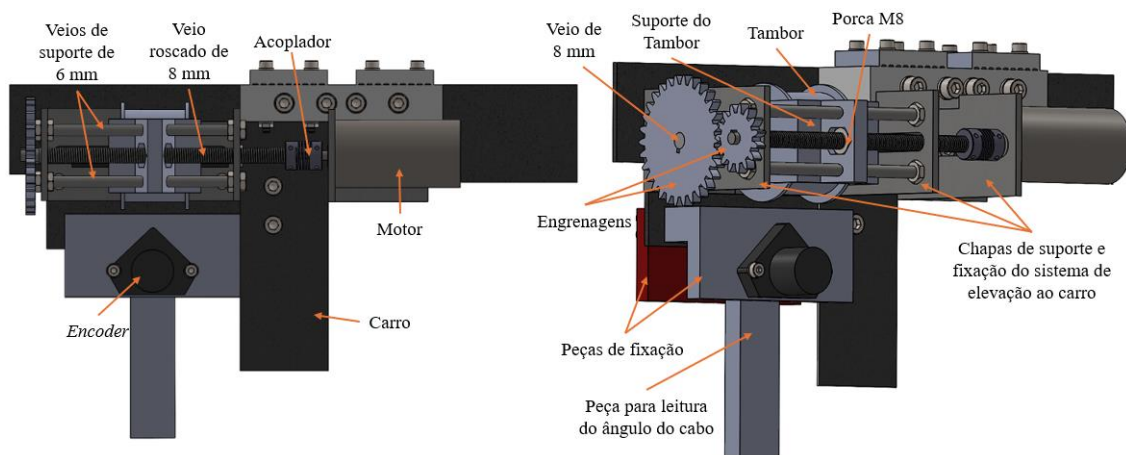


Figura 4.9 - Desenho 3D do projeto mecânico em SolidWorks

Ao longo deste processo foi necessário utilizar várias peças para a construção do equipamento. Recorreu-se à tecnologia de impressão 3D para fabricar: engrenagens, suporte do tambor, tambor, as peças de fixação e a peça para leitura do ângulo do cabo. Já algumas das peças adquiridas em metal foram: 1 parafuso roscado de 8 mm, 2 veios de 6 mm, 3 chapas de metal, porcas M8 e M6, anilhas, outros parafusos e um acoplador de eixos flexível.

4.3.2. Desenvolvimento do Sistema de Elevação de Carga

O desenvolvimento do sistema iniciou pelo dimensionamento do tambor, tendo em conta que a finalidade era enrolar um cabo de 2 metros. Para assegurar que o cabo fosse enrolado de forma eficaz, foi necessário implementar um sistema de guia para o tambor. Isto significa que à medida que o cabo fosse enrolado, o tambor avançava para evitar sobreposição do cabo e garantir que não se enrolasse sobre si mesmo.

Considerando um diâmetro do tambor de 50 mm, o seu perímetro será

$$\text{Perímetro}_{\text{tambor}} = 2 \pi r^2 = 157,08 \text{ mm} \quad (4.53)$$

Posto isto, como o cabo tem 2 metros é possível determinar o número de rotações necessárias para enrolar completamente o cabo no tambor,

$$N^{\circ} \text{ rotações} = \frac{2000}{157,08} = 13 \quad (4.54)$$

Supondo que são necessárias no máximo 13 rotações, e como o diâmetro do cabo é de 2,5 mm, é possível calcular o comprimento do tambor.

$$\text{Comprimento}_{\text{tambor}} = 13 \times 2,5 = 32,5 \text{ mm} \quad (4.55)$$

Assim, o comprimento considerado para o tambor foi de 33 mm.

Além disso, foi importante adicionar abas com diâmetro de 58 mm nas extremidades do tambor para conter o cabo e desenhar um furo de 8 mm para o veio. Neste furo foi previsto um escatel para que o tambor rode em conjunto com o seu veio (Figura 4.10).

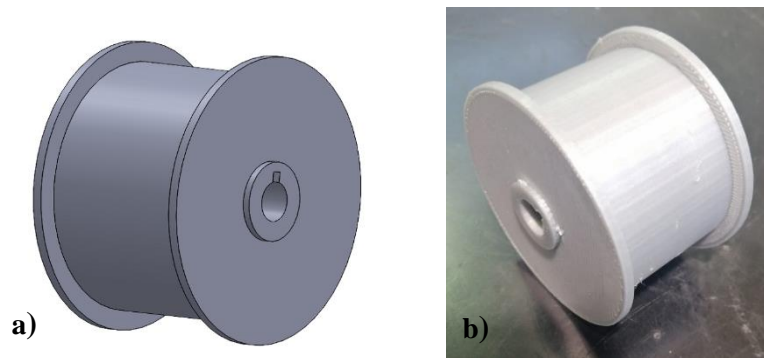


Figura 4.10 - a) Desenho 3D do tambor b) Impressão em plástico do tambor

De seguida, foi necessário criar uma estrutura onde seria encaixado o tambor (Figura 4.11), para que este, para além de rodar sobre si mesmo, também tivesse um movimento horizontal.

Nesta estrutura foi projetado um furo para colocar uma porca com rosca de 8 mm, a fim de assegurar que este suporte se desloque ao longo de um veio roscado com o mesmo diâmetro. Além disso, foram adicionados dois furos de 6 mm paralelos para colocar dois veios, proporcionado assim mais suporte e um melhor movimento a esta estrutura.

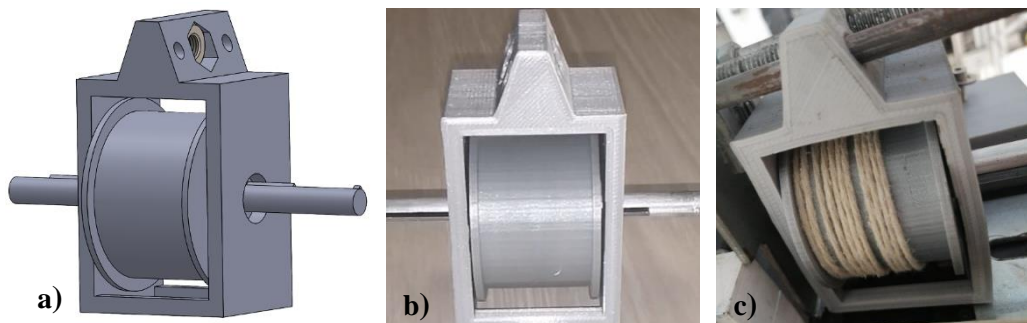


Figura 4.11 - a) Desenho 3D suporte do tambor b), c) Resultado do suporte do tambor

Uma vez que o motor iria ser colocado na extremidade do veio roscado de 8 mm , teria de ser criado um sistema de transmissão entre o motor e o veio de 8 mm do tambor. Para isso foram dimensionadas duas engrenagens, uma para encaixar no veio roscado de 8 mm , onde se movimenta o suporte do tambor horizontalmente, e outra para o veio de 8 mm que passa no interior do tambor, para fazer rodar este em torno de si próprio.

Primeiramente, foi importante determinar o fator de transmissão entre as engrenagens. Visto que o veio roscado M8 possui um passo de $1,25\text{ mm}$ [43], e que o diâmetro do cabo é de $2,5\text{ mm}$, era desejável que após uma rotação completa, iniciasse uma nova a cada $2,5\text{ mm}$. Desta forma, o fator entre as duas engrenagens é de $\frac{2,25}{1,25} = 2$. O que significa que se o tambor se desloca $1,25\text{ mm}$ numa rotação, então em 2 rotações desloca-se $2,5\text{ mm}$, que corresponde ao diâmetro do cabo. Portanto, enquanto o tambor completa 1 rotação, o parafuso deve dar 2 rotações. Consequentemente, para uma engrenagem com 30 dentes, a outra engrenagem deve possuir 15 dentes.

Uma vez que o diâmetro primitivo é o quociente entre o módulo e o número de dentes, e definindo que o módulo é igual a 2, calculam-se os diâmetros primitivos para cada engrenagem.

$$\text{Diâmetro primitivo}_{\text{Engrenagem } 30} = 60\text{ mm} \quad (4.56)$$

$$\text{Diâmetro primitivo}_{\text{Engrenagem } 15} = 30\text{ mm} \quad (4.57)$$

O diâmetro primitivo é o diâmetro imaginário que passa pelos pontos de contacto entre duas engrenagens que estão em contacto e a transmitir movimento, sendo um parâmetro essencial para determinar as outras medidas das engrenagens.

Desta maneira, foi possível determinar o comprimento entre os centros das duas engrenagens, cujo valor é de $\frac{60+30}{2} = 45\text{ mm}$. Consequentemente, o sistema de elevação foi dimensionado tendo em conta essa medida.

Nas Tabela 4.2 e Tabela 4.3 é possível observar as características de ambas as engrenagens, e nas Figura 4.12 e Figura 4.13 apresentam-se os desenhos 3D e impressão 3D das mesmas.

Tabela 4.2 - Características da engrenagem do veio M8 do tambor

Engrenagem 30	
Módulo	2
Número de dentes	30
Espessura, mm	5
Diâmetro primitivo, mm	80
Diâmetro nominal do eixo, mm	8

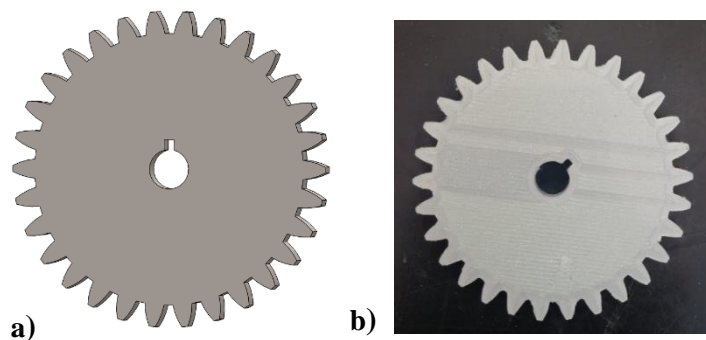


Figura 4.12 - a) Desenho 3D engrenagem 30 dentes b) Impressão da engrenagem 30 dentes

O diâmetro nominal do eixo é o diâmetro no qual a engrenagem é montada, e que define o tamanho do furo central da engrenagem para encaixar no eixo de acionamento.

É importante destacar que o veio roscado de 8 mm ligado ao motor precisou de passar pelo processo de torneamento numa das suas extremidades para desgastar a rosca do veio. Para criar um escatel e adicionar uma chaveta, de modo a garantir a fixação entre o veio e a engrenagem. Por isso, o veio nominal foi considerado de 6 mm para a engrenagem de 15 dentes. Além disso, nesta engrenagem houve ainda a necessidade de ser adicionado um freio.

Tabela 4.3 - Características da engrenagem do veio roscado M8

Engrenagem 15	
Módulo	2
Número de dentes	15
Espessura, mm	5
Diâmetro primitivo, mm	30
Diâmetro nominal do eixo, mm	6

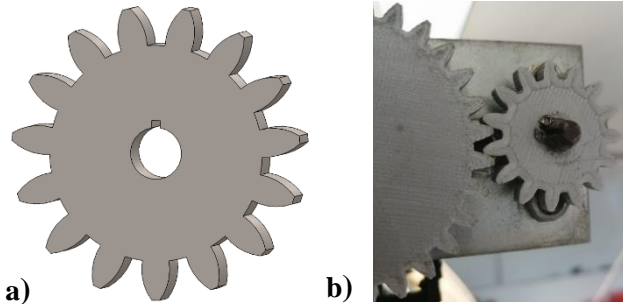


Figura 4.13 - a) Desenho 3D engrenagem 15 dentes b) Impressão da engrenagem 15 dentes

De forma que este conjunto fique fixado no carro, foram adicionadas duas chapas, onde ficam apoiados e fixados os quatro veios, como demonstrado na Figura 4.14.

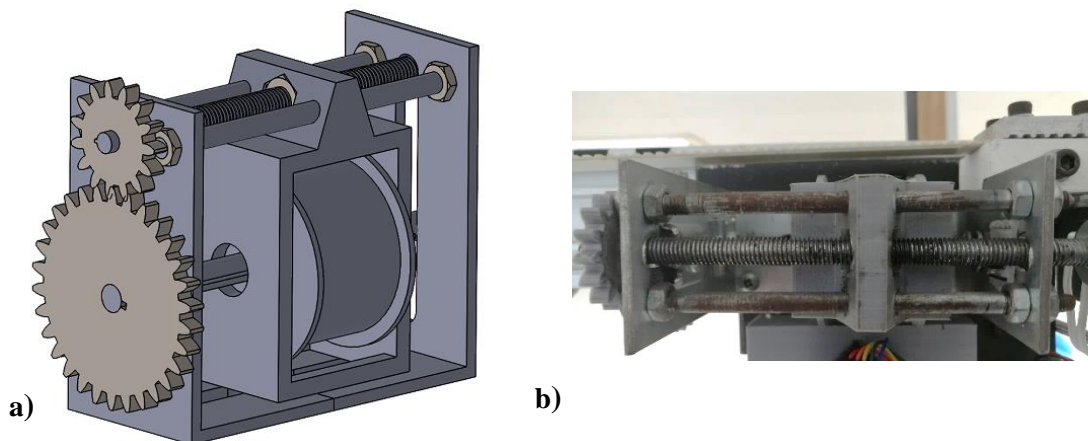


Figura 4.14 - a) Desenho 3D conjunto b) Resultado da vista de cima do conjunto

Para fixar os veios M6 nas chapas de suporte foi necessário abrir rosca nos extremos dos veios para acomodar as porcas M6.

Após isto, seguiu-se para o dimensionamento da estrutura fixadora do motor, onde foi utilizada uma chapa igual às anteriores e onde foi necessário fazer furos que correspondessem com os furos de fixação do motor.

De modo a garantir a transmissão entre o motor e o parafuso roscado M8 foi utilizado um acoplador flexível, como é possível observar pela Figura 4.15. Utilizou-se um acoplador de eixo de 5mm/8mm, com o propósito de interligar mecanicamente o motor DC ao veio M8 que passa pelo suporte do tambor e pela engrenagem de 15 dentes. Este permite um acoplamento flexível do eixo absorvendo eventuais excentricidades. Estes diâmetros correspondem com o diâmetro do veio do motor e o diâmetro do veio roscado, respetivamente. Ainda, o diâmetro exterior do acoplador é de 19 mm e comprimento de 25 mm.

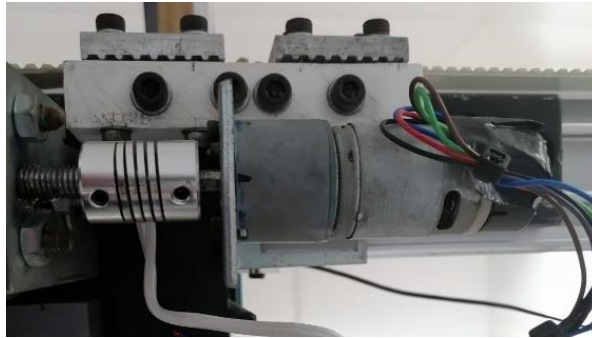


Figura 4.15 - Montagem do acoplador

O último conjunto projetado corresponde à estrutura onde foi posicionado o *encoder*. Este conjunto é constituído pela peça pelo qual o cabo atravessa, e que está conectada ao *encoder* (Figura 4.16), e pelas peças de fixação (Figura 4.17 e Figura 4.18), que como o nome indica, fixam a primeira peça ao carro.

A peça por onde passa o cabo serve para auxiliar na leitura do ângulo de oscilação deste. Num dos seus extremos é colocado um ímã, que por sua vez está magneticamente conectado ao *encoder*, que realiza as medições das oscilações da peça (e conseqüentemente as oscilações do cabo).

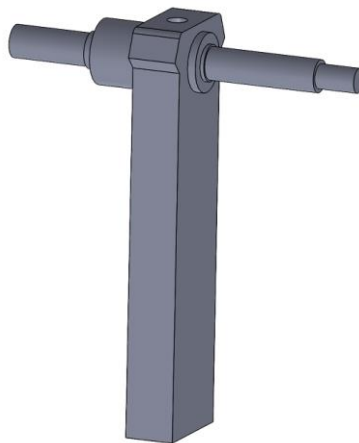


Figura 4.16 - Desenho 3D da peça para leitura do ângulo do cabo

Na peça de fixação 1 foi necessário realizar um furo na parte superior para que o cabo atravessasse vindo do tambor, que está posicionado imediatamente acima desta peça.

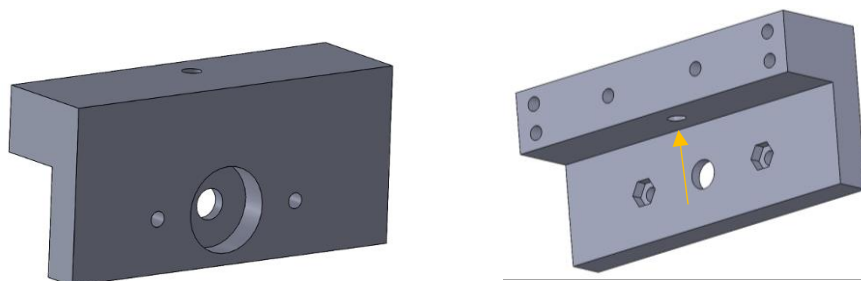


Figura 4.17 - Desenho 3D da peça de fixação 1

De modo que a peça conectada ao *encoder* tivesse apoio em ambos os lados, foi adicionada a peça de fixação 2. Esta peça pertencia ao equipamento pré-existente, tendo sido aproveitada.

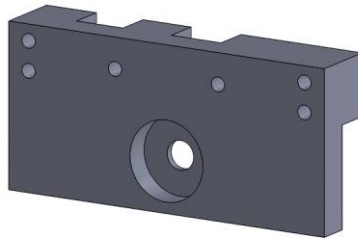


Figura 4.18 - Desenho 3D da peça fixação 2

Em ambas as peças de fixação encontram-se rolamentos para garantir a rotação do eixo, reduzindo o atrito.

Por fim, foi realizada a montagem do *encoder*, assim como as ligações elétricas necessárias (Figura 4.19).

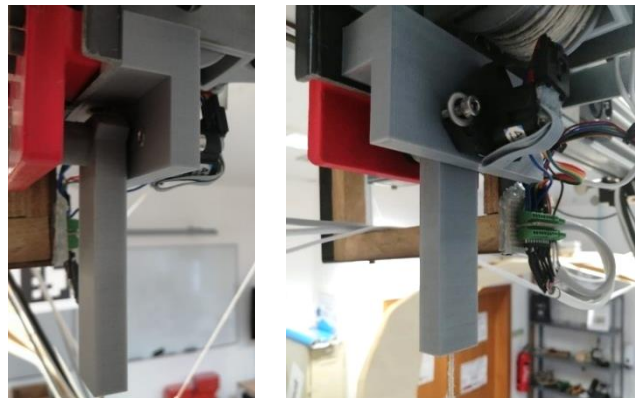


Figura 4.19 - Montagem do encoder

Na Figura 4.20 é apresentado o equipamento concebido depois de aplicado no LRAFI da ESTG.



Figura 4.20 - Resultado do projeto mecânico desenvolvido

4.4. Hardware de Controlo

Após o desenvolvimento e montagem do projeto mecânico, avançou-se para a construção do *hardware* de controlo do sistema. Neste incluem-se os sensores e atuadores seleccionados, o método de controlo do sistema e o processo de aquisição e tratamento de dados (Figura 4.21).

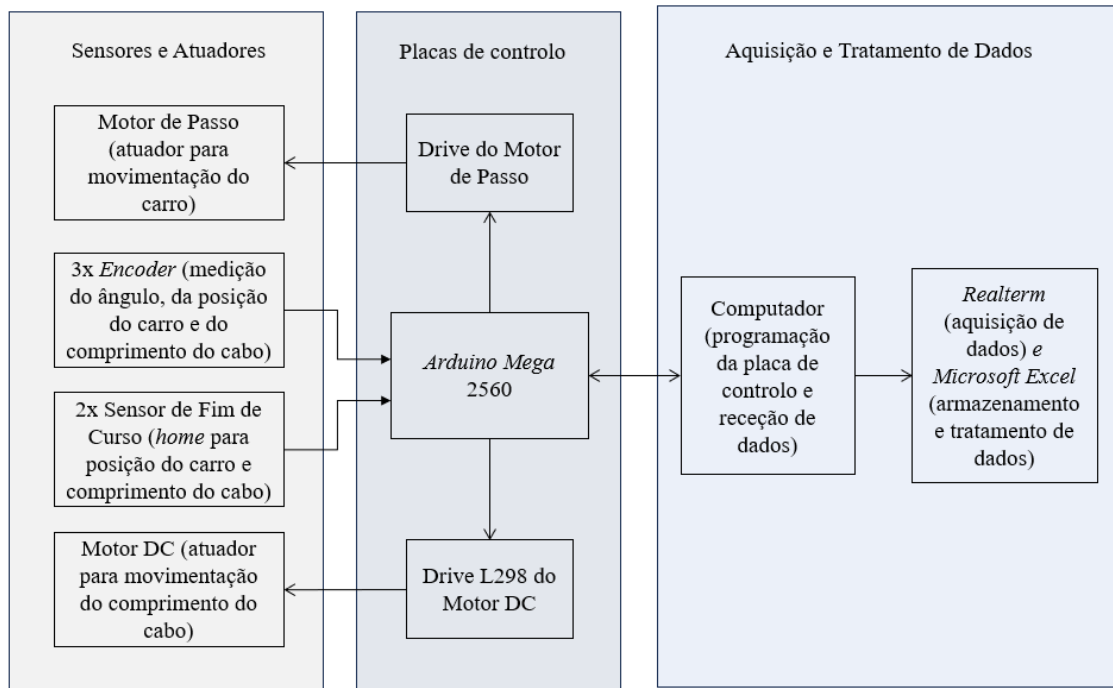


Figura 4.21 - Arquitetura do sistema de controlo do equipamento

4.5.1. Atuadores e Sensores

O motor seleccionado para realizar o movimento do cabo foi o EMG30, um motor de 12 V, equipado com *encoder* incremental (Tabela 4.4 e Figura 4.22) e um redutor de velocidade.

Tabela 4.4 - Especificações do motor EMG30

Parâmetro	Valor
Tensão nominal	12 V
Torque nominal	1,5 kg.cm
Velocidade nominal	170 rpm
Corrente nominal	530 mA
Contagens do <i>encoder</i> por volta do eixo de saída	360
Velocidade mínima	1,5 rpm
Velocidade máxima	200 rpm



Figura 4.22 - Motor EMG30 para controlo do comprimento do cabo

Este motor possui 6 conexões apresentadas na Tabela 4.5 [45]:

Tabela 4.5 - Conexões do motor EMG30

Cor do fio	Conexão
Roxo	Sensor Hall sinal de saída B
Azul	Sensor Hall sinal de saída A
Verde	Sensor Hall GND
Castanho	Sensor Hall VCC
Vermelho	Positivo do Motor
Preto	Negativo do Motor

De modo a controlar o motor DC, responsável por variar o comprimento do cabo, foi necessário incorporar um *driver* para esse efeito.

O módulo L298 (Tabela 4.6 e Figura 4.23) permite controlar até dois motores DC, que possuam tensões de funcionamento entre 5V e 35V.

Tabela 4.6 - Especificações do L298

Parâmetro	Valor
Tensão Operacional	5V – 46V
Corrente Operacional	2A
Tensão Lógica	5V
Corrente Lógica	0-36mA
Potência Máxima (W)	25W

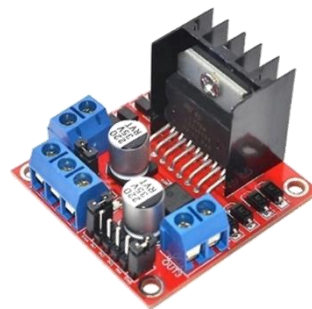


Figura 4.23 - L298

Este *driver* permite controlar a velocidade do motor através do sinal *Pulse Width Modulation* ou PWM, e a direção do motor através da ponte H.

O sinal PWM ajusta o valor médio da tensão, ligando e desligando a energia num ritmo rápido (série de pulsos ON-OFF). A tensão média depende da quantidade de tempo em que o sinal fica ligado versus a quantidade de tempo que o sinal fica desligado, num determinado período de tempo. A tensão média é proporcional à largura dos pulsos, esses pulsos são conhecidos como *Duty Cycle*. Assim, conectou-se a saída PWM do Arduino ao pino *Enable* do driver L298.

Relativamente à ponte H, esta permite controlar o sentido de rotação, invertendo o fluxo da corrente, quando ativados dois interruptores específicos à mesma ponte.

O L298 é constituído pelos pinos de alimentação: pino GND e terminal VCC. O pino GND, que corresponde ao pino de referência, é conectado ao terminal negativo (0V) da fonte de alimentação. O terminal VCC é usado para fornecer energia ao motor, sendo, neste caso ligado ao terminal +12V da fonte. As entradas de controlo lógico correspondem aos pinos *Enable A* e *Enable B*, que são utilizados para controlar a velocidade do motor A e B, respetivamente. Os pinos de entrada IN1, IN2, IN3 e IN4 controlam os interruptores do circuito da ponte H. Os pinos de entrada 1 e entrada 2 são usados para controlar o sentido de rotação do motor A, e as entradas 3 e 4 para o motor B. Se a entrada 1 for LOW e a entrada 2 for HIGH o motor avança, e vice-versa, se a entrada 1 for HIGH e a entrada 2 for LOW o motor move-se no sentido contrário. Caso as entradas sejam iguais, ou LOW ou HIGH o motor pára. Existem também os pinos de saída, OUT1 e OUT2, e OUT3 e OUT4, que correspondem aos blocos que fornecem a saída aos motores A e B, respetivamente [44].

No caso deste sistema foi apenas utilizado um motor DC, logo só foram usados os pinos de um canal.

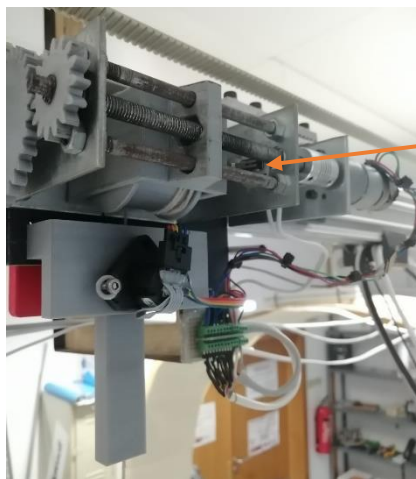
Na Figura 4.24 está apresentada a montagem do driver L298 numa placa de circuito.



Figura 4.24 - Montagem do driver L298

Foi notada a necessidade de acrescentar um sensor para conhecer a posição do cabo (Figura 4.25) quando o sistema é iniciado, uma vez que o *encoder* do motor é incremental e não guarda a medida de comprimento do cabo. Assim, com um sensor de fim de curso, é possível definir um valor de referência no arranque do equipamento.

Um sensor de fim de curso, também denominado *microswitch*, é acionado quando é submetido a uma força física pequena. A sua função principal indicar que uma posição específica foi alcançada. Desta forma, quando o suporte do tambor atinge o sensor significa que o comprimento do cabo está na posição de referência inicial.



Sensor de fim de curso para o comprimento do cabo

Figura 4.25 - Implementação do sensor de fim de curso para o comprimento do cabo

Como já foi mencionado anteriormente, o equipamento possui um *encoder* absoluto e magnético para medir a posição angular do cabo. O *encoder* utilizado foi o AEAT-6012-A06 (Tabela 4.7 e Figura 4.26) [46].

Tabela 4.7 - Especificações do encoder AEAT-6012-A06

Parâmetro	Valor
Tipo	Absoluto
Tensão de alimentação	+5 V
Frequência máxima	1 MHz
Resolução	12 bits



Figura 4.26 - Encoder absoluto AEAT-6012-A06

4.5.2. Placa de Controle

Para recolher, processar e transmitir dados, e controlar o sistema foi utilizada a placa *Arduino Mega 2560* (Tabela 4.8 e Figura 4.27).

Tabela 4.8 - Características do Arduino Mega 2560.

Parâmetro	Valor
Tensão de alimentação	7-12 V _{DC}
Tensão de funcionamento	5 V _{DC}
Nº de entradas/saídas digitais	54 dos quais 15 podem fornecer saída PWM
Nº de pinos para interrupções	6
Nº de entradas analógicas	16
Resolução ADC	10 bits

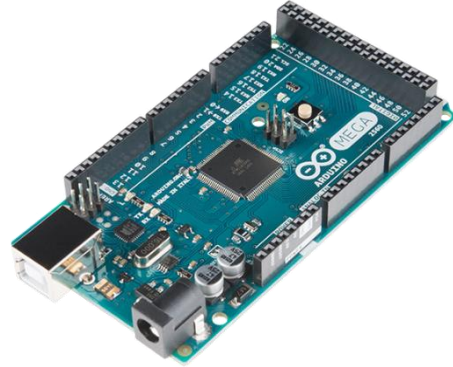


Figura 4.27 - Arduino Mega 2560.

Todos os componentes empregados para controlar o sistema estão ligados à placa de controle. Desta forma, foi elaborado e implementado um esquema de ligações elétricas, detalhado no Apêndice E.

4.5. Software de Controle

4.5.1. Controle do Equipamento e Transmissão de Dados

Procedeu-se ao desenvolvimento do programa de controlo do equipamento através do microcontrolador do *Arduino Mega 2560*. Tendo em conta a arquitetura do sistema e o esquema de controlo, foi estruturado um programa através do *software Arduino IDE* estando exposto no Apêndice G.

De forma geral, o sistema opera conforme ilustrado na Figura 4.28. Existem cinco etapas principais. Primeiramente, o sistema é inicializado quando está pronto para funcionamento, ou seja, com as fontes de alimentação ativas e sem ativação do botão de emergência. Quando os parâmetros desejados estiverem de acordo com o que está programado no microcontrolador, é ligada a porta série do *Arduino*, e ocorre a rotina de referência dos atuadores. Esta rotina é constituída por um conjunto de instruções que posiciona o carro na posição inicial e ajusta o comprimento inicial do cabo, preparando o sistema para realizar um movimento. Quando a carga estiver estabilizada, o utilizador seleciona o movimento desejado inserindo o comando correspondente, no monitor série do *Arduino* ou num teclado de comando exterior, e o sistema executa esse movimento. Este é concluído quando a posição desejada, pré-definida no programa, é alcançada. Caso seja pretendido que o sistema realize outro movimento, o utilizador deve inserir o comando no qual o sistema retorna à sua posição inicial (denominado *home*).

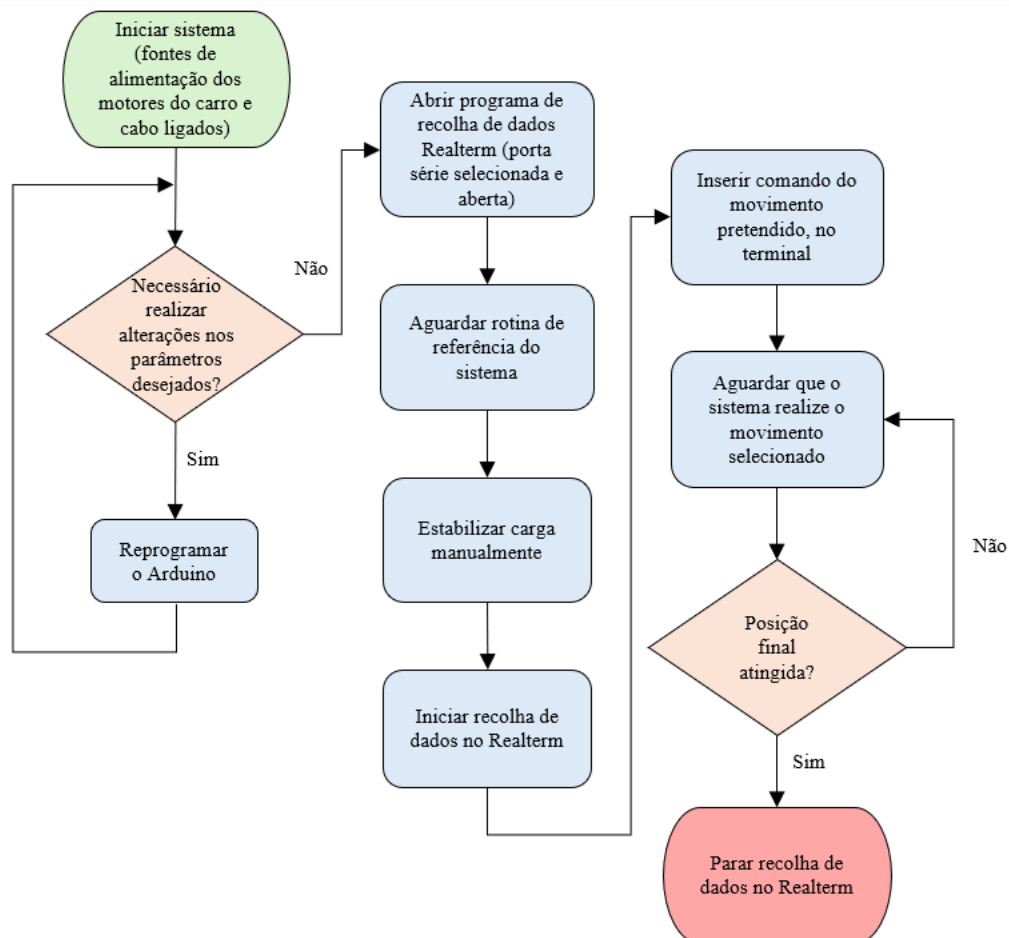


Figura 4.28 - Fluxograma do funcionamento geral do sistema para recolha de dados

Uma vez que o equipamento pré-existente apenas estudava o controlo da posição do carro e do ângulo de oscilação de carga, o programa anterior implementado no *Arduino* só tinha em consideração essas duas variáveis. Após as modificações realizadas no equipamento foi adicionado ao programa a variável de saída do comprimento do cabo. Portanto, existiu a necessidade de acrescentar a entrada de controlo da velocidade do motor, que permite a variação do valor do comprimento do cabo. Além disso, foi importante adicionar uma interrupção para a programação do *encoder*, presente no motor DC. Quando a interrupção é chamada, é gerado o valor do comprimento do cabo, pois quando o motor roda para um lado, é feito um incremento do valor e, se rodar para outro diminui o valor. Relativamente aos subprogramas que permitem a automação de procedimentos iniciais e de emergência, destaca-se o subprograma que referencia os atuadores do sistema, atribuindo as respetivas posições iniciais, no qual foi acrescentado o mesmo para o comprimento do cabo, a partir da leitura fornecida pelo sensor de fim de curso.

Adicionalmente ao programa, destaca-se a elaboração de vários cenários de controlo do equipamento, apresentados no esquema da Figura 4.29.

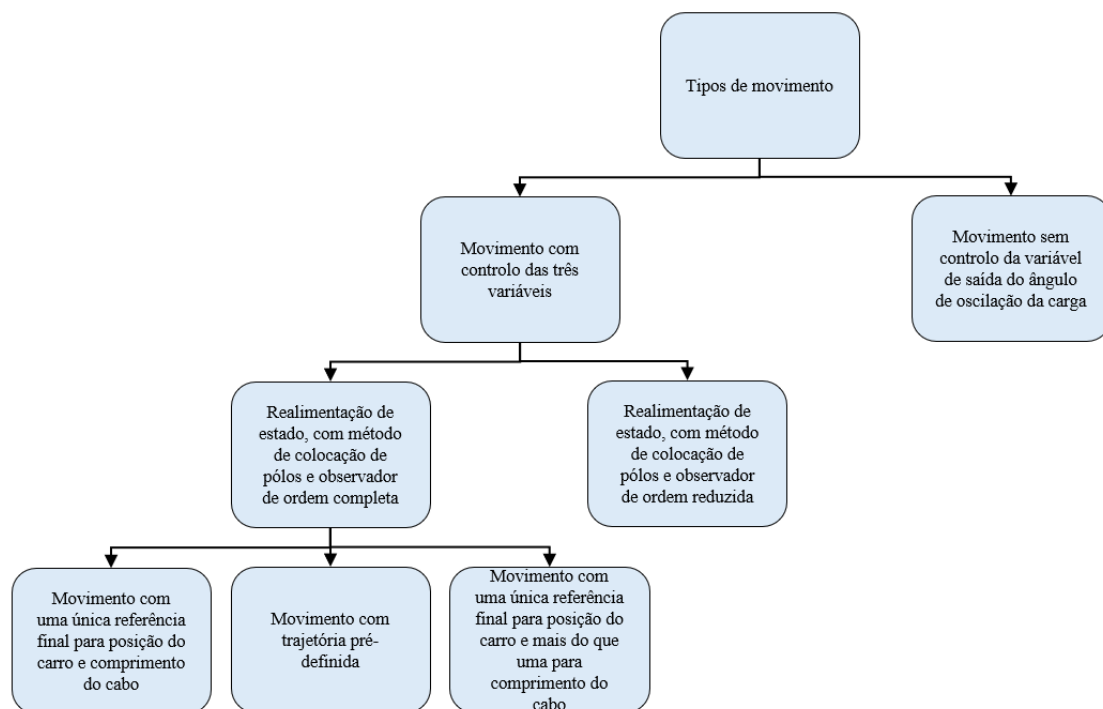


Figura 4.29 - Tipos de movimento programados no Arduino

Um dos cenários previamente existente era o movimento sem controlo, no qual foi adicionado a variação do comprimento do cabo, sendo definida uma variável de erro entre o valor medido do comprimento do cabo e a medida final desejada para o seu comprimento, permitindo ao sistema saber quando atingiu o pretendido. Neste movimento é apenas dada uma referência final tanto à posição do carro como ao comprimento do cabo, sem qualquer controlo do ângulo de oscilação da carga suspensa.

Para controlar o sistema foi implementado o controlador com realimentação de estados com observador de ordem completa. Neste controlador as variáveis de estado do sistema são estimadas de acordo com a expressão (3.33), exceto o comprimento do cabo, que é medido diretamente (pois para este método de controlo o comprimento do cabo foi considerado um sistema desacoplado). De seguida, foi programado o controlador de realimentação de estados para determinar as entradas do motor do carro e do motor do cabo, tendo em conta a equação (3.26). Este sistema de controlo é utilizado para vários cenários, de acordo com a Figura 4.29.

Além disso, foi aplicado o observador de ordem reduzida. Neste são utilizados os valores medidos diretamente do ângulo, da posição e do comprimento do cabo no cálculo das entradas do sistema.

Outro cenário abordado foi a definição de uma trajetória para a carga suspensa. Neste programa foi desenvolvida uma função, que para cada trecho da trajetória é determinado o valor de referência no eixo das abscissas e no eixo das ordenadas, que correspondem à posição do carro e ao comprimento do cabo, respetivamente. Esses valores são considerados no controlo, que determina as entradas do sistema que melhor se adequam.

Para a situação que é considerando um sistema linear por treços foram determinadas, através do MATLAB, e de acordo com as classes definidas para o comprimento do cabo, a matriz de ganho K , a matriz do observador L , a matriz de estado A e a matriz de entrada B . Desta forma, conforme o valor medido do comprimento do cabo, o programa vai procurar os valores às matrizes resultantes, os quais são posteriormente utilizados no controlo.

Para além disso, foi programada uma janela, que permite ao utilizador, recorrendo ao monitor da porta série do *Arduino* seleccionar se pretende inserir uma determinada referência para a posição do carro ou para o comprimento do cabo. Essa referência é guardada e considerada no movimento que o utilizador escolher de seguida.

No Apêndice F está exposto um *grafcet* de nível 1 para mostrar o funcionamento geral do *software*.

Ao longo dos testes realizados através do programa implementado houve a necessidade da correção de aspetos que são apresentados de seguida.

Correção da leitura de ângulo de oscilação de carga

Foi verificado que a leitura do ângulo pelo *encoder* variava conforme a posição do carro, mesmo que fosse observável que o ângulo da carga era de 0° face à vertical gravítica. Estes erros foram identificados como sendo originados por variações no nivelamento e no torsão das diversas secções que constituem a calha guia do carro. Foram recolhidos vários valores de leitura do ângulo, de acordo com a posição do carro, e a partir desses dados foram determinadas funções polinomiais para corrigir os valores de ângulo obtidos experimentalmente. Na Figura 4.30 apresenta-se a comparação entre o valor medido pelo *encoder* e a posição do carro.

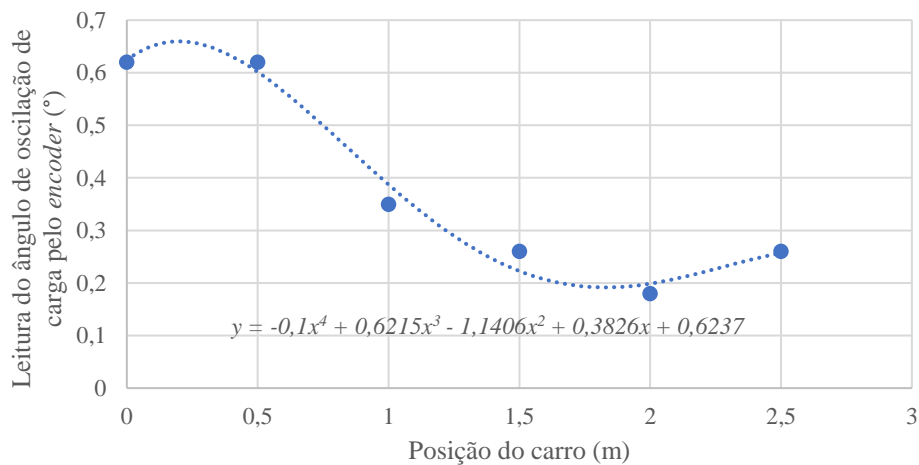


Figura 4.30 - Relação entre o valor medido do ângulo de oscilação de carga pelo *encoder* (°) e a posição do carro no carril (m)

Correção da velocidade de subida e descida do cabo

Verificou-se que a velocidade de subida e descida do cabo eram diferentes para os mesmos valores do comando e que, apesar deste valor poder ser indicado pelo microcontrolador na gama de 0-255, o motor não possui a força nas zonas mais baixas desta gama de velocidades para subir ou descer. Desta forma, foi necessário determinar as velocidades mínimas e máximas para ambas as direções, e definir os limites. Foram realizados testes para verificar a velocidade real do motor em ambas as direções, e obtiveram-se os resultados demonstrados na Figura 4.31.

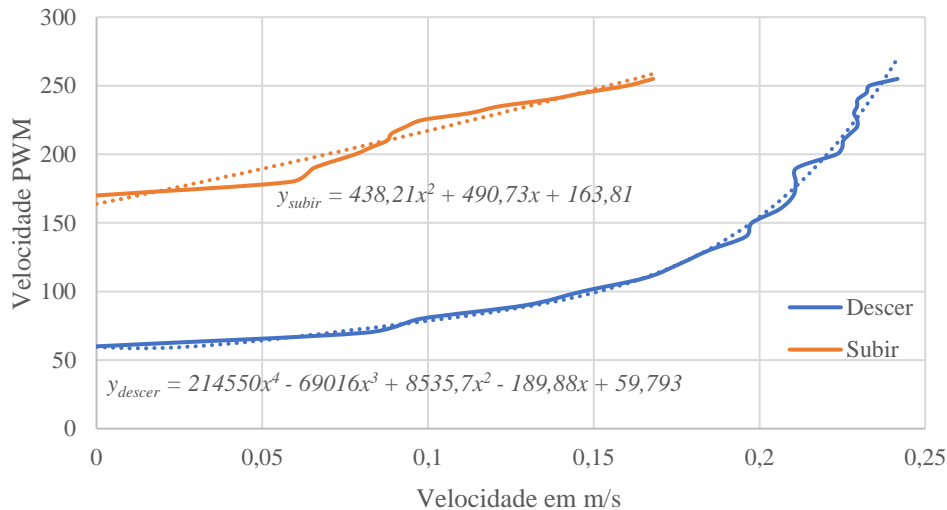


Figura 4.31 - Comparação da velocidade em m/s e a velocidade PWM do motor, na subida e descida da carga

As expressões apresentadas no gráfico correspondem às aproximações polinomiais realizadas para converter a velocidade de entrada do motor PWM em velocidade *m/seg*.

Desta forma, quando no sistema de controlo é determinada a entrada do valor do motor responsável pelo movimento do cabo e do motor responsável pelo movimento do carro, estas são avaliados para calcular os valores de ativação a fornecer ao gerador de PWM dos motores, a partir das funções e limites determinados.

4.5.2. Aquisição e Tratamento de Dados no Computador

Com o objetivo de adquirir e armazenar os dados recebidos no computador por porta série, provenientes da placa *Arduino*, foi utilizado o *software* REALTERM. Neste deve ser selecionada a porta série e a *baud rate*, que devem de estar de acordo com o que está no código desenvolvido. De seguida, são extraídos e guardados em ficheiro de texto, *.txt*, enquanto o sistema está em funcionamento. De seguida, através do *Microsoft Excel*, os dados são transformados em tabela para o seu tratamento.

4.6. Sistemas de Controle

4.6.1. Arquitetura do Sistema de Controle

O sistema de controle para movimentação da carga suspensa, conforme ilustrado na Figura 4.32, emprega um controlador que regula três variáveis principais: o ângulo de oscilação da carga, a posição do carro e o comprimento do cabo. Posteriormente, as saídas do sistema são medidas por sensores e somadas para obter um valor ajustado do sinal de controle.

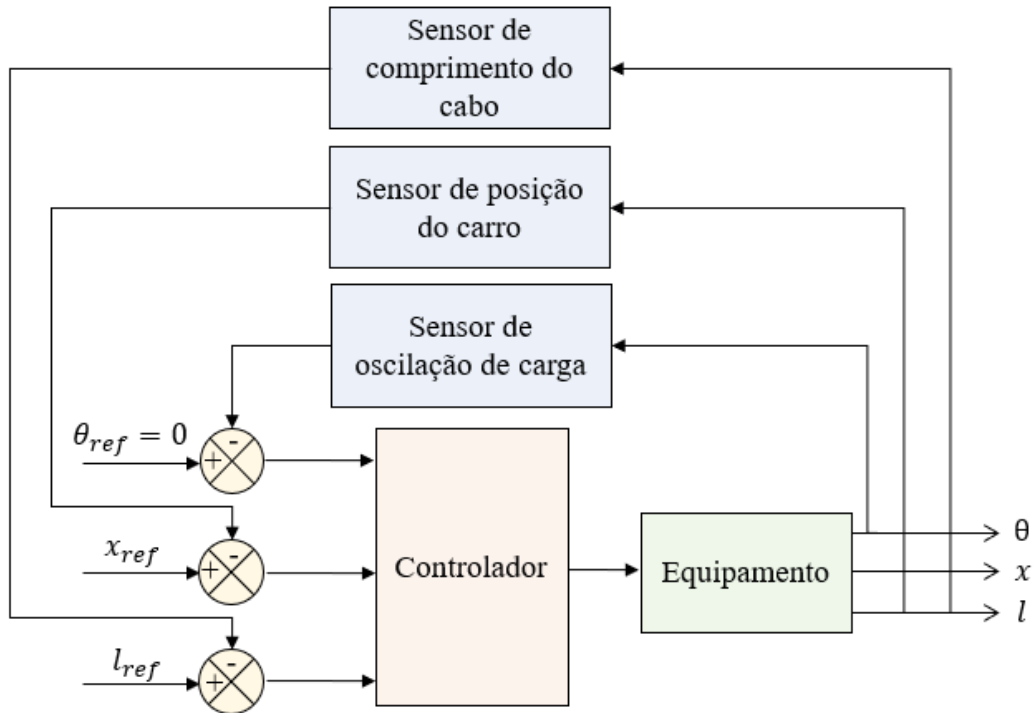


Figura 4.32 - Esquema do sistema de controle

Relativamente à posição do carro e ao comprimento do carro, a entrada corresponde ao sinal de erro entre o valor de referência e a medição obtida por meio de um sensor. No caso do controlador de oscilação do ângulo da carga, ele comporta-se como um regulador, uma vez que a referência é 0° . Já nos outros dois casos, a entrada é um sinal em degrau.

4.6.2. Adaptação do Modelo com Entrada em Velocidade

Na modelação inicial do sistema, as suas entradas eram forças, F_x e F_l . No entanto, como a entrada de comando dos *drivers* dos motores é feita em velocidade, houve a necessidade de reformular o modelo do sistema com entrada em velocidade, ou seja, a velocidade dada ao carro e ao cabo.

A partir do modelo linear obtido no subcapítulo 4.2.1., foi expresso o modelo na forma de um sistema de equações diferenciais,

$$\left\{ \begin{array}{l} \dot{x} = \dot{x} \\ \ddot{x} = \frac{-D_x}{M} \dot{x} - \frac{mg}{M} \theta + \frac{D_\theta}{Ml_0} \dot{\theta} + \frac{1}{M} F_c \\ \dot{\theta} = \dot{\theta} \\ \ddot{\theta} = \frac{D_x}{Ml_0} \dot{x} - \frac{g}{l_0} \theta - \frac{D_\theta(M+m)}{mMl_0^2} \dot{\theta} + \frac{-1}{Ml_0} F_c \\ \dot{l} = \dot{l} \\ \ddot{l} = \frac{-D_l}{m} \dot{l} + \frac{1}{m} F_l \end{array} \right. \quad (4.58)$$

Visto que o modelo é descrito por um conjunto de equações diferenciais de segundo grau, é necessário converter as equações.

Analisando o sistema, é possível reescrever a segunda equação para determinar a força exercida pelo motor em função da velocidade linear e do ângulo do pêndulo, resultando na equação seguinte:

$$\frac{1}{M} F_c = \ddot{x} + \frac{D_x}{M} \dot{x} + \frac{mg}{M} \theta - \frac{D_\theta}{Ml_0} \dot{\theta} \quad (4.59)$$

Substituindo na quarta equação do sistema de equações em (4.58), resulta em:

$$\ddot{\theta} = \frac{D_x}{Ml_0} \dot{x} - \frac{g}{l_0} \theta - \frac{D_\theta(M+m)}{mMl_0^2} \dot{\theta} - \frac{1}{l_0} \left(\ddot{x} + \frac{D_x}{M} \dot{x} + \frac{mg}{M} \theta - \frac{D_\theta}{Ml_0} \dot{\theta} \right) \quad (4.60)$$

$$\Leftrightarrow \ddot{\theta} = \frac{D_x}{Ml_0} v_x - \frac{g}{l_0} \theta - \frac{D_\theta(M+m)}{mMl_0^2} \dot{\theta} - \frac{1}{l_0} v_x - \frac{D_x}{Ml_0} v_x - \frac{mg}{Ml_0} \theta + \frac{D_\theta}{Ml_0^2} \dot{\theta} \quad (4.61)$$

$$\Leftrightarrow \ddot{\theta} = \left(-\frac{D_\theta(M+m)}{mMl_0^2} + \frac{D_\theta}{Ml_0^2} \right) \dot{\theta} + \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) \theta - \frac{1}{l_0} v_x \quad (4.62)$$

Como a equação obtida possui derivadas no sinal de entrada, então é necessário determinar novas variáveis de estado para que a entrada seja a velocidade. Desta forma, recorreu-se a um método de transformação de representação de sistema em espaço de estado sugerida por Vaccaro [39].

Primeiramente, é integrada a equação diferencial, até obter uma variável de saída no lado esquerdo da mesma [39],

$$\iint \ddot{\theta} = \left(-\frac{D_\theta(M+m)}{mMl_0^2} + \frac{D_\theta}{Ml_0^2} \right) \iint \dot{\theta} + \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) \iint \theta - \frac{1}{l_0} \iint v_x \quad (4.63)$$

ou seja,

$$\theta = \left(-\frac{D_\theta(M+m)}{mMl_0^2} + \frac{D_\theta}{Ml_0^2} \right) \int \theta + \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) \iint \theta - \frac{1}{l_0} \int v_x \quad (4.64)$$

Assim, define-se θ como a primeira variável de estado X_1 , ou seja $\dot{X}_1 = \dot{\theta}$ e $X_1 = \theta$.

Derivando a equação (4.64) obtém-se:

$$\dot{X}_1 = \left(-\frac{D_\theta(M+m)}{mMl_0^2} + \frac{D_\theta}{Ml_0^2} \right) \theta + \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) \int \theta - \frac{1}{l_0} v_x \quad (4.65)$$

Definindo a segunda variável de estado X_2 por:

$$X_2 = \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) \int \theta \quad (4.66)$$

Obtém-se a primeira equação de estado do modelo:

$$\dot{X}_1 = \left(-\frac{D_\theta(M+m)}{mMl_0^2} + \frac{D_\theta}{Ml_0^2} \right) X_1 + X_2 - \frac{1}{l_0} v \quad (4.67)$$

Derivando X_2 obtém-se a segunda equação de estado,

$$\dot{X}_2 = \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) \theta = \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) X_1 \quad (4.68)$$

De seguida, caracteriza-se a terceira variável de estado X_3 , que corresponde à terceira equação de estado:

$$\dot{X}_3 = \dot{x} = v_x \quad (4.69)$$

Esta variável de estado diz respeito à derivada da posição e permite seguir a posição linear do carro.

Considera-se a quarta variável de estado X_4 e sua derivada como:

$$\dot{X}_4 = \dot{l} = v_l \quad (4.70)$$

Esta variável de estado equivale à derivada do comprimento do cabo. E representa a quarta equação de estado.

Definiu-se que as entradas do modelo serão a velocidade v_x e v_l .

$$V = [v_x \ v_l]^T \quad (4.71)$$

E as saídas do modelo como o ângulo de oscilação de carga, X_1 , a posição do carro, X_3 , e o comprimento do cabo, X_4 . Desta forma, é possível escrever um novo modelo linear do sistema em espaço de estado.

Apresenta-se a equação de estado do novo modelo,

$$\dot{X} = AX + BV \quad (4.72)$$

ou seja,

$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \\ \dot{X}_4 \end{bmatrix} = \begin{bmatrix} \left(-\frac{D_\theta(M+m)}{mMl_0^2} + \frac{D_\theta}{Ml_0^2} \right) & 1 & 0 & 0 \\ \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} + \begin{bmatrix} -\frac{1}{l_0} \\ 0 \\ 1 \\ 0 \end{bmatrix} V \quad (4.73)$$

No caso da equação de saída, esta é dada por:

$$Y = CX + D \quad (4.74)$$

$$\begin{bmatrix} x \\ \theta \\ l \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} \quad (4.75)$$

onde

$$D = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.76)$$

4.6.3. Controlabilidade do Sistema

De acordo com o enquadramento teórico referente à definição de controlabilidade exposta, antes de desenvolver um sistema de controlo, foi avaliada a controlabilidade do sistema. Utilizando o MATLAB, por meio do comando $Wc = ctrb(A, B)$ foi possível determinar a matriz de controlabilidade:

$$Wc = \begin{bmatrix} -1,000 & 0,000 & 0,014 & 0,000 & 9,801 & 0,000 & 0,274 & 0,000 \\ 0,000 & 0,000 & 9,801 & 0,000 & -0,137 & 0,000 & -96,05 & 0,000 \\ 1,000 & 0,000 & 0,000 & 0,000 & 0,000 & 0,000 & 0,000 & 0,000 \\ 0,000 & 1,000 & 0,000 & 0,000 & 0,000 & 0,000 & 0,000 & 0,000 \end{bmatrix} \quad (4.77)$$

Através do comando, $rank(Wc)$, foi constatado que a característica da matriz de controlabilidade, Wc , é igual a 4, o que por sua vez, é igual à ordem do sistema em espaço de estados, n . Portanto, verifica-se que a condição de controlabilidade completa é satisfeita. O que significa que é possível afirmar que o sistema é completamente controlável, e através das variáveis de entrada aplicadas, é possível transferir o sistema de um estado inicial para um estado final desejável.

4.6.4. Observabilidade do Sistema

Para verificar a observabilidade do sistema utilizou-se o comando $obsv(A, C)$ para determinar a matriz de observabilidade do sistema.

$$W_o = \begin{bmatrix} 1 & 0 & 1 & 1 \\ -0,014 & 1 & 0 & 0 \\ -9,801 & -0,014 & 0 & 0 \\ 0,2744 & -9,801 & 0 & 0 \end{bmatrix} \quad (4.78)$$

A partir do comando $\text{rank}(W_o)$, descobriu-se que a característica da matriz de observabilidade, W_o , é igual a 3, concluindo que o sistema não é observável.

Desta forma, foi necessário desacoplar o sistema, desenvolvendo duas novas matrizes de estados e de entrada:

$$A_{1,3} = \begin{bmatrix} \left(-\frac{D_\theta(M+m)}{mMl_0^2} + \frac{D_\theta}{Ml_0^2} \right) & 1 & 0 \\ \left(-\frac{g}{l_0} - \frac{mg}{Ml_0} \right) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ e } B_{1,3} = \begin{bmatrix} -\frac{1}{l_0} \\ 0 \\ 1 \end{bmatrix} \quad (4.79)$$

$$A_4 = [0] \text{ e } B_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (4.80)$$

As matrizes de estado, $A_{1,3}$, e de entrada, $B_{1,3}$, dizem respeito as três variáveis de estado do sistema, X_1 , X_2 , e X_3 . Enquanto as outras consideram apenas a variável de estado X_4 .

De seguida, determinou-se a matriz de observabilidade, mas para o sistema constituído pelas três variáveis de estado, X_1 , X_2 , e X_3 , obtendo-se:

$$W_{o_{1,3}} = \begin{bmatrix} 1 & 0 & 1 \\ -0,014 & 1 & 0 \\ -9,801 & -0,014 & 0 \end{bmatrix} \quad (4.81)$$

Por conseguinte, a característica da matriz de observabilidade $W_{o_{1,3}}$ é igual a 3, sendo igual à ordem n do sistema desacoplado. Afirma-se que o sistema é observável para o sistema desacoplado que engloba as três primeiras variáveis de estado, X_1 , X_2 , e X_3 .

4.6.5. Pólos do Sistema

Como já foi mencionado anteriormente, os zeros da equação característica, que correspondem aos pólos do sistema, estão relacionados com a matriz de estado de acordo com a seguinte expressão:

$$|\lambda \cdot I - A| = 0 \quad (4.82)$$

A partir do comando *pole* do MATLAB foi possível determinar os pólos do modelo linear de malha aberta, sendo estes:

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} -0.0070 + 3.1306i \\ -0.0070 - 3.1306i \\ 0 \\ 0 \end{bmatrix} \quad (4.83)$$

Verifica-se que existem dois pólos na origem, λ_3 e λ_4 , pois a sua parte real é igual a zero, e encontram-se na origem do plano complexo. Isto indica que o sistema é do tipo 2 (Figura 4.33), o que significa que o valor do erro estacionário será nulo para uma entrada em degrau. Além disso, observa-se que os pólos λ_1 e λ_2 estão localizados no semiplano esquerdo do plano complexo s , porque a parte real deles é negativa. Assim, qualquer entrada limitada resulta numa saída limitada, e o sistema é considerado estável. Portanto, o sistema é classificado como criticamente estável.

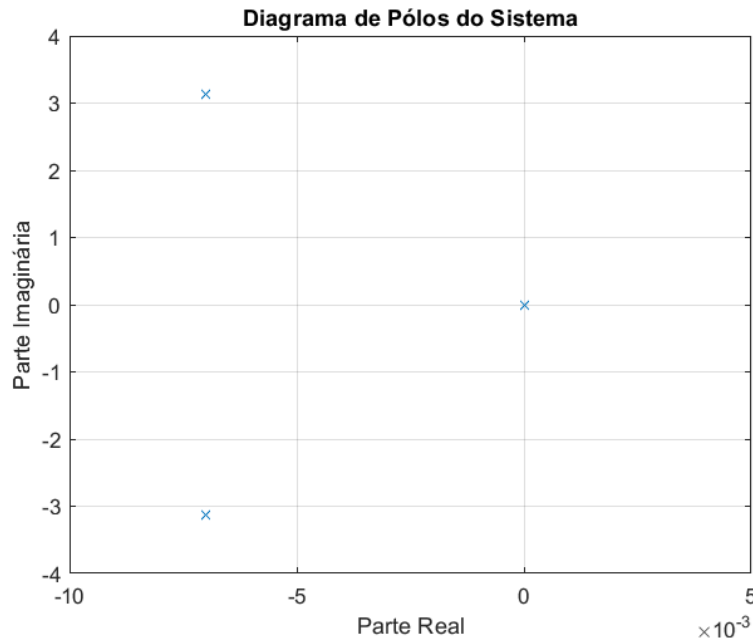


Figura 4.33 - Diagrama de Pólos do Sistema

4.6.6. Simulações de Sistemas de Controle

Para verificar a dinâmica do sistema, e resposta ao sinal de entrada, foram utilizados os *softwares* MATLAB e *Simulink*.

Através do MATLAB foi desenvolvido um programa, exposto no Apêndice B, de modo a simular os métodos de controle do sistema. Neste programa foram inseridos os valores dos parâmetros do modelo matemático, inseridas as matrizes A, B, C e D obtidas para o modelo linear e utilizou-se a função *ss* para gerar o modelo linear em espaço de estado.

De seguida, procedeu-se à simulação do sistema com diferentes controladores através do *Simulink*, sendo possível verificar a resposta do sistema utilizando diferentes parâmetros de controle.

Controlador Proporcional Integral Derivativo, PID

A partir de simulações realizadas, verificou-se que o sistema de controle utilizando o PID necessitaria de 6 controladores. Ou seja, para cada uma das entradas, sendo elas a entrada responsável pelo movimento do carro e a entrada responsável pelo movimento do cabo, são necessários três controladores PID, um para cada uma das três variáveis de interesse, o ângulo de oscilação da carga, a posição do carro e o comprimento do cabo.

O diagrama de blocos para este controlador está representado na Figura 4.34. Cada controlador é representado por um bloco “PID”.

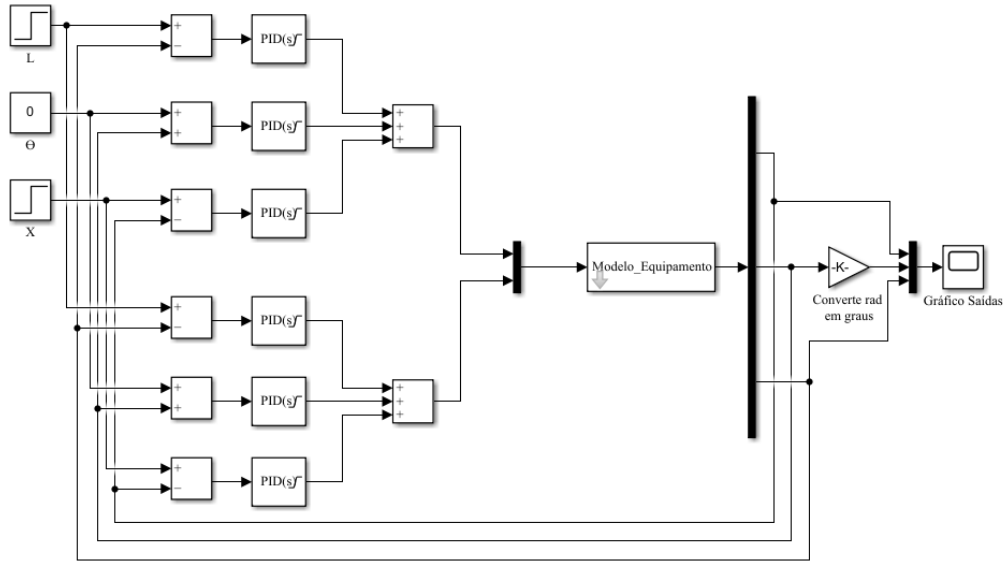


Figura 4.34 - Diagrama de blocos do sistema em malha fechada com o controlador PID no Simulink

Realimentação de estados com método de colocação de pólos

Foram também realizadas simulações do sistema, mas recorrendo ao controlador de realimentação de estados utilizando o método de colocação de pólos.

De acordo com o exposto no capítulo 3.4.2, de modo que seja possível dar ao sistema uma referência diferente de zero, foi necessário ajustar o mesmo (Figura 4.35), adicionando a diferença de $x_{ref} - x$, para que esta seja regulada para zero.

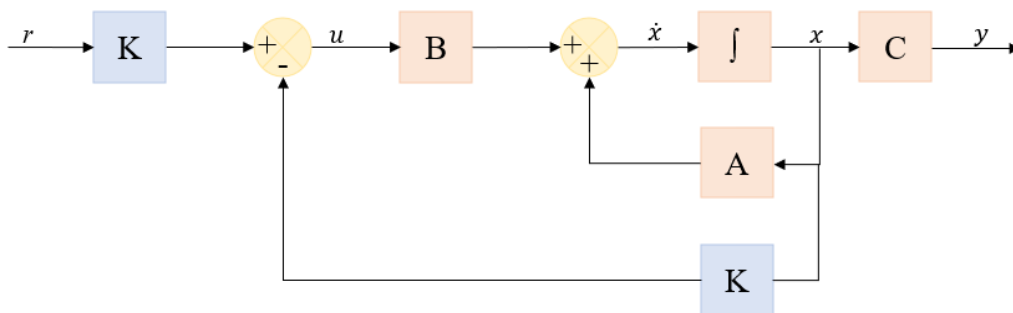


Figura 4.35 - Controle de realimentação de estados com colocação de pólos, dada uma referência ao sistema

As equações que representam o sistema em espaço de estado são:

$$\dot{x} = (A - BK)x + BKx_{ref} \quad (4.84)$$

$$y = Cx \quad (4.85)$$

E as entradas do sistema são dadas por:

$$u = x_{ref}K - Kx = K(x_{ref} - x) \quad (4.86)$$

Com base neste esquema foi concebido um programa no MATLAB para determinar a matriz de ganho K , a partir dos pólos desejados, utilizando o comando *place*. Este comando requer que a multiplicidade dos pólos desejados de malha fechada não seja superior à ordem da matriz B .

Por exemplo, se a matriz B for de $n = 1$, o *place* não funciona caso haja pólos múltiplos no vetor de pólos desejados.

De seguida, foi feito o controlo do sistema através de diagrama de blocos no *Simulink*, onde foram adicionadas as referências para os quatro estados do sistema, cuja configuração pode ser observado na Figura 4.36.

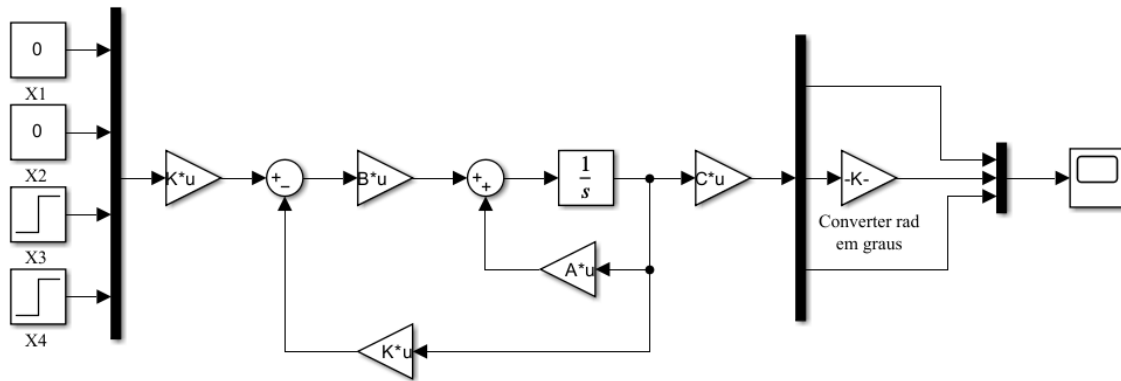


Figura 4.36 - Diagrama de blocos do sistema em malha fechada com realimentação de estados e colocação de pólos, considerando sistema regulador com dada referência no *Simulink*

Observador de ordem completa em sistema contínuo

Como já foi verificado em 4.6.4., foi necessário desacoplar o sistema para este ser observável. Desta forma, o observador de ordem completa adicionado ao sistema de controlo, considerou apenas as variáveis de estado x_1 , x_2 e x_3 (Figura 4.37). Uma vez que é possível medir o comprimento do cabo.

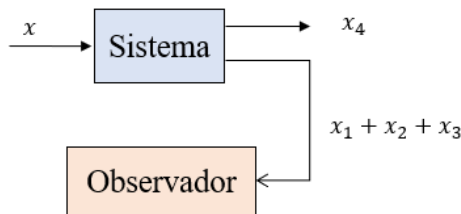


Figura 4.37 - Aplicação do observador a três estados do sistema

De relembrar que para aplicar o observador de ordem completa a matriz de observabilidade deve ser quadrada para ser invertível. Por esse motivo, foi necessário alterar a equação de saída. Como resultado, surgiu a matriz de saída $C_{1,3}$ somando as saídas originais, fica:

$$C_{1,3} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \quad (4.87)$$

Considerado o subcapítulo 4.6.4 sobre a observabilidade do sistema, temos:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -\frac{D_\theta(M+m)}{mMl_0^2} & 1 & 0 \\ \frac{D_\theta}{Ml_0^2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} -\frac{1}{l_0} \\ 0 \\ 1 \end{bmatrix} u_1 \quad (4.88)$$

$$[\dot{x}_4] = u_2 \quad (4.89)$$

Assim, o esquema de controlo para o controlador de realimentação de estados com observador de ordem completa foi projeto conforme representado na Figura 4.38.

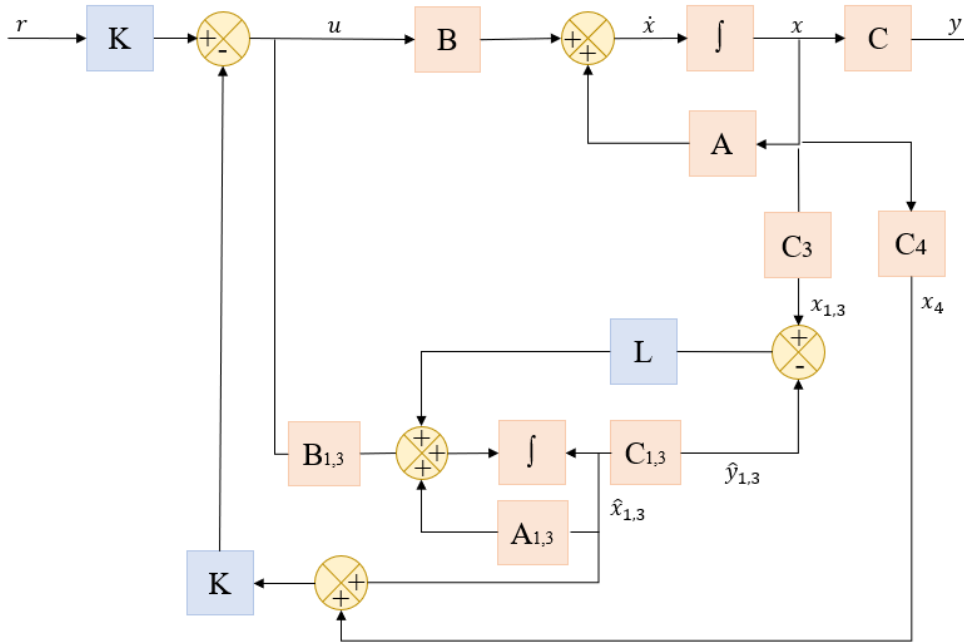


Figura 4.38 - Esquema do controlo de realimentação de estados com colocação de pólos e observador de ordem completa

Sendo que o vetor estimado será dado por:

$$\dot{\hat{x}}_{1,3} = A_{1,3}\hat{x}_{1,3} + B_{1,3}u + Ly_{1,3} - LC_{1,3}\hat{x}_{1,3} \quad (4.90)$$

ou seja,

$$\dot{\hat{x}}_{1,3} = (A_{1,3} - LC_{1,3})\hat{x}_{1,3} + B_{1,3}u + Ly_{1,3} \quad (4.91)$$

A partir do MATLAB, foi concebido um digrama de blocos para este controlador (Figura 4.39), em que a matriz de ganho L é determinada através do comando `place(A13', C13', q)`, no qual o vetor q é constituído pelos pólos desejados.

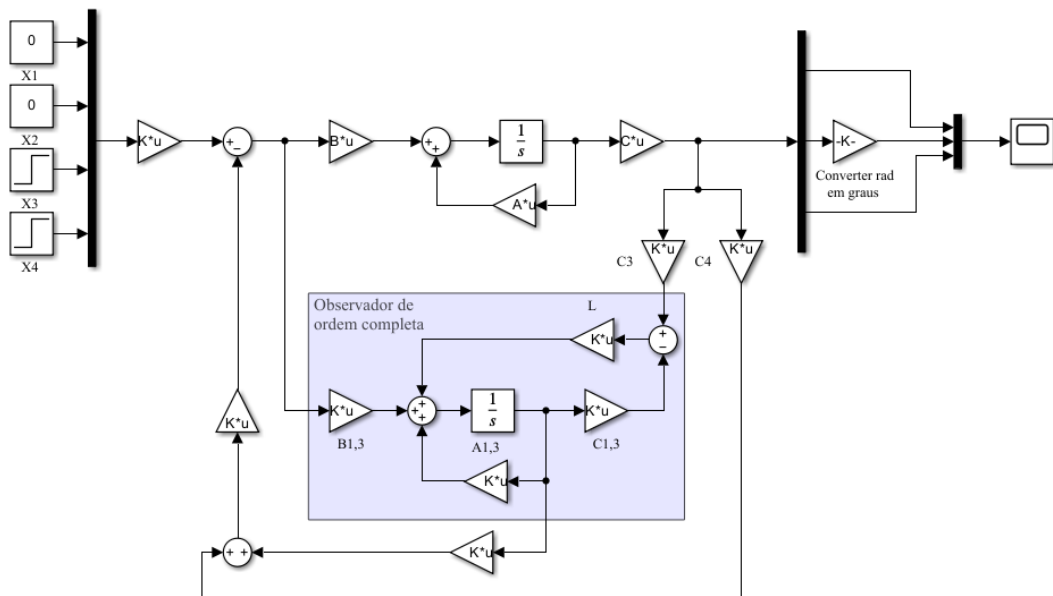


Figura 4.39 - Diagrama de blocos do sistema em malha fechada com realimentação de estados e colocação de pólos, com observador de ordem completa no Simulink

Para as simulações realizadas através do *Simulink* foram definidos os valores de referência para cada variável de estado. No caso da posição do carro foi definido 2,25 m, para o ângulo de oscilação foi de 0 rad e, inicialmente, para o comprimento do cabo 0,9 m. No gráfico da Figura 4.40 são demonstrados os resultados dos sistemas de controlo aplicado ao sistema, utilizando o controlador PID e o controlador de realimentação de estados com observador de ordem completa com a técnica de colocação de pólos.

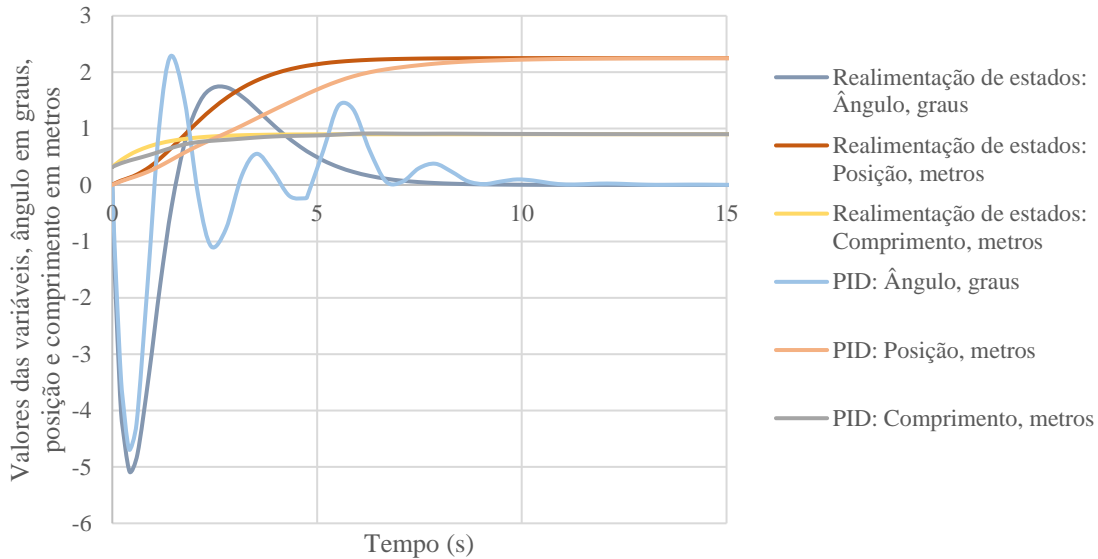


Figura 4.40 - Simulações obtidas entre diferentes controladores

Verificando pelo gráfico obtiveram-se melhores resultados utilizando o controlador de realimentação de estados, pois as três variáveis em estudo estabilizam mais rápido, comparando com o controlo PID. No entanto, de notar que os ganhos utilizados influenciam os resultados.

Observador de ordem reduzida em sistema contínuo

Visto que é possível medir o ângulo de oscilação de carga, a posição do carro e o comprimento do cabo, apenas será necessário observar, ou estimar, a segunda variável de estado, uma vez que esta não é possível medir.

Portanto, foi aplicado o controlador de observador de ordem reduzida, de acordo com o fundamento teórico já descrito. É possível escrever o sistema em função de uma partição do estado x , no qual \dot{x}_m representa a parte do estado que pode ser diretamente medida e \dot{x}_o a parte não mensurável diretamente. Assim, tendo em conta a expressão,

$$\begin{bmatrix} \dot{x}_m \\ \dots \\ \dot{x}_o \end{bmatrix} = \begin{bmatrix} A_{mm} & \vdots & A_{mo} \\ \dots & \vdots & \dots \\ A_{om} & \vdots & A_{oo} \end{bmatrix} \begin{bmatrix} x_m \\ \dots \\ x_o \end{bmatrix} + \begin{bmatrix} B_m \\ \dots \\ B_o \end{bmatrix} u \quad (4.92)$$

obteve-se

$$\dot{x} = \begin{bmatrix} -\frac{D_\theta(M+m)}{mMl_0^2} & 0 & 1 \\ 0 & 0 & 0 \\ \frac{D_\theta}{Ml_0^2} & 0 & 0 \end{bmatrix} x + \begin{bmatrix} -\frac{1}{l_0} \\ 1 \\ 0 \end{bmatrix} u \quad (4.93)$$

De seguida, e de acordo com os fundamentos teóricos, foi determinada a equação para o observador de estado de ordem reduzida a partir da equação,

$$\dot{\hat{N}} = \hat{A}\hat{N} + \hat{F}U + \hat{B}x_m \tag{4.94}$$

Uma vez que se trata de um sistema com três estados, dos quais são medido dois, então o número de estados a estimar é um, o que implica que L tenha de ter dimensão 1×2 .

$$L = [L_1 \quad L_2] \tag{4.95}$$

Na Figura 4.41 está apresentado o diagrama de blocos deste controlo, implementado no *Simulink*.

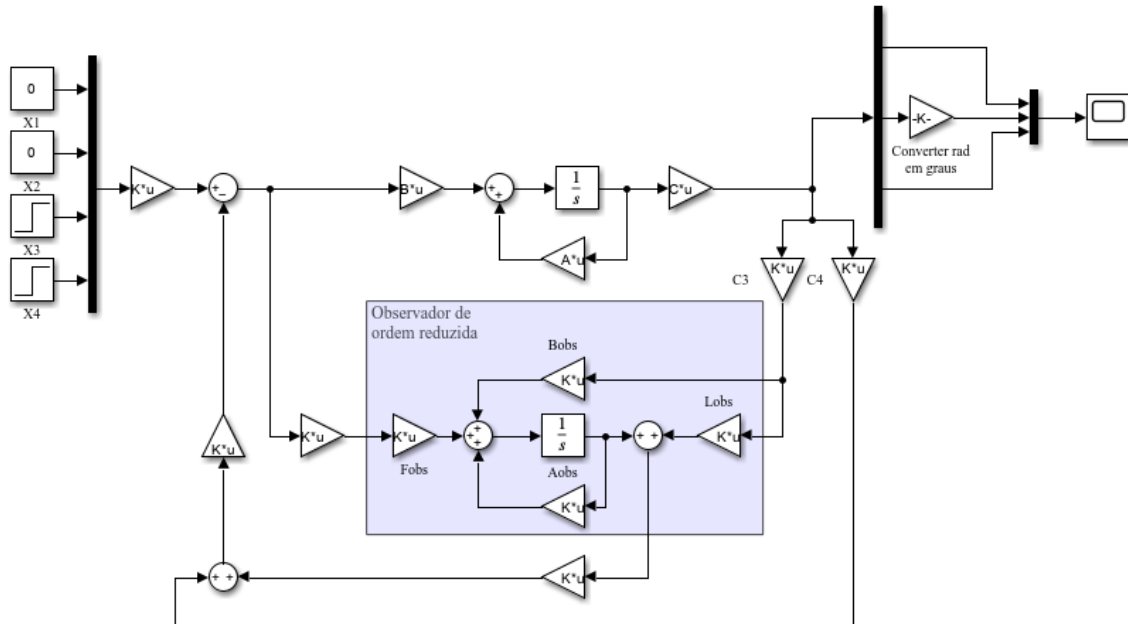


Figura 4.41 - Diagrama de blocos do sistema em malha fechada com realimentação de estados e colocação de pólos, com observador de ordem reduzida no Simulink

No gráfico da Figura 4.42 apresenta-se a comparação entre a variável estimada e a variável sem estimar, obtido a partir da simulação no *Simulink*.

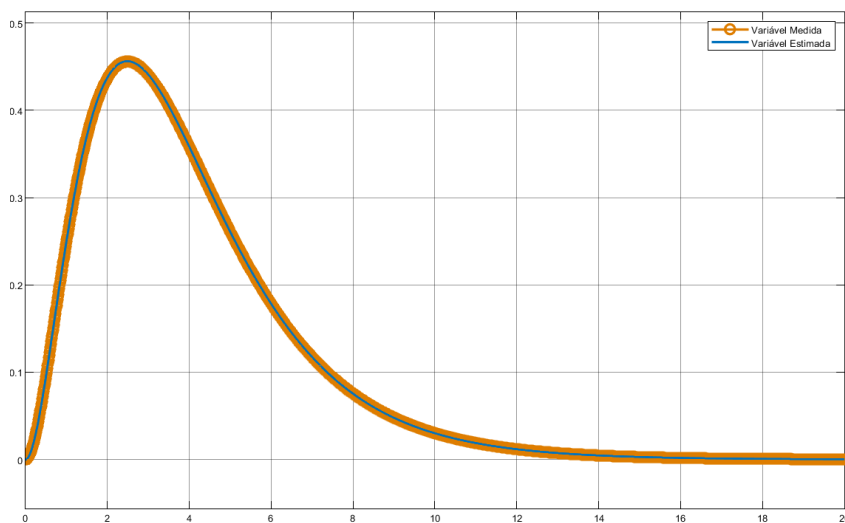


Figura 4.42 - Resultado entre variável estimada e não estimada

Verifica-se que o observador foi corretamente implementado e tem bom desempenho, porque os gráficos correspondem.

Discretização do sistema

Uma vez que a implementação do controlador é num microcontrolador, é necessário realizar a discretização do modelo, sendo este representado pelas seguintes equações,

$$x(k+1) = \Phi x(k) + \Gamma u(k) \quad (4.96)$$

$$y(k) = Cx(k) + Du(k) \quad (4.97)$$

Através do comando *c2d* do MATLAB determinou-se o sistema discreto do sistema em espaço de estado em estudo obtendo,

$$x(k+1) = \begin{bmatrix} 0,9994 & 0,009998 & 0 & 0 \\ -0,09799 & 0,9995 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} -0,009998 & 0 \\ 0,00049 & 0 \\ 0,01 & 0 \\ 0 & 0,01 \end{bmatrix} u(k) \quad (4.98)$$

$$y(k) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(k) \quad (4.99)$$

No entanto, também é possível aplicando a matriz exponencial a MT , em que T é o período de amostragem, como descrito no capítulo 3.5.

Observador de ordem completa em sistema discreto

À semelhança do sistema em contínuo também foi necessário desacoplar o sistema discreto, realizando o processo anterior.

$$x_{1,3}(k+1) = \begin{bmatrix} 0,9994 & 0,009998 & 0 \\ -0,09799 & 0,9995 & 0 \\ 0 & 0 & 1 \end{bmatrix} x_{1,3}(k) + \begin{bmatrix} -0,009998 & 0 \\ 0,00049 & 0 \\ 0,01 & 0 \end{bmatrix} u_{1,3}(k) \quad (4.100)$$

$$y_{1,3}(k) = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} x_{1,3}(k) \quad (4.101)$$

No programa desenvolvido no *Arduino* IDE foram aplicadas as matrizes anteriores, e ainda determinados os pólos conforme os pólos seleccionados na simulação, pela expressão de relação entre os pólos em s e os pólos em z :

$$p_z = e^{p_s T} \quad (4.102)$$

Utilizou-se o comando *place* no MATLAB de modo a descobrir a matriz de ganho K para o sistema discreto, que posteriormente foi implementada no programa do *Arduino*.

Observador de ordem reduzida em sistema discreto

Para o sistema discreto com observador de ordem reduzida foi considerada a expressão

$$\begin{bmatrix} \dot{x}_m \\ \dots \\ \dot{x}_o \end{bmatrix} = \begin{bmatrix} A_{mm} & \vdots & A_{mo} \\ \dots & \vdots & \dots \\ A_{om} & \vdots & A_{oo} \end{bmatrix} \begin{bmatrix} x_m \\ \dots \\ x_o \end{bmatrix} + \begin{bmatrix} B_m \\ \dots \\ B_o \end{bmatrix} u \quad (4.103)$$

determinando-se

$$\begin{aligned} A_{mm} &= \begin{bmatrix} 0,9994 & 0 \\ 0 & 1 \end{bmatrix} & A_{oo} &= [0,9995] \\ A_{mo} &= \begin{bmatrix} 0,00998 \\ 0 \end{bmatrix} & B_m &= \begin{bmatrix} -0,00998 \\ 0,01 \end{bmatrix} \\ A_{om} &= [-0,09799 \quad 0] & B_o &= [0,00049] \end{aligned}$$

Para comparar a variável de estado do sistema medida e estimada foi desenvolvido um programa no MATLAB (Apêndice B). Sendo G a matriz de ganho composta por g_1 e g_2 , variaram-se esses valores de ganho para entender a sua influência em xh , que corresponde à variável que se pretende estimar. Os resultados obtidos estão apresentados na Figura 4.43.

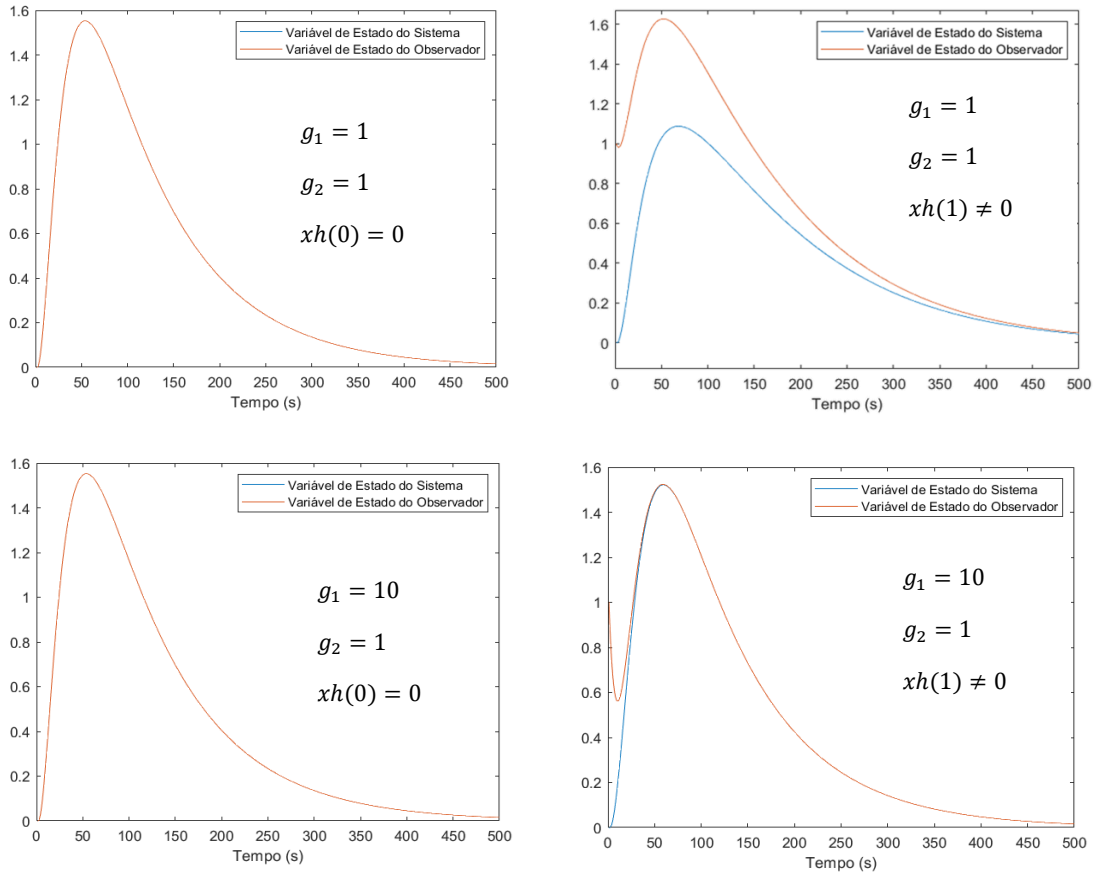


Figura 4.43 - Comparação e influência de valores do observador

Verifica-se que o observador faz uma boa aproximação do valor da variável, e quanto maior o valor do ganho melhor a estimação do observador.

4.6.7. Simulações com Trajetória

Com o objetivo de simular uma rota específica para o equipamento, foi desenvolvida uma trajetória. Na Figura 4.44 está representada a trajetória que se pretende que o sistema siga, no qual o eixo das ordenadas, yy , corresponde ao movimento do comprimento do cabo, e o eixo das abscissas, xx corresponde ao movimento da posição do carro.

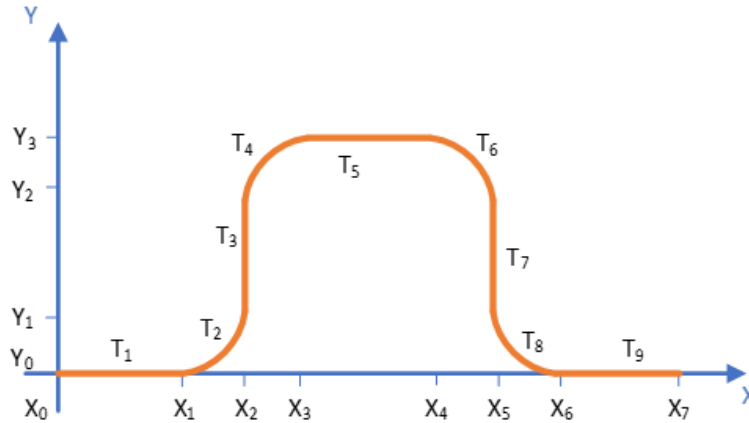


Figura 4.44 - Trajetória da carga desejada dividida em troços

Para que o sistema realize esta trajetória tem de ser concebida uma referência que corresponda com o movimento que se pretende. Desta forma, foi determinada uma equação paramétrica de uma trajetória por ramos, constituída por troços verticais, horizontais e circulares.

Considera-se uma velocidade constante V , o que origina uma velocidade angular constante w nos troços circulares. Considera-se também que cada troço j tem um instante e uma posição iniciais t_j, x_j, y_j e um instante e posição finais que coincidem com as iniciais do troço seguinte, $t_{j+1}, x_{j+1}, y_{j+1}$. Os troços circulares necessitam de mais dois termos internos que indiquem o ponto da circunferência que corresponde ao início, que toma a forma de um ângulo inicial θ_i e o raio da circunferência r . Assim, genericamente, teremos as equações para cada um dos três tipos de troços, apresentadas nas Tabela 4.9, Tabela 4.10 e Tabela 4.11.

Tabela 4.9 - Equações do troço horizontal i

t	x	y	t_{i+1}	x_{i+1}	y_{i+1}
$t_i < t < t_{i+1}$	$x = x_i + Vt$	$y = y_i$	$t_i + \frac{x_{i+1} - x_i}{V}$	$x_i + V(t_{i+1} - t_i)$	y_i

Tabela 4.10 - Equações do troço vertical i

t	x	y	t_{i+1}	x_{i+1}	y_{i+1}
$t_i < t < t_{i+1}$	$x = x_i$	$y = y_i + Vt$	$t_i + \frac{y_{i+1} - y_i}{V}$	x_i	$y_i + V(t_{i+1} - t_i)$

Tabela 4.11 - Equações do troço circular i

t	x	y
$t_i < t < t_{i+1}$	$x = x_i + r \cos(\theta_i + w(t - t_i))$	$y = y_i + r \sin(\theta_i + w(t - t_i))$
t_{i+1}	x_{i+1}	y_{i+1}
$t_i + \frac{\text{atan2}(y_{i+1} - y_i, x_{i+1} - x_i) - \theta_i}{w}$	$x_{i+1} = x_i + r \cos(\theta_i + w(t_{i+1} - t_i))$	$y_{i+1} = y_i + r \sin(\theta_i + w(t_{i+1} - t_i))$

Desta forma, tomando como exemplo a trajetória da Figura 4.44, e considerando que os troços circulares cobrem todos arcos de 90° , foi gerado um programa no MATLAB (Apêndice B).

Visto que as saídas da trajetória são a posição do carro e o comprimento do cabo, estas correspondem com as referências destas duas variáveis. No *Simulink*, exposto o diagrama de blocos no Apêndice C, foi adicionada a trajetória como referência para os dois estados e o resultado pode ser visto na Figura 4.45.

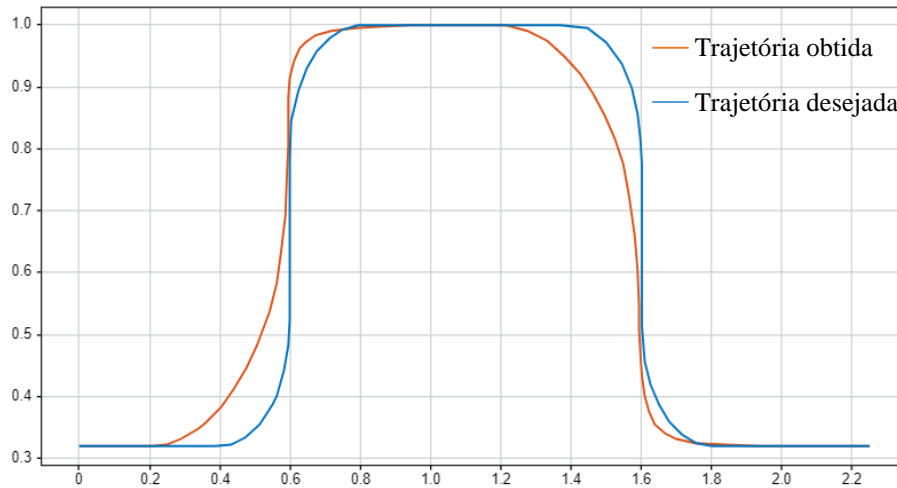


Figura 4.45 - Comparação entre trajetória desejada versus trajetória simulação

5. Testes Experimentais e Resultados

5.1. Condições de Realização dos Testes

Após simulações realizadas iterativamente utilizando MATLAB e *Simulink* para analisar a resposta do sistema sob diferentes valores das matrizes de ganho, procedeu-se aos testes experimentais. Para conduzir esses testes, foi seguido o processo descrito na Figura 4.28. Os parâmetros definidos para realizar tanto as simulações quanto os ensaios estão presentes na Tabela 5.1.

Tabela 5.1 - Parâmetros considerados na realização de testes experimentais

Parâmetro	Valor
Massa da carga suspensa (<i>kg</i>)	0,5
Massa do carro máxima (<i>kg</i>)	3
Comprimento total do carril (<i>m</i>)	18
Posição inicial do carro (<i>m</i>)	0
Posição final do carro (<i>m</i>)	2,25
Comprimento máximo do cabo (<i>m</i>)	2
Comprimento inicial do cabo (<i>m</i>)	0,44
Comprimento final do cabo (<i>m</i>)	1,0
Tipo de entrada na referência da posição do carro	Degrau
Tipo de entrada na referência do comprimento do cabo	Degrau
Ângulo máximo de oscilação de carga (°)	25
Referência do ângulo (°)	0
Velocidade linear máxima do carro (<i>m/s</i>)	3,34
Velocidade linear máxima de subir da carga (<i>m/s</i>)	0,16
Velocidade linear máxima de descer da carga (<i>m/s</i>)	0,24

Como mencionado anteriormente, um dos sistemas de controlo utilizado considerou um observador de ordem reduzida, uma vez que apenas uma variável de estado não é possível ser medida diretamente. Foi realizada uma comparação da resposta do sistema utilizando este observador e utilizando o observador de ordem completa. Pela Figura 5.1 verifica-se que os gráficos se sobrepõem, com uma ligeira variação na amplitude do ângulo de oscilação no observador de ordem reduzida.

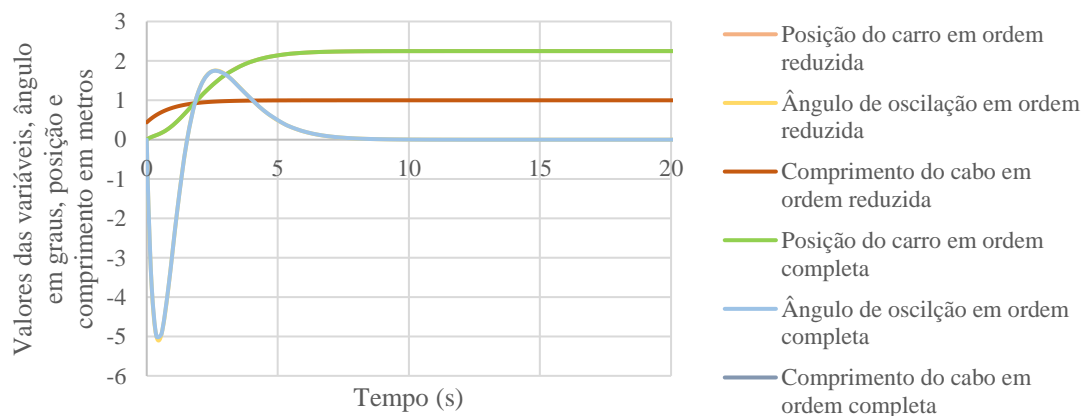


Figura 5.1 - Simulação e comparação do controlo com observador de ordem completa e com observador de ordem reduzida

No entanto, na prática, este controlador mostrou maior sensibilidade ao ruído e um tempo ligeiramente superior de estabilização, em relação ao controlador com o observador de ordem completa. Nas Figura 5.2, Figura 5.3 e Figura 5.4 são apresentados os resultados experimentais e comparações entre ambos os sistemas de controle, para o ângulo de oscilação de carga, posição do carro e comprimento do cabo, respetivamente.

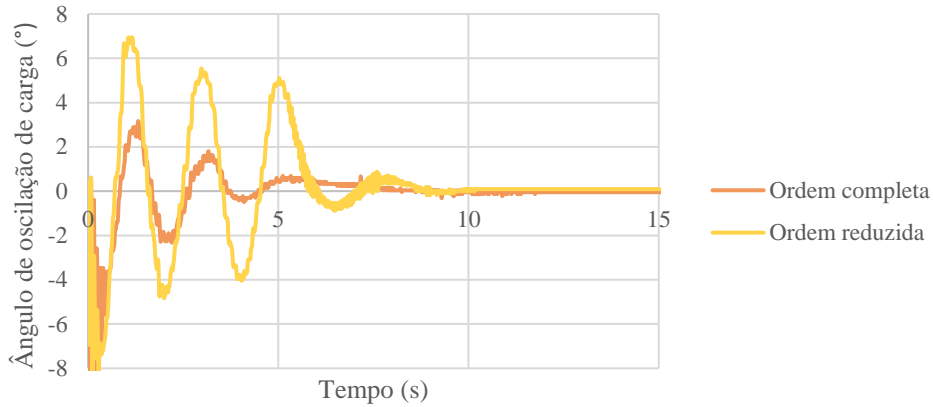


Figura 5.2 - Resultados experimentais do ângulo de oscilação de carga para controlador com observador de ordem reduzida versus observador de ordem completa

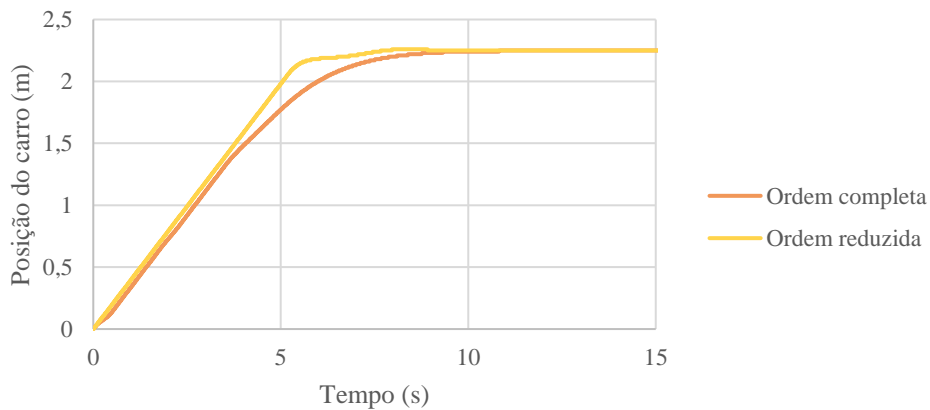


Figura 5.3 - Resultados experimentais da posição do carro para controlador com observador de ordem reduzida versus observador de ordem completa

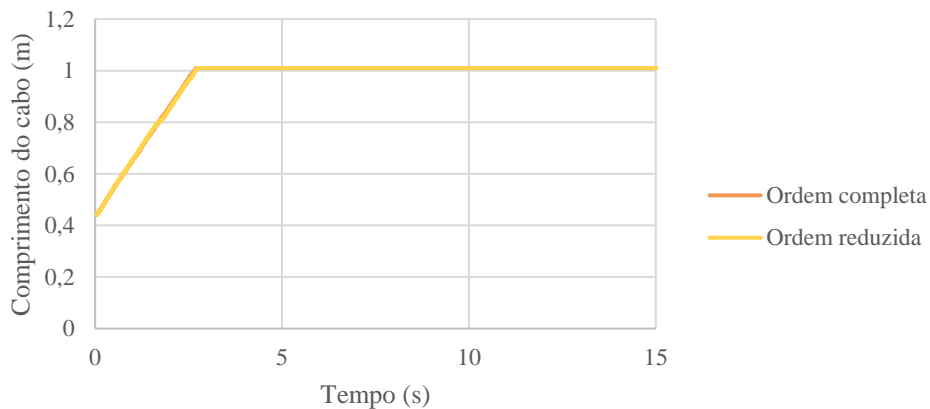


Figura 5.4 - Resultados experimentais do comprimento do cabo para controlador com observador de ordem reduzida versus observador de ordem completa

Ao analisar os gráficos, constata-se que o ângulo de oscilação (Figura 5.2) apresenta uma maior amplitude no controlador com observador de ordem reduzida, e leva mais tempo a estabilizar. Enquanto no controlo com observador de ordem completa, o ângulo tem amplitude inicial entre $[-9^{\circ};3^{\circ}]$ aproximadamente, e nos 5 segundos o ângulo encontra-se entre $[-0,5^{\circ};0,5^{\circ}]$, no controlo com observador de ordem reduzida a amplitude inicial está entre $[-8,5;7^{\circ}]$, e nos 5 segundos está entre $[-4^{\circ};5^{\circ}]$. Na posição do carro (Figura 5.3), quando esta se aproxima da referência final, verifica-se que no observador de ordem reduzida, o carro ultrapassa a posição desejada, voltando depois atrás para estabilizar. Relativamente ao comprimento do cabo (Figura 5.4), em ambos se observa um comportamento semelhante, isto deve-se ao facto de que esta variável de saída é considerada desacoplada em ambos os controlos, sendo que não é considerada para nenhum dos observadores. Uma das razões para o controlador com observador de ordem reduzida ser tão diferente dos resultados em simulação, e de não ser tão eficaz, é a sua maior sensibilidade na medição das variáveis de saída, especialmente no ângulo de oscilação, em comparação com o observador de ordem completa [38]. Na prática também foi verificado uma grande ressonância no equipamento quando utilizado o controlador com observador de ordem reduzida. Desta forma, foi selecionado o sistema de controlo com realimentação de estados com observador de ordem completa, aplicando o método de colocação de pólos, para controlo na movimentação da carga suspensa, pois este apresentou melhores resultados relativamente ao controlador com observador de ordem reduzida.

De modo a analisar o desempenho do modelo e do sistema de controlo, de seguida, serão apresentados os resultados que dizem respeito a um movimento controlado com uma referência final para a posição do carro e para o comprimento do cabo, um movimento controlado com trajetória pré-definida para a carga suspensa, e testes realizados considerado um modelo linear do sistema por troços.

5.2. Movimento Controlado

No presente subcapítulo, são demonstrados os resultados para um movimento controlado utilizando o controlo com observador completo, para as variáveis de estado x_1 , x_2 e x_3 . Pois a quarta variável, x_4 , correspondente ao comprimento do cabo, é um sistema desacoplado do sistema constituído pelas outras três variáveis de estado. Este movimento consiste em dar apenas uma referência final ao comprimento do cabo e à posição do carro que o sistema deverá atingir.

Foram realizados vários ensaios alterando a matriz de ganho de realimentação de estados para analisar a influência que a posição dos pólos tem na resposta e no desempenho do sistema. Na Tabela 5.2 foram expostos os valores dos ganhos que foram usados para algumas das simulações realizadas.

Tabela 5.2 - Valores dos ganhos para cada variável, para diferentes simulações

Simulação (ou n° do teste)	K_1	K_2	K_3	K_4
1	-3,5709	-0,4932	0,2185	1,0940
2	-3,5709	-0,4932	0,2185	9,5163
3	-1,9908	-0,8527	0,0339	9,5163
4	-2,4640	-0,7895	0,0524	9,5163
5	-2,5612	-0,7836	0,0529	0,8960
6	-3,0132	-0,6816	0,0914	1,0940
7	-2,1783	-0,8245	0,0436	1,0940
8	-2,0869	-0,8416	0,0363	1,0940
9	-2,2884	-0,8438	0,0303	1,0940

Nas Figura 5.5, Figura 5.6 e Figura 5.7 são apresentados os resultados das simulações do ângulo de oscilação da carga, da posição do carro e do comprimento do carro, respectivamente, considerando os valores de ganho apresentadas anteriormente.

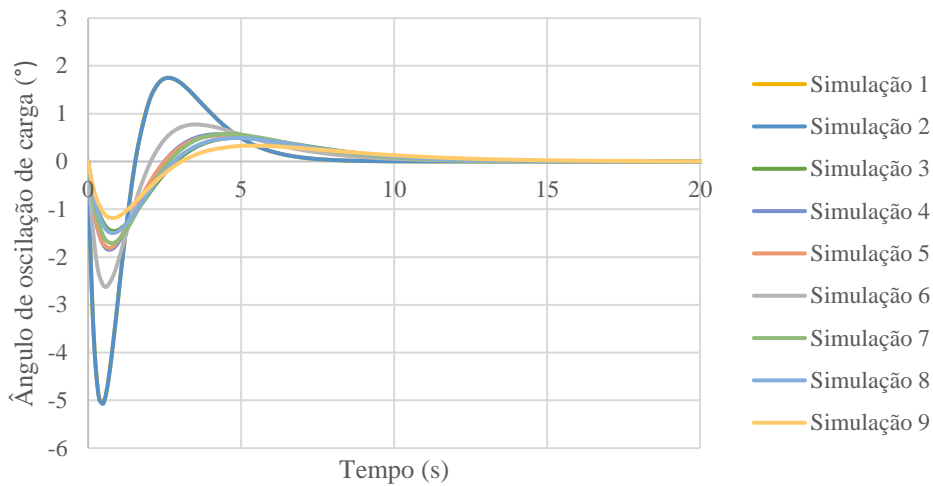


Figura 5.5 - Comparação de simulações do ângulo de oscilação para diferentes pólos

Pela Figura 5.5 verifica-se que nas simulações 1 e 2 há maior amplitude inicial do ângulo de oscilação da carga, mas menor tempo de acomodação, relativamente às outras. Nestas duas, o valor de ganho K_1 é o mais negativo, e o valor de K_2 está mais perto de zero, comparativamente com as outras matrizes de ganho.

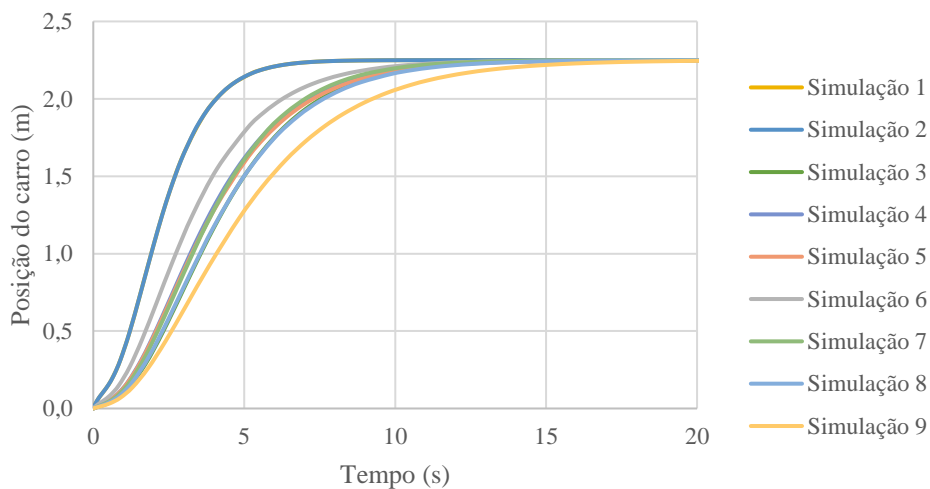


Figura 5.6 - Comparação de simulações da posição do carro para diferentes pólos

Analisando a Figura 5.6 observa-se que a posição do carro nos resultados 1 e 2 estabiliza mais rapidamente.

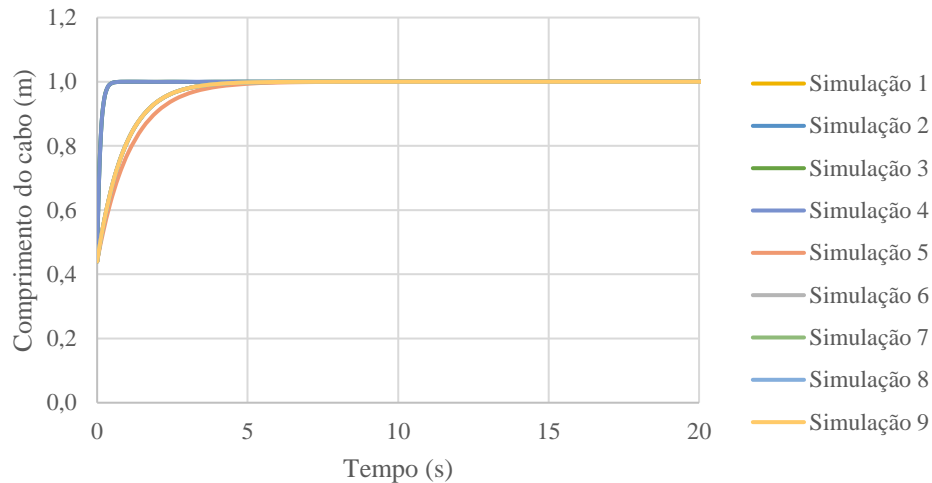


Figura 5.7 - Comparação de simulações do comprimento do cabo para diferentes pólos

Através da Figura 5.7 é possível afirmar que o comprimento do cabo nas simulações 2, 3 e 4 acomoda em menos tempo em relação às outras. Para estas três simulações o valor de K_4 é maior. Sendo que a simulação 5 possui o menor valor de K_4 , este tem um maior tempo de estabilização.

Na Tabela 5.3 foram registadas, para cada uma das simulações, o tempo de acomodação do ângulo de oscilação de carga, da posição do carro e do comprimento do cabo.

Tabela 5.3 - Resultados dos valores das variáveis obtidos através de simulações

Simulação (ou n° do teste)	Acomodação da oscilação do ângulo (segundos)	Acomodação da posição do carro (segundos)	Acomodação do comprimento do cabo (segundos)
1	9,95	8,26	4,47
2	10,03	8,28	0,50
3	16,31	15,41	0,54
4	15,94	15,37	0,48
5	16,76	15,76	5,69
6	14,91	14,91	5,11
7	15,33	14,33	4,69
8	17,27	16,27	4,97
9	19,72	20,44	4,83

Quando os pólos desejados estão mais afastados da origem e localizados no semiplano esquerdo do plano complexo, a amplitude do sinal de controlo tende a ser maior e o tempo de acomodação para qualquer variável de saída tende a ser menor. A partir das simulações, foi observado que no vetor de pólos desejados, quando maior o número de pólos escolhidos com parte real negativa e mais afastados do eixo imaginário, o valor de ganho K_1 tem tendência a ser menor (mais negativo), enquanto os valores de K_3 e K_4 tendem a ser maiores. Isto ocorre porque os pólos mais distantes do eixo imaginário e com valores negativos da parte real correspondem a uma resposta mais rápida do sistema, o que exige maiores valores de ganho K para atender aos requisitos de desempenho desejados do sistema.

Realizando uma análise aos resultados, verificou-se que quanto menor o ganho K_1 (ganho do ângulo) menor é o tempo de acomodação do ângulo de oscilação e maior a sua amplitude inicial. Quando maior o valor de ganho K_3 (ganho da posição) menor o tempo de acomodação da posição do carro. No caso do K_4 (ganho do comprimento) quanto maior o seu valor menor o tempo de acomodação do valor de comprimento desejado. No entanto, isto não é assim tão linear. Por exemplo, nas simulações 1 e 2, os valores dos ganhos K_1 , K_2 e K_3 são iguais, mas no caso do K_4 o valor na simulação 2 é maior. Consequentemente, o tempo de acomodação do comprimento do cabo é menor na simulação 2, e em contrapartida, o tempo de estabilização do ângulo e da posição na simulação 1 são menores. Em relação às simulações 3 e 5, seria de esperar que o tempo de acomodação do ângulo de oscilação e da posição do carro fossem menores para a simulação 5, pois tem menor valor do ganho K_1 e maior valor do ganho K_3 , contudo não é o que se observa nos resultados. Mas sabe-se que há um pólo desejado na simulação 3 localizado no semiplano esquerdo do plano complexo e mais afastado da origem, comparando com os pólos desejados na simulação 5.

Conforme as prioridades de controlo das três variáveis de saída, oscilação do ângulo de carga, posição do carro e comprimento do cabo, podem ser seleccionados os valores de ganho mais adequados, estando estes apresentados na Tabela 5.4.

Tabela 5.4 - Teste de simulação escolhido de acordo com a prioridade de controlo

Prioridade	Teste de Simulação
Ângulo de oscilação do cabo	Simulação 1
Posição do carro	Simulação 1
Comprimento do cabo	Simulação 4

No entanto, comparado a simulação 1 com o teste experimental 2 (Figura 5.8), verifica-se uma grande amplitude do ângulo de oscilação e diferenças entre ambos. Facto que se justifica porque a escolha dos pólos é um compromisso entre a velocidade de resposta do erro e sensibilidade aos distúrbios e ruídos, ou seja, para valores de pólos mais afastados do eixo imaginário maior a velocidade de resposta do erro, e maior os efeitos provocados pelos distúrbios e ruídos.

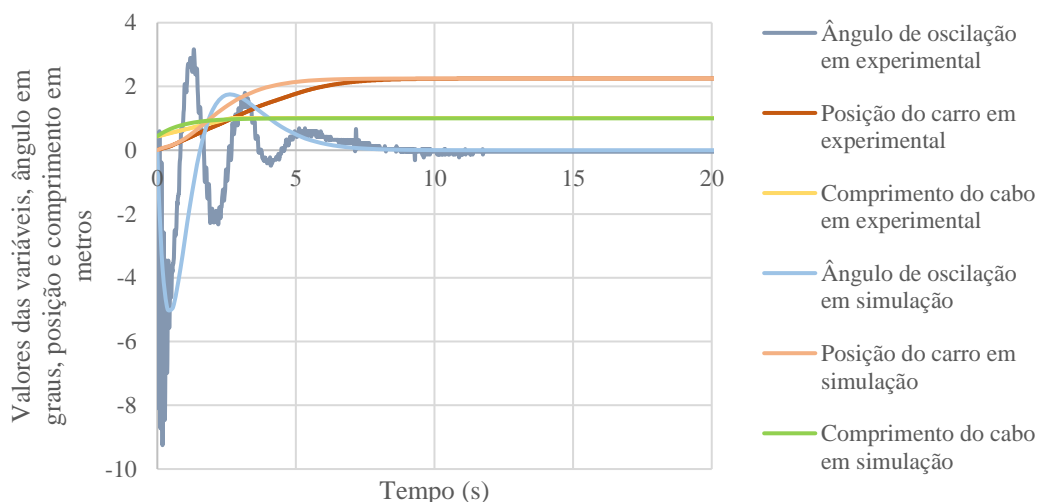


Figura 5.8 - Comparação das variáveis de saída entre simulação e teste experimental para os valores de ganho da simulação 1

Desta forma, optou-se por seleccionar a simulação com os melhores resultados seguintes, para as prioridades de controlo do ângulo de oscilação da carga e para a posição do carro, sendo esta a simulação 6.

De seguida, são apresentadas comparações entre o experimental e simulado dos testes 6 e 4. Em primeiro lugar são apresentados os gráficos de oscilação do ângulo de carga, seguido da posição do carro, e por fim o comprimento do cabo.

Comparação 1: Teste Experimental 6

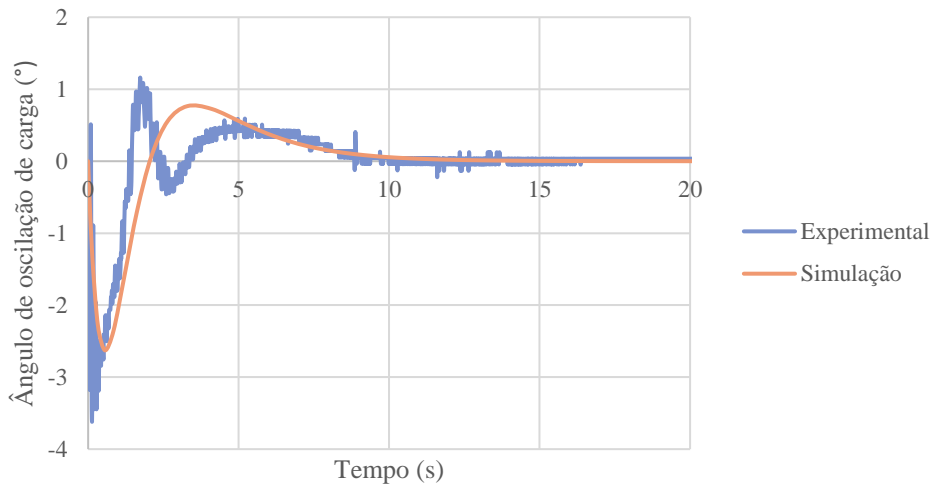


Figura 5.9 - Comparação dos valores do teste 6 do ângulo de oscilação, entre resultado simulado e experimental

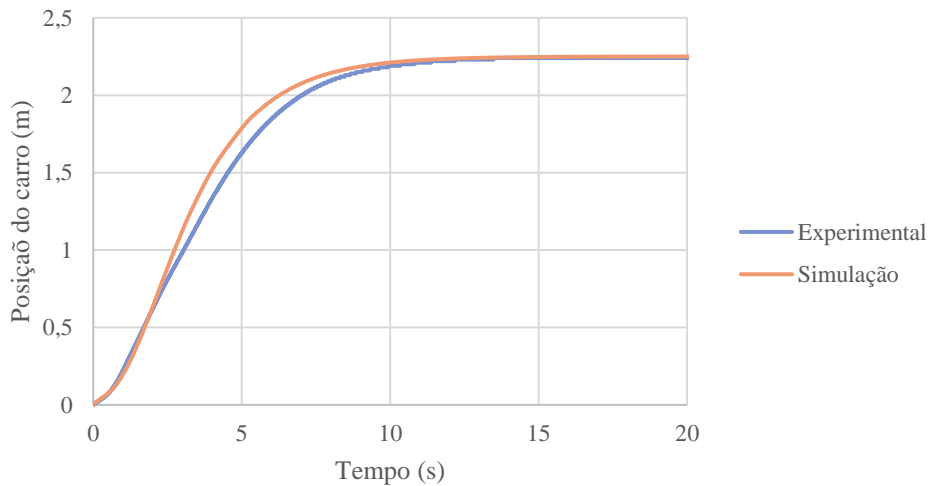


Figura 5.10 - Comparação dos valores do teste 6 da posição do carro, entre resultado simulado e experimental

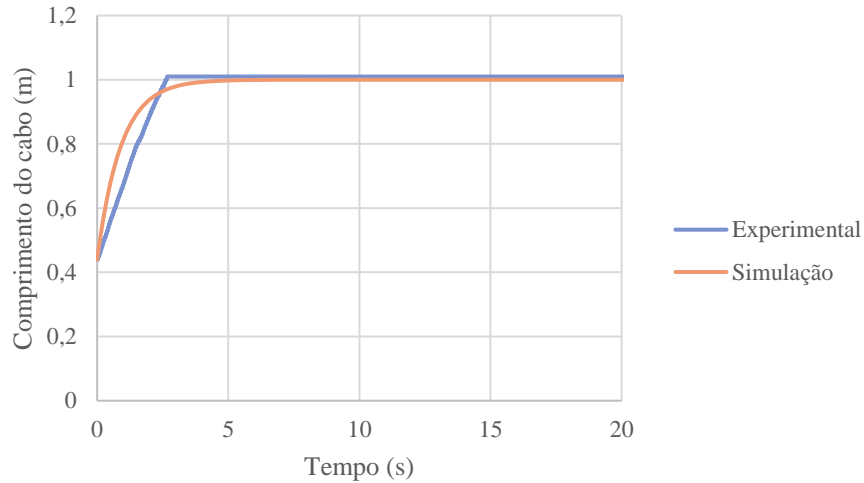


Figura 5.11 - Comparação dos valores do teste 6 do comprimento do cabo, entre resultado simulado e experimental

Relativamente ao teste 6 verifica-se uma discrepância no comportamento do ângulo de oscilação entre a simulação e a realidade, observando algumas diferenças de amplitude e de fase entre as duas curvas (Figura 5.9). No entanto, o tempo de acomodação do ângulo de oscilação, cerca de 15 segundos, é idêntico em ambos, se ignorarmos os ruídos no gráfico do teste experimental, cujo valor de erro é próximo de $\pm 0,01^\circ$. Ainda, na prática a oscilação máxima do ângulo registada foi de cerca de $[-4,4^\circ; 1,2^\circ]$, diferente da simulação que está entre $[-2,6^\circ; 0,76^\circ]$. Experimentalmente, salienta-se que após o pico de oscilação, em torno dos 2,7 segundos, aquando estabilização do valor do comprimento do cabo e enquanto ainda é realizado o transporte da carga para a posição desejada, o ângulo está limitado a uma pequena faixa entre $[-0,5^\circ; 0,5^\circ]$. Convergingo para 0° após a estabilização da posição do carro, nos 13,5 segundos. Este valor difere da simulação, que foi atingido nos 14,9 segundos, sendo mais rápido a atingir valores próximos da posição desejada (Figura 5.10). No caso do comprimento do cabo, na simulação a mudança de valores de comprimento é realizada de forma mais suave, em contraste do que acontece na realidade, onde o valor de referência é alcançado mais rapidamente, em relação à simulação cujo valor foi de 5,1 segundos (Figura 5.11).

Comparação 2: Teste Experimental 4

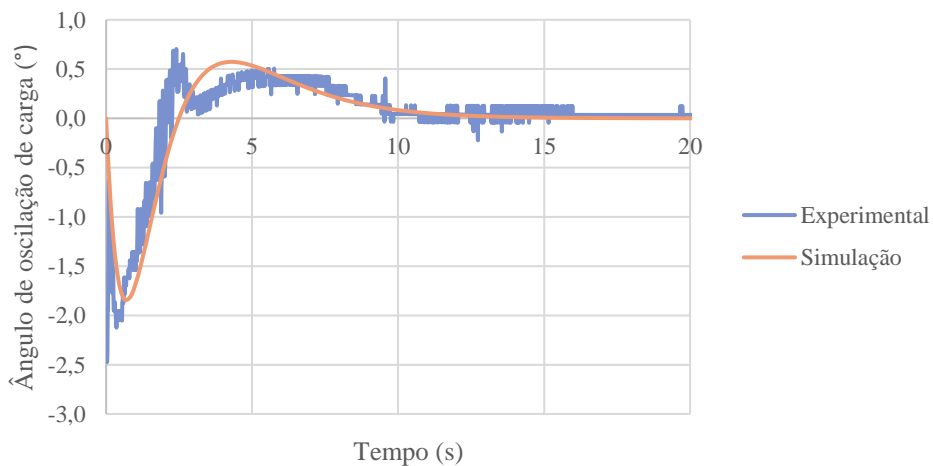


Figura 5.12 - Comparação dos valores do teste 4 do ângulo de oscilação, entre resultado simulado e experimental

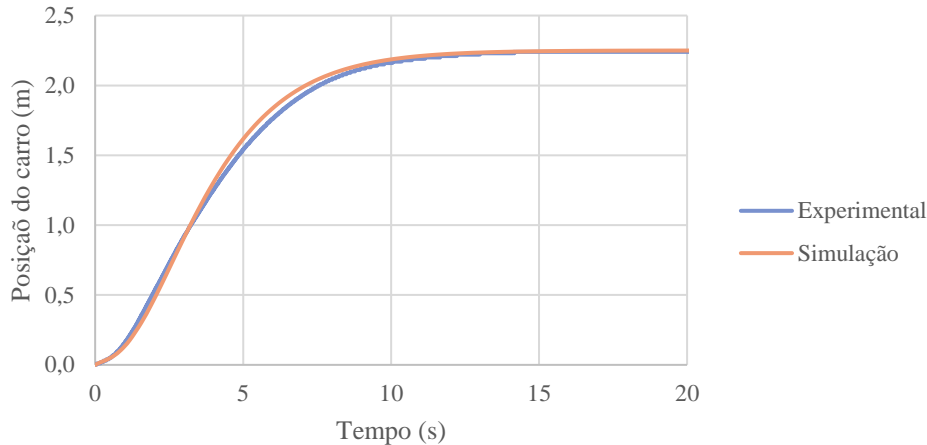


Figura 5.13 - Comparação dos valores do teste 4 da posição do carro, entre resultado simulado e experimental

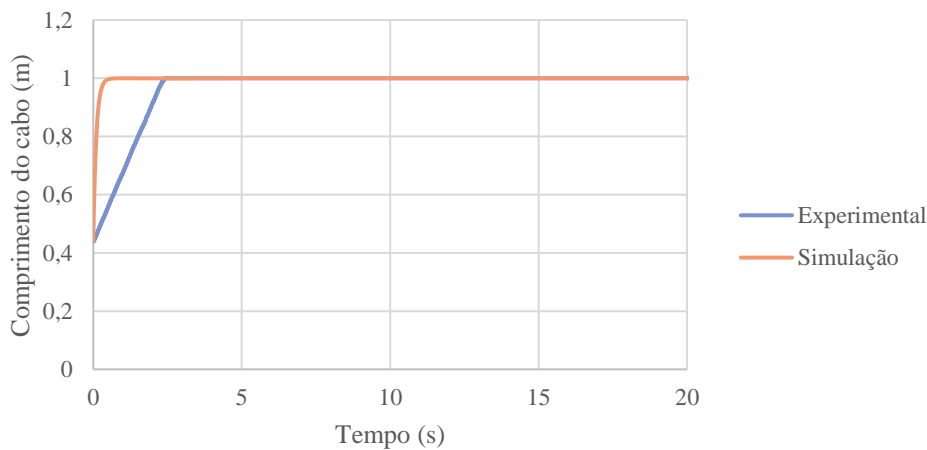


Figura 5.14 - Comparação dos valores do teste 4 do comprimento do cabo, entre resultado simulado e experimental

Comparando as curvas de simulação e de experimental para os valores do teste 4, observa-se uma semelhança entre os resultados do ângulo de oscilação e da posição do carro (Figura 5.12). O ângulo máximo está entre $[-2,5^\circ; 0,7^\circ]$ no experimental, enquanto na simulação é de $[-1,8^\circ; 0,6^\circ]$. No teste experimental, após o tempo de estabelecimento do comprimento do cabo, em 2,4 segundos, o ângulo de oscilação fica limitado a $[-0,2^\circ; 0,5^\circ]$, nos 10 segundos verifica-se um erro residual de $\pm 0,1^\circ$, atingido os 0° nos 16 segundos. Este valor é bastante idêntico ao obtido na simulação, cujo valor foi de 15,94 segundos. Já a posição do carro também tem um tempo de acomodação semelhante entre simulação e teste de laboratório, cujos valores são 15,37 segundos e 14,2 segundos, respetivamente (Figura 5.13). No caso do comprimento do cabo, é notada uma diferença entre ambas as curvas, pois experimentalmente o comprimento do cabo demora mais tempo de acomodação, enquanto na simulação estava previsto um tempo de 0,48 segundos (Figura 5.14). Isto pode dever-se ao facto de que o motor do cabo não consegue responder tão rapidamente ao controlo.

De notar, que devido à presença de ruído na leitura do ângulo de oscilação, a análise através dos gráficos torna-se mais subjetiva, relativamente à análise do comportamento da posição do carro e do comprimento do cabo.

Para a reforçar o facto de que para valores de pólos mais afastados do eixo imaginário do plano complexo, mais se verifica os efeitos de ruídos e perturbações, é apresentada na Figura 5.15 o gráfico experimental para o teste com os valores mais próximos da origem, que corresponde à simulação 9.

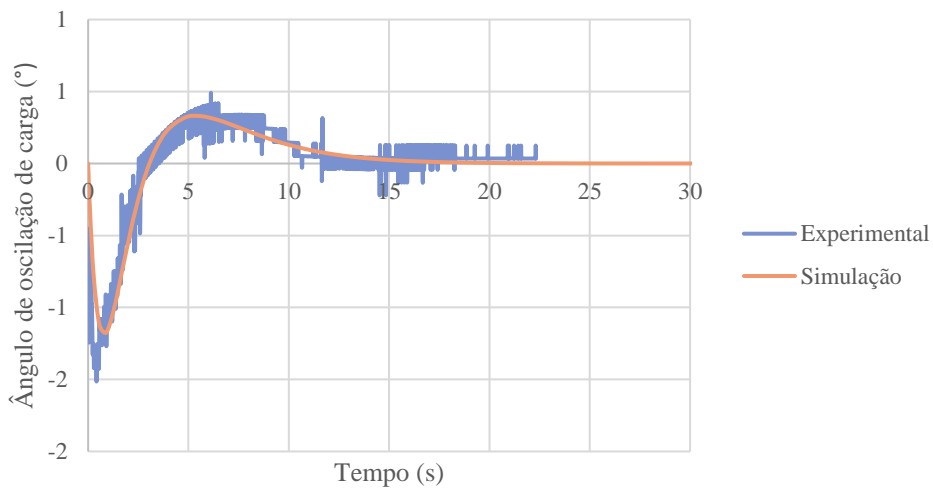


Figura 5.15 - Comparação dos valores do teste 9 do ângulo de oscilação, entre resultado simulado e experimental

Verifica-se que para os valores de ganho da simulação 9, a resposta do ângulo de saída é mais semelhante entre simulação e experimental, embora os erros de leitura observados.

De seguida, seria interessante realizar uma comparação deste movimento, com e sem controlo. Na Figura 5.16 apresenta-se a comparação entre simulação e experimental do sistema sem controlo, cuja carga foi solta com 8° de ângulo inicial.

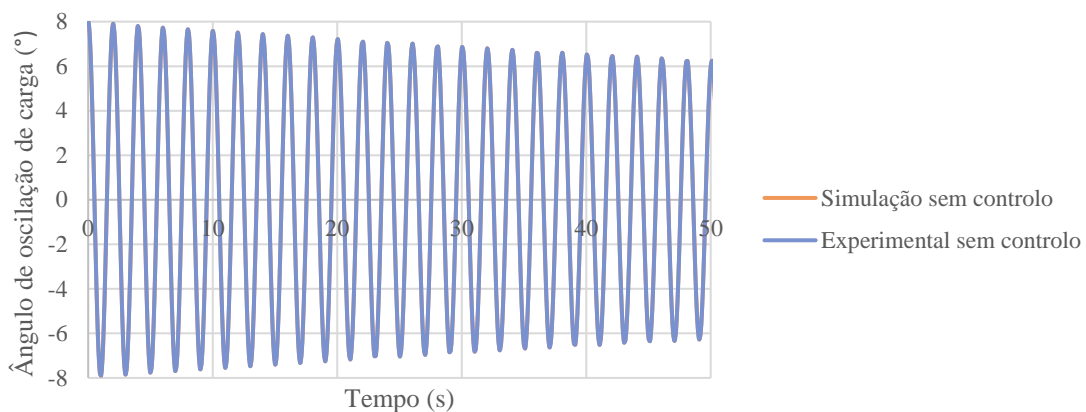


Figura 5.16 - Comparação entre simulação e experimental sem controlo do sistema

Verifica-se que os gráficos de ambos estão sobrepostos, o que significa que a modelação do sistema foi realizada de forma correta.

Adicionalmente, foi realizado um teste experimental do movimento sem controlo, e com velocidades máximas para o motor do carro e motor do cabo (Figura 5.17), de modo a comparar com o movimento com controlo.

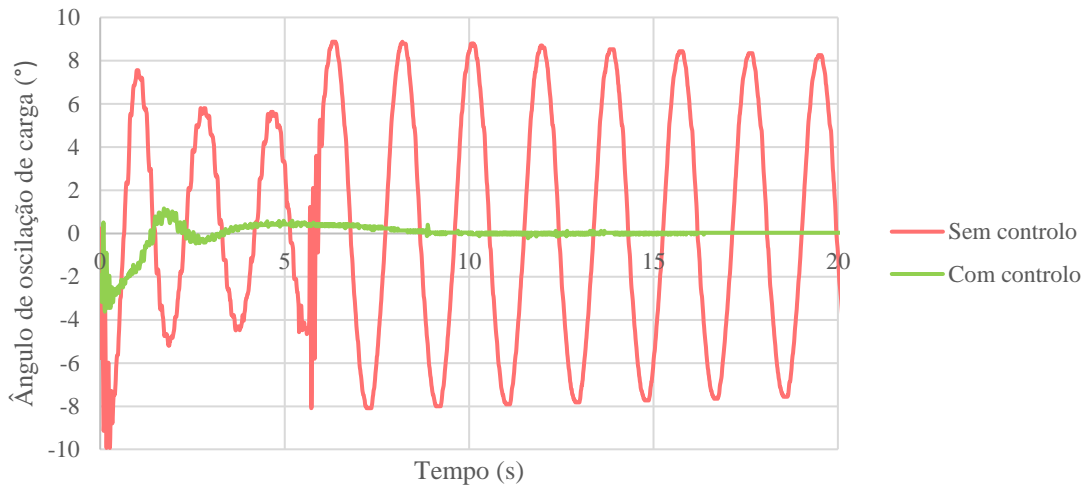


Figura 5.17 - Resultado do ângulo de oscilação de carga, movimento com controle versus sem controle

No movimento sem controle, quando o carro atinge a posição final verifica-se uma perturbação, após isso a carga fica a oscilar até parar (Figura 5.18).

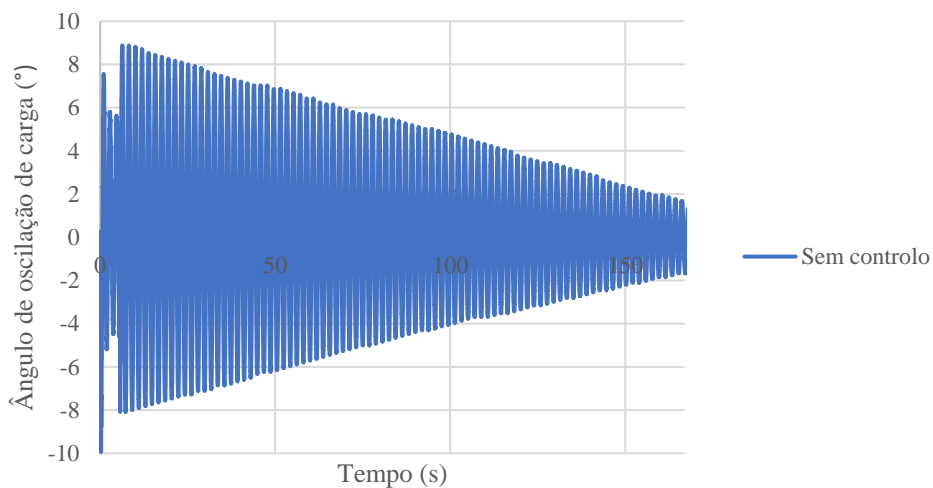


Figura 5.18 - Movimento sem controle

Verifica-se que o movimento com controle tem grande significância para a oscilação da carga, comparando os resultados com o movimento sem controle. Constata-se uma diferença de amplitude inicial de aproximadamente $[-8^{\circ}; 7^{\circ}]$. Durante o movimento do carro, em ambos, se observa uma diminuição do ângulo de oscilação. No entanto, no movimento sem controle, quando o carro chega à posição final, em 5,7 segundos, devido à inércia, o ângulo de oscilação aumenta, seguindo-se a sua diminuição durante minutos. No movimento controlado, o carro demora a acomodar 13,4 segundos, e nesse período de tempo o ângulo de oscilação está próximo de 0° , ao contrário do movimento sem controle, cujo valor é aproximadamente $7,3^{\circ}$.

Nas Figura 5.20 e Figura 5.19 estão expostos os resultados para a posição do carro e do comprimento do cabo.

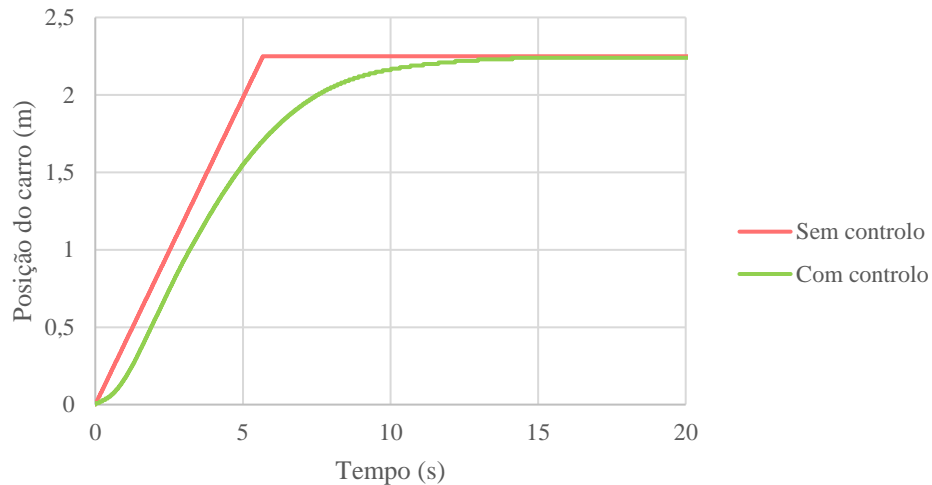


Figura 5.19 - Resultado da posição do carro, movimento com controle versus sem controle

Relativamente à posição do carro, observa-se que com o movimento sem controle é atingido o valor de posição desejado num menor tempo, o que era expectável. Isto ocorre porque a velocidade do motor é máxima ao longo de todo o movimento, obtendo-se um gráfico com reta linear. Ao contrário do movimento controlado, do qual o seu gráfico é representado por uma curva. Contudo, as diferenças entre estes tempos de acomodação são insignificantes tendo em conta os resultados do ângulo de oscilação da carga.

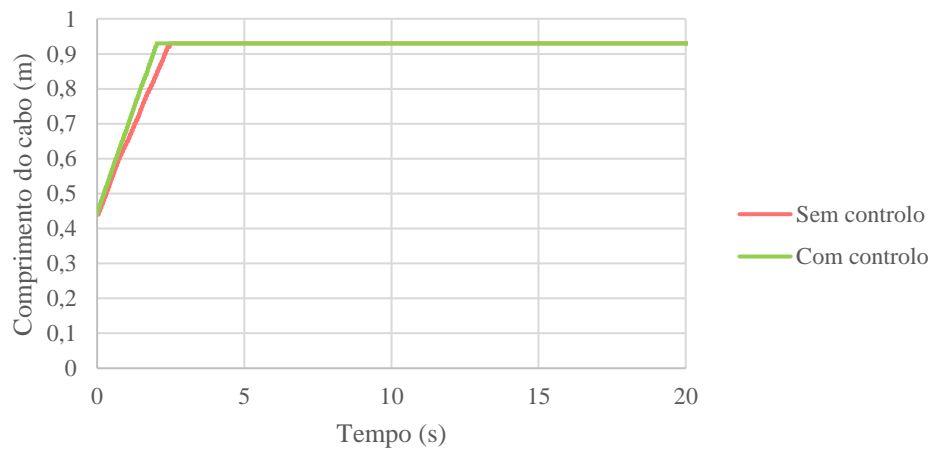


Figura 5.20 - Resultado do comprimento do cabo, movimento com controle versus sem controle

No caso do comprimento do cabo, verifica-se que no movimento com controle o valor de referência é alcançado ligeiramente em menor tempo (2,1 segundos), comparado com o movimento sem controle (2,5 segundos). Esta ligeira diferença pode dever-se ao facto de que uma vez que o ângulo de oscilação é 8° no movimento não controlado, o binário do motor poderá não ser suficientemente eficiente a variar o comprimento do cabo, havendo um menor desempenho do sistema mecânico. É importante referir que os valores de ganho utilizados para esta comparação foram os mesmos utilizados na simulação 4, mas a referência final do cabo está em 0,93 metros.

5.3. Controlo de Trajetória

Neste subcapítulo são demonstrados os testes e resultados nas Figura 5.21, Figura 5.22 e Figura 5.23 do seguimento da trajetória pré-definida para a carga suspensa.

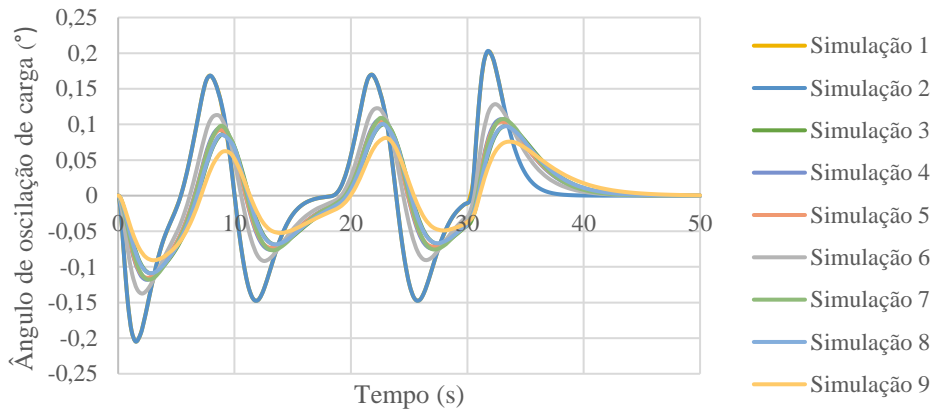


Figura 5.21 - Comparação de simulações do ângulo experimental para diferentes pólos para a trajetória

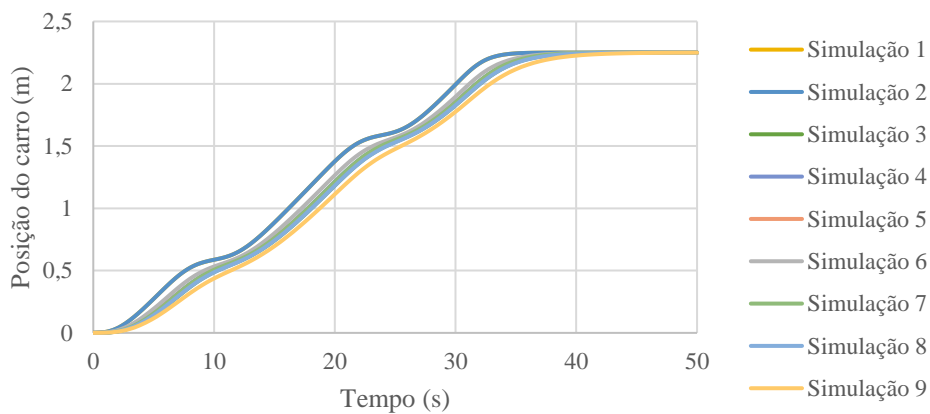


Figura 5.22 - Comparação de simulações da posição do carro para diferentes pólos para a trajetória

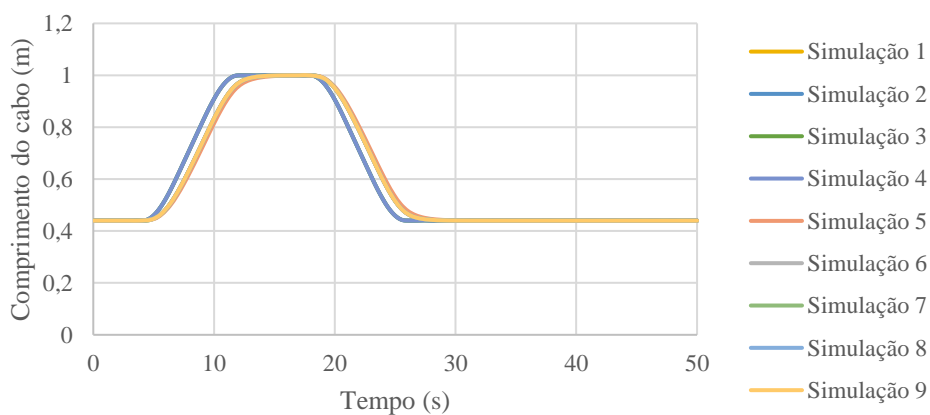


Figura 5.23 - Comparação de simulações do comprimento do cabo para diferentes pólos para a trajetória

Após análise dos gráficos, verificam-se as mesmas conclusões relativamente à localização dos pólos e sua influência na resposta do sistema. Nas simulações 1 e 2 verifica-se uma maior amplitude ângulo de oscilação de carga, mas menor tempo de acomodação, e menor tempo para a posição de carga. No caso do comprimento do cabo, para as simulações com tempo de estabilização mais rápido, o mesmo ocorre com controlo de trajetória.

De seguida, são apresentados dois dos testes experimentais realizados, tendo em conta os valores dos ganhos apresentados anteriormente.

Comparação 1: Teste Experimental 1

Desde a Figura 5.24 à Figura 5.31 são apresentados gráficos de comparação, entre simulação e realidade, da trajetória da carga suspensa, do ângulo de oscilação, da posição do carro e do comprimento do cabo.

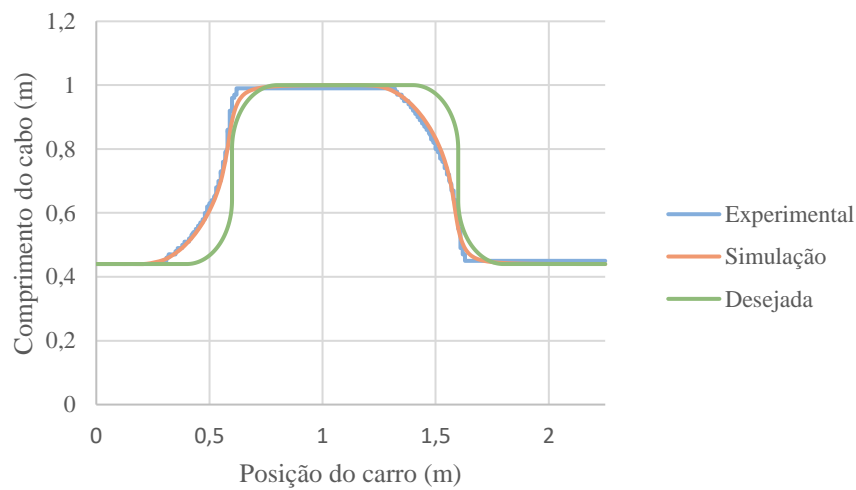


Figura 5.24 - Comparação do teste 1 da trajetória da carga, entre resultado simulado e experimental

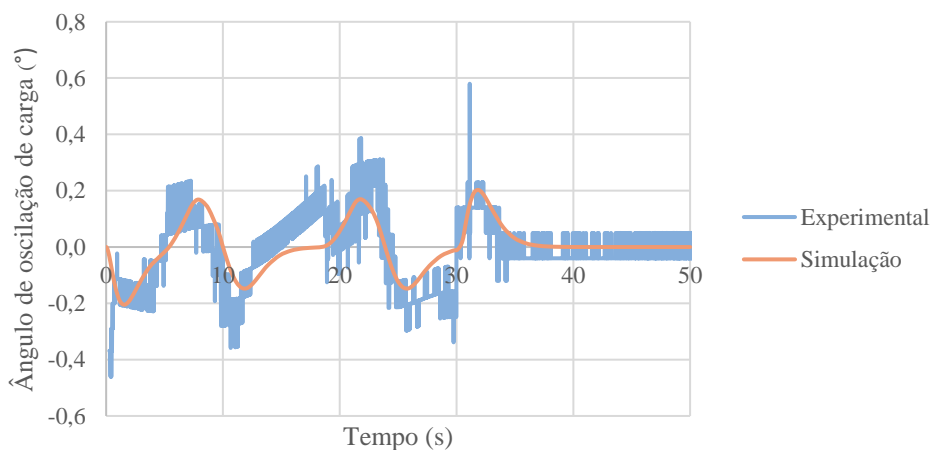


Figura 5.25 - Comparação do teste 1 do ângulo de oscilação, entre resultado simulado e experimental

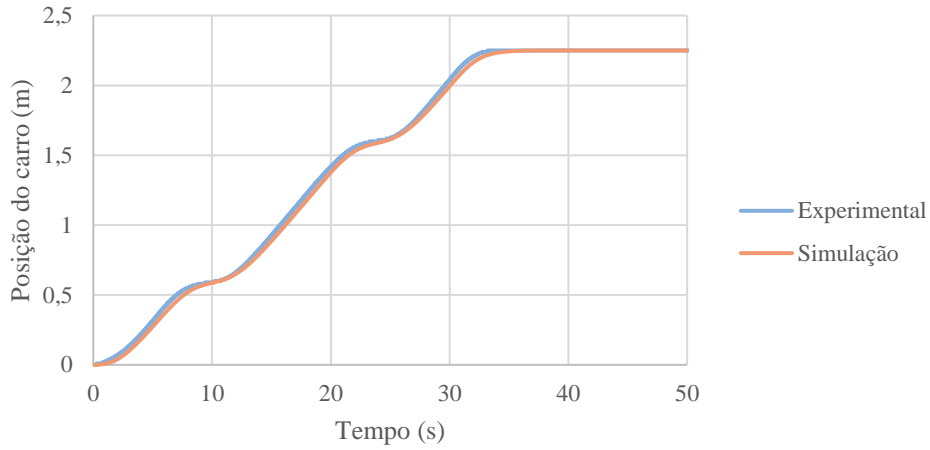


Figura 5.26 - Comparação do teste 1 da posição do carro, entre resultado simulado e experimental

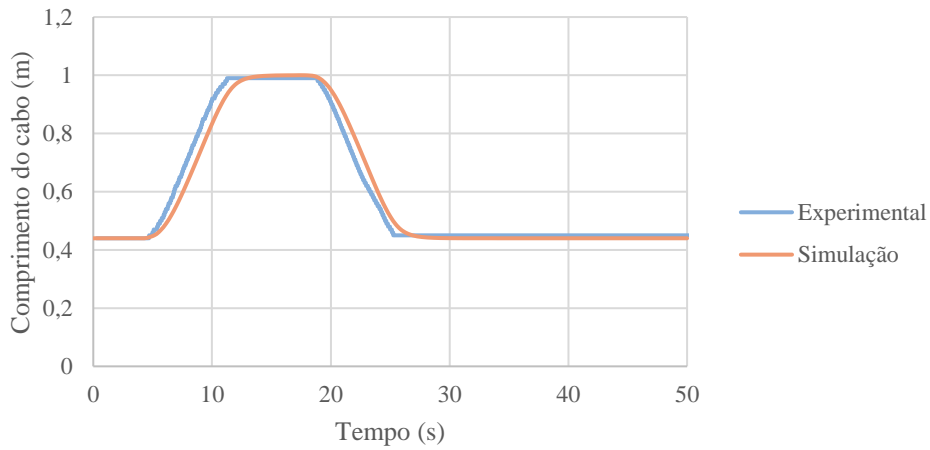


Figura 5.27 - Comparação do teste 1 do comprimento do cabo, entre resultado simulado e experimental

Comparação 2: Teste Experimental 2

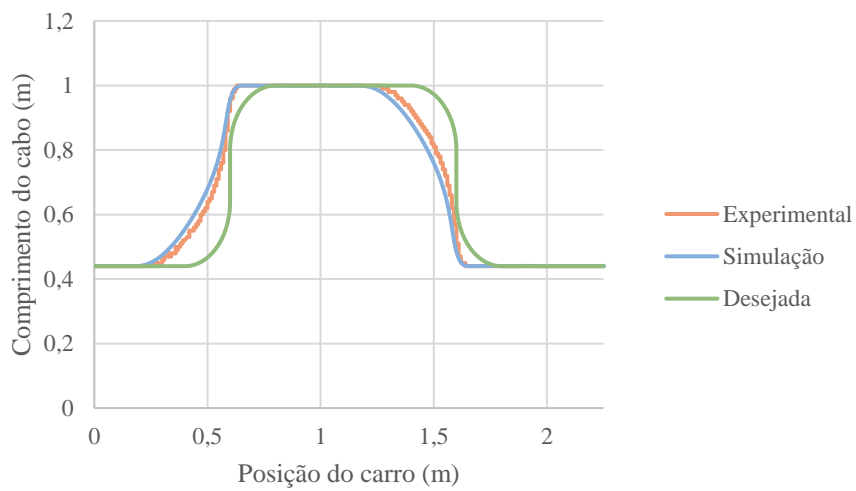


Figura 5.28 - Comparação do teste 2 da trajetória da carga, entre resultado simulado e experimental

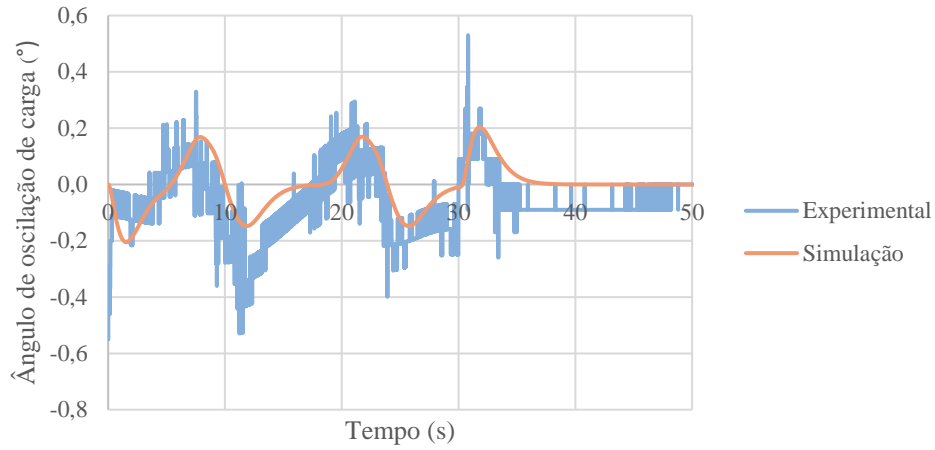


Figura 5.29 - Comparação do teste 2 do ângulo de oscilação, entre resultado simulado e experimental

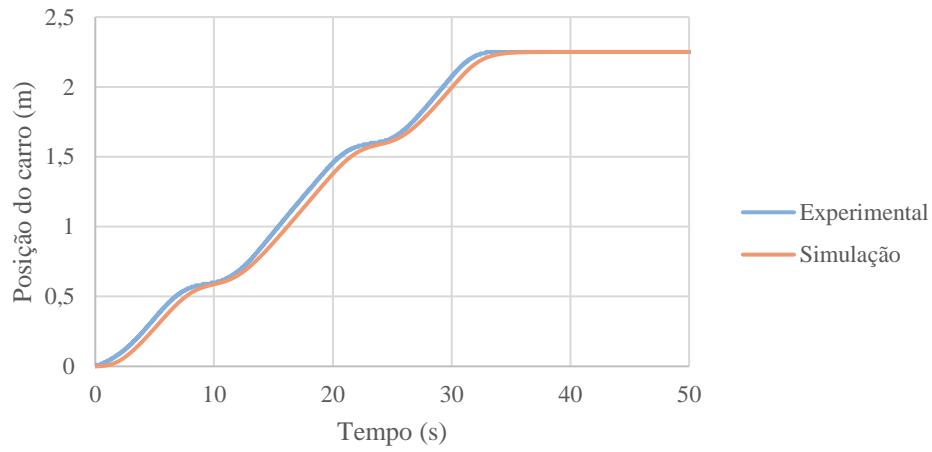


Figura 5.30 - Comparação do teste 2 da posição do carro, entre resultado simulado e experimental

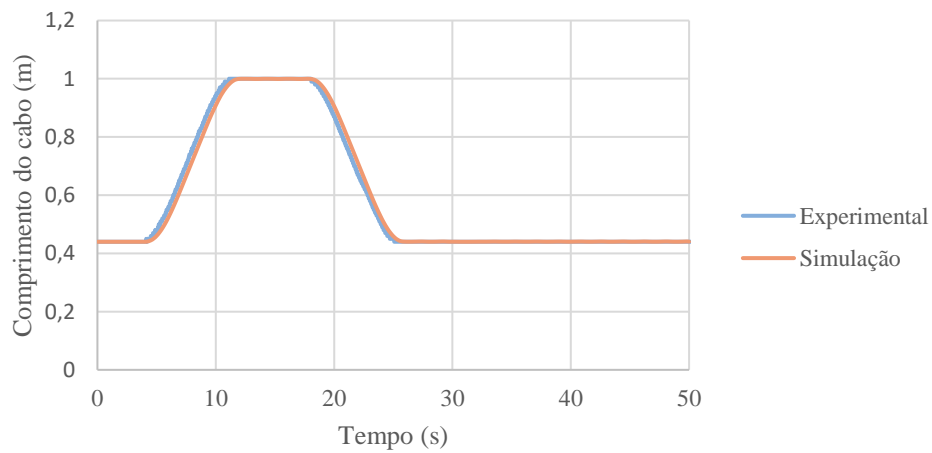


Figura 5.31 - Comparação do teste 2 do comprimento do cabo, entre resultado simulado e experimental

O teste experimental 1 é o que mais se assemelha à trajetória simulada, mas em ambos os testes há um afastamento em relação à trajetória desejada. Isto significa que o sistema de controle se ajusta de forma a atingir as referências desejadas finais, desviando-se da trajetória desejada, não seguindo rigorosamente a mesma, para reduzir as oscilações da carga. Relativamente à amplitude do ângulo de oscilação, e ignorando picos de medição, no primeiro teste está limitado a um intervalo de $[-0,46^\circ; 0,39^\circ]$ e no segundo um intervalo de $[-0,55^\circ; 0,28^\circ]$. Em ambos os testes, aos 35 segundos verifica-se um erro residual na leitura do ângulo de aproximadamente $0,1^\circ$. Analisando os resultados da variável de saída de posição do carro, no teste 2 existe maior afastamento relativamente ao teste 1, entre simulação e experimental. No que diz respeito ao comprimento do cabo, no caso prático ele tende a aumentar o tamanho mais rapidamente que na simulação, e por isso, nos testes experimentais 1 e 2 acomoda-se mais depressa comparando com as simulações. Contudo, no teste 2 as curvas entre simulação e experimental são mais semelhantes do que no teste 1, isto deve-se porque o ganho K_4 no teste 2 é superior, exigindo ao sistema um melhor desempenho para esta variável de saída.

5.4. Controlo com Modelo por Troços

Outra situação considerada foi a utilização de um modelo linear por troços, uma vez que o comprimento do cabo varia, o ponto de equilíbrio do modelo linear também varia. Consequentemente, os valores das matrizes de estado e de entrada em espaço de estados, as matrizes de ganho de realimentação de estados e do observador também variam, conforme o comprimento medido. Estas matrizes foram determinadas a partir do MATLAB, sendo que o programa está apresentado no Apêndice D. De referir que foram consideradas 13 classes para dividir o comprimento do cabo em troços.

Para analisar melhor o desempenho do controlador utilizando o modelo linear por troços, foi atribuída mais do que uma referência ao comprimento do cabo. O resultado está exposto na Figura 5.32.

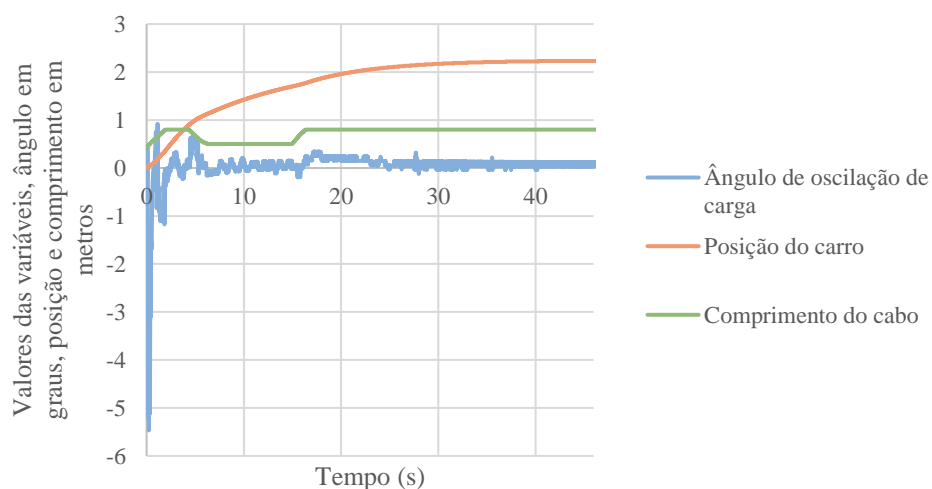


Figura 5.32 - Resultado experimental da aplicação do controlo considerado modelo linear por troços, aplicando várias referências ao comprimento do cabo

De modo a comparar este modelo com o modelo linearizado em torno de valor constante para o comprimento do cabo (1 metro), foi realizado o movimento anterior para este último. Os resultados estão apresentados na Figura 5.33.

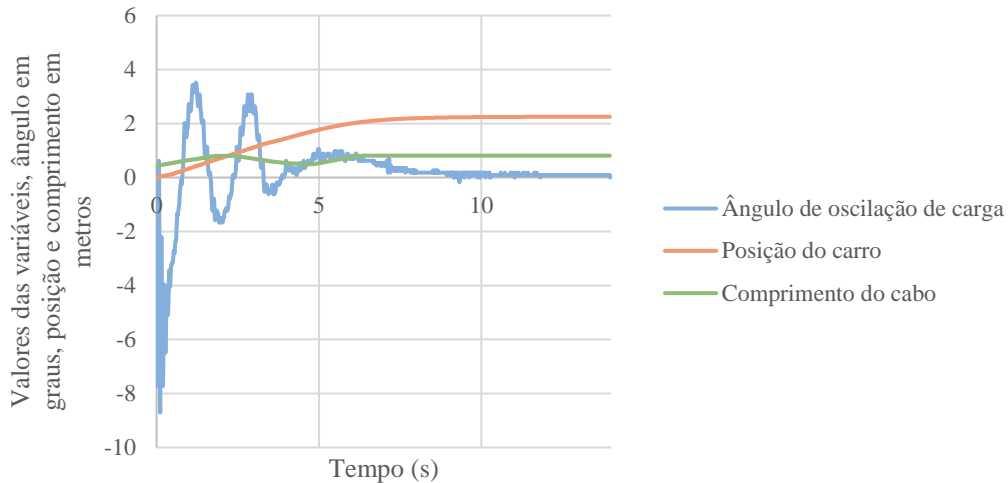


Figura 5.33 - Resultado experimental da aplicação do controle considerado modelo linear em torno do comprimento do cabo de 1 metro, aplicando várias referências ao comprimento do cabo

Pela leitura dos gráficos é possível afirmar que considerando um modelo por troços, o resultado do ângulo de oscilação de carga tem melhor desempenho, tendo um ângulo de oscilação com uma amplitude muito menor, entre $[-1,1^\circ;1^\circ]$, excluído o pico de medição inicial. Aproximadamente nos 28 segundos o ângulo é praticamente 0° . Em relação ao comprimento do cabo, o movimento é mais nítido e mais demorado, conseqüentemente a posição do carro tem maior tempo de acomodação, utilizando o modelo linear por troços.

5.5. Discussão de Resultados

Nos testes experimentais, confirmou-se uma observação consistente com o que foi descrito no enquadramento teórico: o observador de ordem reduzida é mais sensível a ruídos quando as variáveis de saída não são medidas de forma rigorosa. Desta forma, foi utilizado o sistema de controle de realimentação de estados com observador de ordem completa, com o método de colocação de pólos.

Verificou-se que a escolha dos pólos do sistema possui um impacto na resposta das variáveis de saída. Quanto mais afastados estiverem do eixo imaginário complexo, e com parte real negativa, maior será a amplitude do sinal e, portanto, menor será o tempo de acomodação das variáveis de saída. Contudo, maior será visível os efeitos de ruídos e perturbações. Para o movimento que é dada uma referência final à posição do carro e ao comprimento do cabo, foram apresentadas comparações gráficas entre a resposta do sistema em simulação e experimentalmente, para diferentes valores de ganho K . Constaram-se diferenças entre os diversos resultados obtidos, no entanto o tempo de acomodação era semelhante entre simulado e experimental.

De seguida, foi realizada uma análise do comportamento do sistema com e sem controle. Inicialmente, comparou-se o resultado simulado com o experimental, quando a carga é solta a 8° com o eixo vertical, verificando que os gráficos se sobrepõem. Numa fase seguinte, foram impostas as velocidades máximas tanto ao motor do carro como ao motor do cabo. De notar que em ambos, o movimento se iniciou com a carga suspensa com 0° de oscilação. Sem controle, o ângulo de oscilação vai diminuindo ao longo do movimento do carro, mas quando este atinge a posição final desejada, há um sobressalto e o ângulo volta a aumentar, ficando a oscilar até parar, (demorando mais do que 5 minutos). No movimento com controle, o ângulo está bastante próximo

de 0° a partir dos 10 segundos (com erro residual de $\pm 0,1^\circ$), e ao mesmo tempo o movimento não controlado tem oscilação de $7,3^\circ$. Constata-se uma diferença de amplitude inicial de aproximadamente $[-8^\circ; 7^\circ]$. Relativamente à variável de saída de posição do carro, sem controlo atinge a posição desejada em 5,7 segundos, e com controlo atinge em 13,4 segundos. No entanto, este resultado pode ser justificável pelo facto de que a entrada da velocidade do carro varia, conforme o controlo é realizado, enquanto no movimento não controlado a velocidade do carro é sempre a máxima. Além do mais, a comparação foi realizada com a matriz de ganho selecionada para o critério da rápida estabilização do comportamento do cabo, e não da posição do carro. A variável de saída do comprimento do cabo alcança o valor de referência em 2,5 segundos e em 2,1 segundos, para o movimento sem controlo e com controlo, respetivamente.

Nos testes experimentais do movimento com trajetória, os resultados são semelhantes entre o experimental e o simulado. Ainda, a trajetória desejada é sempre ligeiramente diferente da trajetória simulada e da trajetória experimental, isto porque o sistema de controlo compensa, de modo a reduzir o ângulo de oscilação e a atingir a referência final desejada para o valor de comprimento do cabo. Os limites do ângulo de oscilação são inferiores a $0,6^\circ$.

No último teste verificou-se a eficácia de considerar um modelo linear por troços, pois o ângulo de oscilação teve uma amplitude entre $[-1,1^\circ; 1^\circ]$, excluído o ruído inicial, cujo valores são muito inferiores ao modelo linear. Isto, porque o sistema é adaptado ao comprimento do cabo em determinado momento, acabando por se obter um melhor desempenho do sistema.

Analisando os gráficos do ângulo de oscilação, verifica-se um grande ruído, que pode ter várias causas. Uma vez que a resolução do *encoder* é de 12 bits, equivalente a $0,08$ graus, o erro associado à medição do ângulo de oscilação é aproximadamente de $0,08$ graus, o que pode contribuir para o ruído observado nos gráficos dos ângulos de oscilação. Além disso, o facto de que dependendo da posição do carro no carril, a leitura do ângulo varia, influencia os erros de leitura do mesmo. Isto está relacionado com a estrutura do carril, a discrepância de alturas ou desalinhamentos dos suportes do perfil vertical com o carril, e com a folga entre o carril e o carro. Outro fator que prejudica o desempenho do sistema de controlo é o motor, que se mostrou pouco eficiente. Por exemplo, a velocidade de subida e descida tem uma diferença de cerca de 10 m/s. Ao longo dos testes experimentais, verificou-se que se a velocidade do motor do cabo for muito grande perto da zona de referência, a carga sobe e desce muito rapidamente, e influencia o ângulo de oscilação da carga suspensa. Assim, foi necessário determinar retas para ajustar esse facto. Também o pequeno intervalo de velocidades do motor não é o ideal. As diferenças detetadas entre o modelo simulado e o real também é justificável pelo facto de que o comportamento do motor do cabo não ser tão linear na realidade, como é considerado na simulação. Estes últimos, também influenciam o comportamento do comprimento do cabo, assim como a utilização de peças plásticas leva a uma menor rigidez e suporte do peso da carga, quando é realizada a elevação e descida da carga.

De modo a melhorar o desempenho do controlo do sistema propõe-se as melhorias. Para melhorar o desempenho na elevação e descida da carga suspensa, utilizar um motor com mais binário, mais potente e robusto para o movimento do cabo. Assim, irá resultar numa maior eficiência no desempenho do sistema de controlo, e melhores resultados. De modo a obter medições do ângulo mais confiáveis, seria benéfico corrigir a leitura do ângulo a nível mecânico. Seria também relevante o estudo de alterar o posicionamento do sensor de fim de curso para o comprimento do cabo. Por exemplo fixado na peça por onde passa o cabo vindo do tambor. Por fim, substituir as peças de impressão 3D em plástico por peças maquinadas num material com maior rigidez e durabilidade.

6. Conclusão

Primeiramente, na elaboração do presente projeto foi importante fazer uma revisão da literatura relacionada com equipamentos de elevação e transporte de cargas suspensas, incluindo mecânica e sistemas de controlo.

Na etapa seguinte, foram realizadas alterações ao equipamento pré-existente no Laboratório de Robótica Avançada e Fábricas Inteligentes no Politécnico de Leiria. Através do *SolidWorks* foi realizado o projeto de equipamento de elevação e descida de uma carga suspensa com 0,5 kg. Seguindo-se a montagem do mesmo, no qual peças foram construídas através de impressão 3D, e outras adquiridas.

Foi realizada a modelação matemática do sistema simplificando, sendo representando em espaço de estados para implementação do sistema de controlo posterior. Uma vez que o modelo inicial era não linear foi importante realizar a linearização do mesmo, verificando que os modelos são semelhantes quando o comprimento do cabo que suspende a carga varia em torno de um ponto definido, neste caso 1 metro, que correspondia ao ponto médio do comprimento total do cabo.

Depois, através do MATLAB e do *Simulink* foi desenvolvido um programa para prever e analisar o comportamento do sistema, realizando simulações de diferentes métodos de controlo, tais como PID, realimentação de estados recorrendo ao método de colocação de pólos, e ainda, foi adicionado um observador, uma vez que nem todas as variáveis de estado poderiam ser medidas. Sendo definido o método de controlo a ser utilizado.

Com as alterações ao equipamento e ao modelo foi necessário adicionar componentes ao *hardware* e *software* já existente.

Posteriormente, foram realizadas simulações e testes experimentais para averiguar se o modelo obtido corresponde à realidade, se a resposta do sistema ao controlo implementado está de acordo com o esperado, e quais benefícios o sistema de controlo teve para o equipamento.

Constatou-se que na realidade o controlo com o observador de ordem completa era o mais adequado, assim foi utilizado esse.

Através das simulações realizadas selecionou-se a melhor matriz de ganho conforme as prioridades para o controlo do sistema. De forma geral, o tempo de acomodação das diferentes variáveis foi semelhante entre o simulado e o experimental, apesar das diferenças detetadas nos gráficos. Para o teste 6 aconselha-se o seu uso quando o objetivo é a estabilização da carga, e o posicionamento do carro. No caso do teste 4 aplica-se aquando da finalidade do sistema é o posicionamento rápido da carga numa determinada altura. Utilizando o método de colocação de pólos, verificou-se que o tempo de acomodação e a presença de ruídos depende da localização dos pólos desejados. Consequentemente, os resultados obtidos alteram conforme os valores dos ganhos do controlo. Desta forma, é importante existir um equilíbrio entre a velocidade da resposta e a sensibilidade a erros.

Comparado o movimento não controlado com o movimento controlado verificou-se uma diferença entre a acomodação do ângulo de oscilação da carga, a posição do carro e do comprimento do cabo. Para as três variáveis de saída verifica-se uma diferença de 7,4° (nos 10 segundos, existindo um erro residual de $\pm 0,1^\circ$), 7,7 segundos e 0,4 segundos, para o ângulo de oscilação, posição do carro e comprimento do cabo, respetivamente. De referir que o tempo de acomodação do carro foi inferior no movimento não controlado. Para as outras variáveis, o tempo de acomodação foi menor no movimento controlado.

De seguida, foi definida uma trajetória para carga suspensa, dividindo os troços numa equação paramétrica. Através dos ensaios realizados, observou-se que para a trajetória, os controlos experimentais e simulados são bastante idênticos, sendo que foi possível limitar o ângulo de oscilação a valores inferiores $0,6^\circ$.

Uma vez que, desde o início do projeto, o modelo do equipamento foi linearizado em torno do ponto médio do comprimento do cabo, considerou-se analisar o modelo linear por troços. Assim, à medida que o comprimento do cabo variasse, o modelo do equipamento acompanhava essa variação e era alterado o ponto em torno do qual o sistema era linearizado. Verificou-se que este modelo tem mais semelhanças com o modelo não linear do que com o modelo linear em torno de 1 metro de comprimento do cabo. De seguida, foi estudada esta alteração na prática. É possível afirmar que os resultados com o modelo por troços são melhores relativamente ao modelo linear em torno apenas de um ponto.

No último teste verificou-se a eficácia de considerar um modelo linear por troços, pois o ângulo de oscilação teve uma amplitude entre $[-1,1^\circ; 1^\circ]$, excluído o ruído inicial, valores que são muito inferiores ao modelo linear. Isto, porque o sistema é adaptado ao comprimento do cabo em determinado momento, acabando por se obter um melhor desempenho do sistema.

Em suma, conclui-se que os objetivos propostos foram atingidos. Foi possível alterar o equipamento de modo que a carga realizasse um movimento vertical, para se assemelhar às pontes rolantes reais. Foi também desenvolvido um sistema de controlo para controlar os movimentos do equipamento, verificando-se as respetivas vantagens. Desta forma, com a execução do presente projeto, espera-se a abertura de outros caminhos para futuros estudos sobre a movimentação de cargas.

6.1. Propostas de Trabalho Futuro

Durante o desenvolvimento do presente projeto, foram reconhecidas oportunidades para trabalhos futuros:

- Implementar as soluções apresentadas para melhoramento do desempenho do sistema;
- Considerar que o comprimento varia ao longo do tempo, ou seja, sistema linear dependente do tempo;
- Testar outras estratégias de controlo;
- Considerar que o cabo é flexível;
- Conceber um sistema para três dimensões;
- Desenvolver um sistema que permita a utilização de várias cargas, e que as possa transportar de um local para outro;
- Criar um programa que possibilite a inserção das referências pelo utilizado, e durante o movimento seja possível observar gráficos do movimento a ser realizado, assim como gravá-lo.

7. Referências Bibliográficas

- [1] L. Ramli, Z. Mohamed, A. M. Abdullahi, H. I. Jaafar, e I. M. Lazim, «Control strategies for crane systems: A comprehensive review», *Mechanical Systems and Signal Processing*, vol. 95, pp. 1–23, out. 2017.
- [2] M. Rasool Mojallizadeh, B. Brogliato, e C. Prieur, «Modeling and control of overhead cranes: a survey», INRIA Grenoble; UGA (Université Grenoble Alpes); GIPSA Lab; INP Grenoble; LJK/Grenoble University - INRIA, Research Report, out. 2022. Acedido: 7 de dezembro de 2022. [Em linha]. Disponível em: <https://hal.inria.fr/hal-03817363>
- [3] A. Khatamianfar e A. V. Savkin, «Real-Time Robust and Optimized Control of a 3D Overhead Crane System», *Sensors*, vol. 19, n.º 15, Art. n.º 15, jan. 2019.
- [4] H. Le Xuan, N. Thai, P. Van Hung, e L. Nguyen, «Consistency of Control Performance in 3D Overhead Cranes under Payload Mass Uncertainty», *Electronics*, vol. 9, p. 657, abr. 2020.
- [5] M. J. Maghsoudi, L. Ramli, S. Sudin, Z. Mohamed, A. R. Husain, e H. Wahid, «Improved unity magnitude input shaping scheme for sway control of an underactuated 3D overhead crane with hoisting», *Mechanical Systems and Signal Processing*, vol. 123, pp. 466–482, mai. 2019.
- [6] V. Caneiro, «Comando Avançado do Movimento de Cargas Suspensas», Instituto Politécnico de Leiria, Leiria, 2021.
- [7] L. Tuan, S. chan Moon, W. G. Lee, e S.-G. Lee, «Adaptive sliding mode control of overhead cranes with varying cable length», *Journal of Mechanical Science and Technology*, vol. 27, mar. 2013.
- [8] Q. Wu, X. Wang, L. Hua, e M. Xia, «Modeling and nonlinear sliding mode controls of double pendulum cranes considering distributed mass beams, varying roped length and external disturbances», *Mechanical Systems and Signal Processing*, vol. 158, p. 107756, set. 2021.
- [9] K. Alhazza, A. Mohammed, K. Alghanim, e Z. Masoud, «An Iterative Learning Control Technique for Point-to-Point Maneuvers Applied on an Overhead Crane», *Shock and Vibration*, vol. 2014, p. 11, mai. 2014.
- [10] S. Zhang, X. He, H. Zhu, X. Li, e X. Liu, «PID-like coupling control of underactuated overhead cranes with input constraints», *Mechanical Systems and Signal Processing*, vol. 178, p. 109274, out. 2022.
- [11] Q. Zhou, K. Wang, X. Xiong, e J. Zhao, «Optimization of Bridge Crane Control System Using Fuzzy PID Control and Speed Control of Frequency Converter», *Journal of Physics: Conference Series*, vol. 1802, p. 032007, mar. 2021.
- [12] H.-H. Lee, «A new design approach for the anti-swing trajectory control of overhead cranes with high-speed hoisting», *International Journal of Control - INT J CONTR*, vol. 77, pp. 931–940, jul. 2004.
- [13] L. Ramli, Z. Mohamed, M. Ö. Efe, I. M. Lazim, e H. I. Jaafar, «Efficient swing control of an overhead crane with simultaneous payload hoisting and external disturbances», *Mechanical Systems and Signal Processing*, vol. 135, p. 106326, jan. 2020.
- [14] L. Anh Tuan, A. Janchiv, G.-H. Kim, e S.-G. Lee, «Feedback linearization control of overhead cranes with varying cable length», em *2011 11th International Conference on Control, Automation and Systems*, out. 2011, pp. 906–911.

- [15] R. Miranda-Colorado, «Robust observer-based anti-swing control of 2D-crane systems with load hoisting-lowering», *Nonlinear Dyn*, vol. 104, n.º 4, pp. 3581–3596, jun. 2021.
- [16] Q. H. Ngo e K.-S. Hong, «Adaptive sliding-mode control of container cranes», *IET Control Theory and Applications*, vol. 6, pp. 662–668, mar. 2012.
- [17] T. W. Yang e W. J. O'Connor, «Wave Based Robust Control of a Crane System», em *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, out. 2006, pp. 2724–2729.
- [18] H. V. Pham *et al.*, «An Efficient Adaptive Fuzzy Hierarchical Sliding Mode Control Strategy for 6 Degrees of Freedom Overhead Crane», *Electronics*, vol. 11, n.º 5, Art. n.º 5, jan. 2022.
- [19] T. L. Nguyen, H. Q. Nguyen, e M. D. Duong, «Payload motion control for a varying length flexible gantry crane», *Automatika*, vol. 62, n.º 3–4, pp. 520–529, out. 2021.
- [20] K. T. Mohamed, M. H. Abdel-razak, E. H. Haraz, e A. A. Ata, «Fine tuning of a PID controller with inlet derivative filter using Pareto solution for gantry crane systems», *Alexandria Engineering Journal*, vol. 61, n.º 9, pp. 6659–6673, set. 2022.
- [21] M. H. Fatehi, M. Eghtesad, e R. Amjadifard, «Modelling and Control of an Overhead Crane System with a Flexible Cable and Large Swing Angle», *Journal of Low Frequency Noise, Vibration and Active Control*, vol. 33, pp. 395–410, dez. 2014.
- [22] M. Ali Mohammed, M. Maguire, e K. Kim, «Simulated Annealing Algorithm Based Tuning of LQR Controller for Overhead Crane», em *2020 13th International Conference on Developments in eSystems Engineering (DeSE)*, dez. 2020, pp. 37–42.
- [23] Y. Sun, J. Lyu, J. Fang, Z. Fu, e D. Dong, «Robust LQR Anti-Swing Control for Quay-Side Crane System with Variable Load», em *2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, jul. 2018, pp. 796–801.
- [24] J. Smoczek, «Fuzzy crane control with sensorless payload deflection feedback for vibration reduction», *Mechanical Systems and Signal Processing*, vol. 46, n.º 1, pp. 70–81, mai. 2014.
- [25] H.-H. Lee e S.-K. Cho, «A new fuzzy-logic anti-swing control for industrial three-dimensional overhead cranes», em *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation*, mai. 2001, pp. 2956–2961 vol.3.
- [26] A. Mary, A. Miry, T. Kara, e M. Miry, «Nonlinear state feedback controller combined with RBF for nonlinear underactuated overhead crane system», *Journal of Engineering Research*, vol. 9, n.º 3A, Art. n.º 3A, set. 2021, Acedido: 11 de dezembro de 2022. [Em linha]. Disponível em: <https://kuwaitjournals.org/jer/index.php/JER/article/view/9159>
- [27] G. Bartolini, A. Pisano, e E. Usai, «Second-order sliding-mode control of container cranes», *Automatica*, vol. 38, n.º 10, pp. 1783–1790, out. 2002.
- [28] H.-H. Lee, Y. Liang, e D. Segura, «A Sliding-Mode Antiswing Trajectory Control for Overhead Cranes With High-Speed Load Hoisting», *Journal of Dynamic Systems Measurement and Control-transactions of The Asme - J DYN SYST MEAS CONTR*, vol. 128, dez. 2006.
- [29] E. Q. Hussein, A. Q. Al-Dujaili, e A. R. Ajel, «Design of Sliding Mode Control for Overhead Crane Systems», *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 881, n.º 1, p. 012084, jul. 2020, doi: 10.1088/1757-899X/881/1/012084.

- [30] F. Debrouwere, M. Vukov, R. Quirynen, M. Diehl, e J. Swevers, «Experimental Validation of Combined Nonlinear Optimal Control and Estimation of na Overhead Crane». agosto de 2014.
- [31] G. O. Tysse, A. Cibicik, L. Tingelstad, e O. Egeland, «Lyapunov-based damping controller with nonlinear MPC control of payload position for a knuckle boom crane», *Automatica*, vol. 140, p. 110219, jun. 2022.
- [32] Z. Zhang, Y. Wu, e J. Huang, «Differential-flatness-based finite-time anti-swing control of underactuated crane systems», *Nonlinear Dynamics*, vol. 87, n.º 3, pp. 1749–1761, fev. 2017, doi: 10.1007/s11071-016-3149-7.
- [33] M. Karkoub e M. Zribi, «Robust Control Schemes for an Overhead Crane», *Journal of Vibration and Control - J VIB CONTROL*, vol. 7, pp. 395–416, mar. 2001.
- [34] L. A. Tuan, S.-G. Lee, D. H. Ko, e L. C. Nho, «Combined control with sliding mode and partial feedback linearization for 3D overhead cranes», *International Journal of Robust and Nonlinear Control*, vol. 24, n.º 18, pp. 3372–3386, 2014, doi: 10.1002/rnc.3061.
- [35] Y. Shi, Li, L. Bai, e Huang, «Research on Nonlinear Control Method of Underactuated Gantry Crane Based on Machine Vision Positioning», *Symmetry*, vol. 11, p. 987, ago. 2019.
- [36] H. Yavuz e S. Beller, «An intelligent serial connected hybrid control method for gantry cranes», *Mechanical Systems and Signal Processing*, vol. 146, p. 107011, jan. 2021.
- [37] J. D. de Deus, A. Noronha, M. Pimenta, T. Peña, e P. Brogueira, *Introdução à Física*, 2ª. Lisboa, Portugal: McGrawHill, 2000.
- [38] K. Ogata, *Engenharia de Controlo Moderno*, 5ª. São Paulo, Brasil: Pearson, 2010.
- [39] R. J. Vaccaro, *Digital Control: A State-Space Approach*. Nova Iorque, Estados Unidos da América: McGraw-Hill College, 1995.
- [40] N. S. Nise, *Control Systems Engineering*, 6ª. Wiley, 2010.
- [41] M. Johansson, «Piecewise Linear Control Systems», Doctoral Thesis (monograph), Department of Automatic Control, Lund Institute of Technology (LTH), 1999.
- [42] Neves, Carlos, Baptista, Diogo, Moreira, Miguel, e Caneiro, Vicente, «Movement of Suspended Loads: Dynamics System and Control Theory», inesc coimbra, jan. 2023.
- [43] trfastenings, «Thread geometry-metric coarse». Acedido: 25 de março de 2024. [Em linha]. Disponível em: <https://www.trfastenings.com/knowledge-base/thread-geometries/metric-coarse-standard>
- [44] alldatasheet, «L298 Datasheet». Acedido: 25 de fevereiro de 2024. [Em linha]. Disponível em: <http://www.alldatasheet.com/view.jsp>
- [45] «EMG30 - GearMotor with encoder | Motor DC com Engrenagens | PTR006083», PTRobotics. Acedido: 25 de fevereiro de 2024. [Em linha]. Disponível em: <https://www.ptrobotics.com/motor-dc-com-engrenagens/6083-emg30-gearmotor-with-encoder.html>
- [46] Broadcom, «AEAT-6012-A06». Acedido: 25 de fevereiro de 2024. [Em linha]. Disponível em: <https://www.broadcom.com/products/motion-control-encoders/magnetic-encoders/aeat-6012-a06>

Apêndices

Apêndice A – Programa em MATLAB para comparação dos modelos não linear, linear e linear por troços

Função de entrada do movimento do carro

```
function fc=fc_value(t)
fc=0;
end
```

Função de entrada do movimento do cabo do modelo não linear

```
function fl=f1_value(t,y,m,M,d1,dx,dt,fc,g)
%fl=-(-d1*y(6)*M*y(5)+m*(y(5)^2)*(y(4)^2)*M+g*m*cos(y(3))*M*y(5)-
m*sin(y(3))*fc*y(5)+m*sin(y(3))*dx*y(2)*y(5)-m*(sin(y(3))^2)*d1*y(6)*y(5)+
m*sin(y(3))*dt*y(4)*cos(y(3)))/(y(5)*M+y(5)*m*(sin(y(3)))^2);
fl=-5-800*sin(1*t);
end
```

Função de entrada do movimento do cabo do modelo linear

```
function fl_lin_value=f1_lin_value(y_lin,t)
%fl_lin_value=0;
fl_lin_value=-800*sin(1*t);
end
```

Função do valor do comprimento do cabo dividido em classes

```
function l=L_value(y_lin_pw)
l=y_lin_pw(5,:);
if l>0 && l<=0.1
    l=0.05;
elseif l>0.1 && l<=0.25
    l=0.175;
elseif l>0.25 && l<=0.4
    l=0.325;
elseif l>0.4 && l<=0.55
    l=0.475;
elseif l>0.55 && l<=0.7
    l=0.625;
elseif l>0.7 && l<=0.85
    l=0.775;
elseif l>0.85 && l<=1
    l=0.925;
elseif l>1 && l<=1.15
    l=1.075;
elseif l>1.15 && l<=1.3
    l=1.225;
elseif l>1.3 && l<=1.45
    l=1.375;
elseif l>1.45 && l<=1.6
    l=1.525;
elseif l>1.6 && l<=1.75
    l=1.675;
elseif l>1.75 && l<=1.9
    l=1.825;
elseif l>1.9 && l<=2.05
    l=1.975;
end
end
```

Função do modelo linear

```
function dy_lin=ylinearizado(y_lin,m,M,g,dx,dl,dt,L,fl,fc)
%Matriz de estado
A=[0 1 0 0 0 0;
    0 (-dx/M) (-m*g)/M (dt/(M*L)) 0 0;
    0 0 0 1 0 0;
    0 (dx/(M*L)) (-g/L) (-dt*(M+m))/(m*M*(L)^2) 0 0;
    0 0 0 0 0 1;
    0 0 0 0 0 (-dl/m)];
%Matriz de entrada
B=[0 0;1/M 0;0 0;-1/(M*L) 0;0 0;0 1/m];
u=[fc;fl];
dy_lin=A*y_lin+B*u;
end
```

Função do modelo linear com valor do comprimento do cabo dividido em classes

```
function dy_lin_pw=ylinearizado_pw(y_lin_pw,m,M,g,dx,dl,dt,L_value,fl,fc)
%Matriz de estado
A=[0 1 0 0 0 0;
    0 (-dx/M) (-m*g)/M (dt/(M*L_value)) 0 0;
    0 0 0 1 0 0;
    0 (dx/(M*L_value)) (-g/L_value) (-dt*(M+m))/(m*M*(L_value)^2) 0 0;
    0 0 0 0 0 1;
    0 0 0 0 0 (-dl/m)];
%Matriz de entrada
B=[0 0;1/M 0;0 0;-1/(M*L_value) 0;0 0;0 1/m];
u=[fc;fl];
dy_lin_pw=A*y_lin_pw+B*u;
end
```

Função do modelo não linear

```
function dy = modelo(y,m,M,g,dx,dl,dt,fl,fc)
dy(1,1) = y(2);
dy(2,1) = (fc*y(5)-dx*y(2)*y(5)-
fl*sin(y(3))*y(5)+dl*y(6)*sin(y(3))*y(5)+dt*y(4)*cos(y(3)))/(M*y(5));
dy(3,1) = y(4);
dy(4,1) = (-dt*y(4)*M-2*m*y(5)*y(6)*y(4)*M-m*g*y(5)*sin(y(3))*M-
m*y(5)*cos(y(3))*fc+m*y(5)*cos(y(3))*dx*y(2)+m*y(5)*cos(y(3))*fl*sin(y(3))-
m*y(5)*cos(y(3))*dl*y(6)*sin(y(3))-m*(cos(y(3))^2)*dt*y(4))/(m*M*(y(5))^2);
dy(5,1) = y(6);
dy(6,1) = (fl*M*y(5)-dl*y(6)*M*y(5)+m*(y(5)^2)*(y(4)^2)*M+g*m*cos(y(3))*M*y(5)-
m*sin(y(3))*fc*y(5)+m*sin(y(3))*dx*y(2)*y(5)+m*(sin(y(3))^2)*fl*y(5)-
m*(sin(y(3))^2)*dl*y(6)*y(5)-m*sin(y(3))*dt*y(4)*cos(y(3)))/(m*M*y(5));
end
```

Programa Principal

```
clc;clear;
%Valores das constantes
m=0.5;
M=6000;
g=9.8;
dx=600;
dt=0.007;
dl=7000;
L=1;
l=0.1;
```

```

%Intervalo de tempo de simulação
tspan=0:0.01:60;

%Condição inicial
y0=[0;0;5*pi/180;0;1;0];

%Modelo não linear
[t1,y] =
ode45(@(t1,y)modelo(y,m,M,g,dx,dl,dt,fl_value(t1,y,m,M,dl,dx,dt,fc_value,g),fc_value(
t1)),tspan,y0);

%Modelo linear
[t2,y_lin]=ode45(@(t2,y_lin)ylinearizado(y_lin,m,M,g,dx,dl,dt,L,fl_lin_value(y_lin,t2
),fc_value(t2)),tspan,y0);

%Modelo linear com comprimento do cabo dividido em classes (piecewise)
[t3,y_lin_pw]=ode45(@(t3,y_lin_pw)ylinearizado_pw(y_lin_pw,m,M,g,dx,dl,dt,L_value(y_l
in_pw),fl_lin_value(y_lin_pw,t3),fc_value(t3)),tspan,y0);

%Vetores correspondentes aos valores das variáveis obtidos na integração das funções
%Valor do ângulo de oscilação
ang_y=y(:,3);
ang_ylin=y_lin(:,3);
ang_ylinpw=y_lin_pw(:,3);
%Valor do comprimento do cabo
com_y=y(:,5);
com_ylin=y_lin(:,5);
com_ylinpw=y_lin_pw(:,5);
%Valor do deslocamento do carro
pos_y=y(:,1);
pos_ylin=y_lin(:,1);
pos_ylinpw=y_lin_pw(:,1);

%Gráfico Ângulo de Oscilação
subplot(2,1,1)
plot(t1,ang_y*180/pi,t1,ang_ylin*180/pi,t1,ang_ylinpw*180/pi,'LineWidth',1.5)
legend('Modelo não linear','Modelo linear','Modelo linear com L por troços')
title('Modelo Não Linear vs Modelo Linear vs Modelo Linear com L por troços: Ângulo
de Oscilação')
xlabel('Tempo (s)')
ylabel('Ângulo \theta (°)')
grid on

%Gráfico Comprimento do Cabo
subplot(2,1,2)
plot(t1,com_y,t1,com_ylin,t1,com_ylinpw,'LineWidth',1.5)
legend('Modelo não linear','Modelo linear','Modelo linear com L por troços')
title('Modelo Não Linear vs Modelo Linear vs Modelo Linear com L por troços:
Comprimento do Cabo')
xlabel('Tempo (s)')
ylabel('Comprimento l (m)')
grid on

```

Apêndice B – Programa em MATLAB para simulação dos sistemas de controlo

```

clc;clear;

%Valores das constantes dos modelos
m = 0.5;
M = 6000;
L = 1;
g = 9.8;
dt = 0.007;
dx = 600;
dl = 7000;

%Modelo linear com entrada em velocidade
A=[((-dt*(M+m))/(m*M*(L^2))+dt/(M*(L^2))) 1 0 0;((-g/L)-((m*g)/(M*L))) 0 0 0;0 0 0 0;0 0 0 0];
B=[-1/L 0;0 0;1 0;0 1];
C=[0 0 1 0;1 0 0 0;0 0 0 1];
Cl=[1 0 1 1];
D=0;
Modelo_Equipamento=ss(A,B,C,D);
tf(Modelo_Equipamento);

%Controlabilidade do Sistema
Wc = ctrb(A,B);
car_Wc=rank(Wc);

%Observabilidade do Sistema
Wo = obsv(A,Cl);
car_Wo=rank(Wo);

%Polos/Zeros do Sistema
polos = pole(Modelo_Equipamento);
figure(1);
plot(real(polos), imag(polos),'x');
xlabel('Parte Real');
ylabel('Parte Imaginária');
title('Diagrama de Polos do Sistema');
grid on;
xlim([-0.01 0.005]);

%Determinação de K para controlo em Colocação de Polos
p = [-10, -0.8, -0.9, -0.5];
K = place(A,B,p);
Polos = eig(A-B*K);

%Observador de ordem completa
A13 = [((-dt*(M+m))/(m*M*(L^2))+dt/(M*(L^2))) 1 0;((-g/L)-((m*g)/(M*L))) 0 0;0 0 0 0];
B13 = [-1/L 0;0 0;1 0];
C13 = [1 0 1];
C4 = [0 0 0;0 0 0;0 0 0;0 0 1];
W13 = obsv(A13,C13);
car_Wo13=rank(W13);
q = [-6.56057571 -4.374292295 -4.709160753];
L3 = place(A13',C13',q);

%Observador de ordem reduzida
sys_A=[((-dt*(M+m))/(m*M*(L^2))+dt/(M*(L^2))) 1 0;((-g/L)-((m*g)/(M*L))) 0 0;0 0 0 0];
sys_B=[-1/L 0;0 0;1 0];
sys_C=[1 0 0 0;0 1 0 0;0 0 1 0];
Amm=[((-dt*(M+m))/(m*M*(L^2))+dt/(M*(L^2))) 0;0 0];

```

```

Amo=[1;0];
Aom=[((-g/L)-((m*g)/(M*L))) 0];
Aoo=0;
Bm=[-1/L;1];
Bo=[0];
L1=10;L2=15;
L_obs=[L1 L2];
A_obs=-L1;
B_obs=[((-g/L)-((m*g)/(M*L)))+L1*((-dt*(M+m))/(m*M*(L^2))+dt/(M*(L^2)))-L1^2 -L1*L2];
F_obs=L2-L1/L;

```

%Discretização do sistema

```

M = [((-dt*(M+m))/(m*M*(L^2))+dt/(M*(L^2))) 1 0 0 -1/L 0;
      ((-g/L)-((m*g)/(M*L))) 0 0 0 0 0;
      0 0 0 0 1 0;
      0 0 0 0 0 1;
      0 0 0 0 0 0;
      0 0 0 0 0 0];
T = 0.01; %segundos
sysd = c2d(Modelo_Equipamento,T);
A_disc = [0.9994 0.009998 0 0;-0.09799 0.9995 0 0;0 0 1 0;0 0 0 1];
B_disc = [-0.009998 0;0.00049 0;0.01 0;0 0.01];
C_disc = [1 0 0 0;0 1 0 0;0 0 1 0;0 0 0 1];
D_disc = [0 0;0 0;0 0];
p_disc = [0.9048,0.9891,0.9512,0.9231];
K_disc = place(A_disc,B_disc,p_disc);
W_disc = obsv(A_disc,C_disc);
car_W_disc=rank(W_disc);

```

%Observador de ordem reduzida sistema discreto

%Sistema desacoplado discreto

```

sysd_A=[0.9994 0 0.009998;0 1 0;-0.09799 0 0.9995];
sysd_B=[-0.009998;0.01;0.00049];
sysd_C=[1 0 0;0 1 0];
sysd_D=[0;0];
K_sys=place(sysd_A,sysd_B,[0.9048,0.9891,0.9512]);
val_prop=eig(sysd_A);
sysd_Amm=[0.9994 0;0 1];
sysd_Amo=[0.009998;0];
sysd_Aom=[-0.09799 0];
sysd_Aoo=[0.9995];
sysd_Bm=[-0.009998;0.01];
sysd_Bo=[0.00049];
W_sysd=obsv(sysd_A,sysd_C);
car_W_sysd=rank(W_sysd);
%Sistema acoplado discreto
sysda_A=[0.9994 0.009998 0 0;-0.09799 0.9995 0 0;0 0 1 0;0 0 0 1];
sysda_B=[-0.009998 0;0.00049 0;0.01 0;0 0.01];
sysda_C=[0 0 1 0;1 0 0 0;0 0 0 1];
sysda_D=[0 0;0 0;0 0];
K_sysa=place(sysda_A,sysda_B,[0.98906,0.988072,0.987084,0.986098]);

```

%Condições iniciais do sistema

```

x(:,1)=[0;0;0];
y(:,1)=sysd_C*x(:,1);
xa(:,1)=[0;0;0;1];
ya(:,1)=sysda_C*xa(:,1);

```

%Valor inicial da entrada

```

u(:,1)=0;
ua(:,1)=[0;0];

```

%Condição inicial do observador

```

xh(1)=0;
%Referência
ref=[0;2.25;0];
refa=[0;2.25;0;1];

N=500;%Duração da simulação

for k=1:N-1
    x(:,k+1)=sysd_A*x(:,k)+sysd_B*u(:,k);
    y(:,k+1)=sysd_C*x(:,k+1);
    xa(:,k+1)=sysda_A*xa(:,k)+sysda_B*ua(:,k);
    ya(:,k+1)=sysda_C*xa(:,k+1);

%Simulação do observador de ordem reduzida
g1=10;g2=15;
xh(k+1)=(sysd_Aoo-[g1 g2]*sysd_Amo)*xh(k)+(sysd_Bo-[g1 g2]*sysd_Bm)*u(:,k)+[g1
g2]*y(:,k+1)+(sysd_Aom-[g1 g2]*sysd_Amm)*y(:,k);

%Cálculo da nova entrada
xob=[x(1,k+1);x(2,k+1);xh(k+1)];
u(:,k+1)=K_sys*(ref-xob);
xsys=[xob;xa(4,k+1)];
ua(:,k+1)=K_sysa*(refa-xsys);
end

figure(2)
plot(1:N,x(3,:),1:N,xh);
legend('Variável de Estado do Sistema','Variável de Estado do Observador')
xlabel('Tempo (s)')

%Trajetória
XPOS=[0.4 0.6 0.8 1.4 1.6 1.8 2.25];
YPOS=[0.6 0.8 1];
V=0.1;
R=0.2;
t=0:0.01:50;
[xx,yy]=trajetoria(XPOS,YPOS,V,R,t);
figure(3)
plot(xx,yy);
title('Trajetória')
xlabel('Posição do carro')
ylabel('Comprimento do cabo')
ylim([-0.5, 2])
x=0;y=0.4;

```

Função de cálculo da posição no eixo xx e no eixo yy da carga suspensa para realizar a trajetória desejada

```

function [x,y]=trajetoria(XPOS,YPOS,V,R,t)
w=V/R;
%cálculo dos ti
t1=0;
t2=XPOS(1)/V;
t3=t2+pi/(2*w);
t4=t3+(YPOS(2)-YPOS(1))/V;
t5=t4+pi/(2*w);
t6=t5+(XPOS(4)-XPOS(3))/V;
t7=t6+pi/(2*w);
t8=t7+(YPOS(2)-YPOS(1))/V;
t9=t8+pi/(2*w);
tf=t9+(XPOS(7)-XPOS(6))/V;
%cálculo da trajetória
x=zeros(length(t),1);

```

```

y=zeros(length(t),1);
for i=1:length(t)
    if(t(i)<t2)%tramo 1
        x(i)=V*t(i);
        y(i)=0.4;
    elseif(t(i)<t3)%tramo2
        x(i)=XPOS(1)+R*cos(-pi/2+w*(t(i)-t2));
        y(i)=0.4+R+R*sin(-pi/2+w*(t(i)-t2));
    elseif(t(i)<t4)%tramo 3
        x(i)=XPOS(2);
        y(i)=YPOS(1)+V*(t(i)-t3);
    elseif(t(i)<t5)%tramo4
        x(i)=XPOS(2)+R+R*cos(pi-w*(t(i)-t4));
        y(i)=YPOS(2)+R*sin(pi-w*(t(i)-t4));
    elseif(t(i)<t6)%tramo5
        x(i)=XPOS(3)+V*(t(i)-t5);
        y(i)=YPOS(3);
    elseif(t(i)<t7)%tramo6
        x(i)=XPOS(4)+R*cos(pi/2-w*(t(i)-t6));
        y(i)=YPOS(3)-R+R*sin(pi/2-w*(t(i)-t6));
    elseif(t(i)<t8)%tramo7
        x(i)=XPOS(5);
        y(i)=YPOS(2)-V*(t(i)-t7);
    elseif(t(i)<t9)%tramo8
        x(i)=XPOS(5)+R+R*cos(pi+w*(t(i)-t8));
        y(i)=YPOS(1)+R*sin(pi+w*(t(i)-t8));
    elseif(t(i)<tf)%tramo9
        x(i)=XPOS(6)+V*(t(i)-t9);
        y(i)=0.4;
    else
        x(i)=XPOS(7);
        y(i)=0.4;
    end
end

end

%exportar dados
T = table(out.ScopeData.time, out.ScopeData.signals.values)
writetable(T, "ScopeData.xls")

```

Apêndice C – Diagrama de blocos no *Simulink* do sistema de controlo para um movimento com uma trajetória pré-definida da carga suspensa

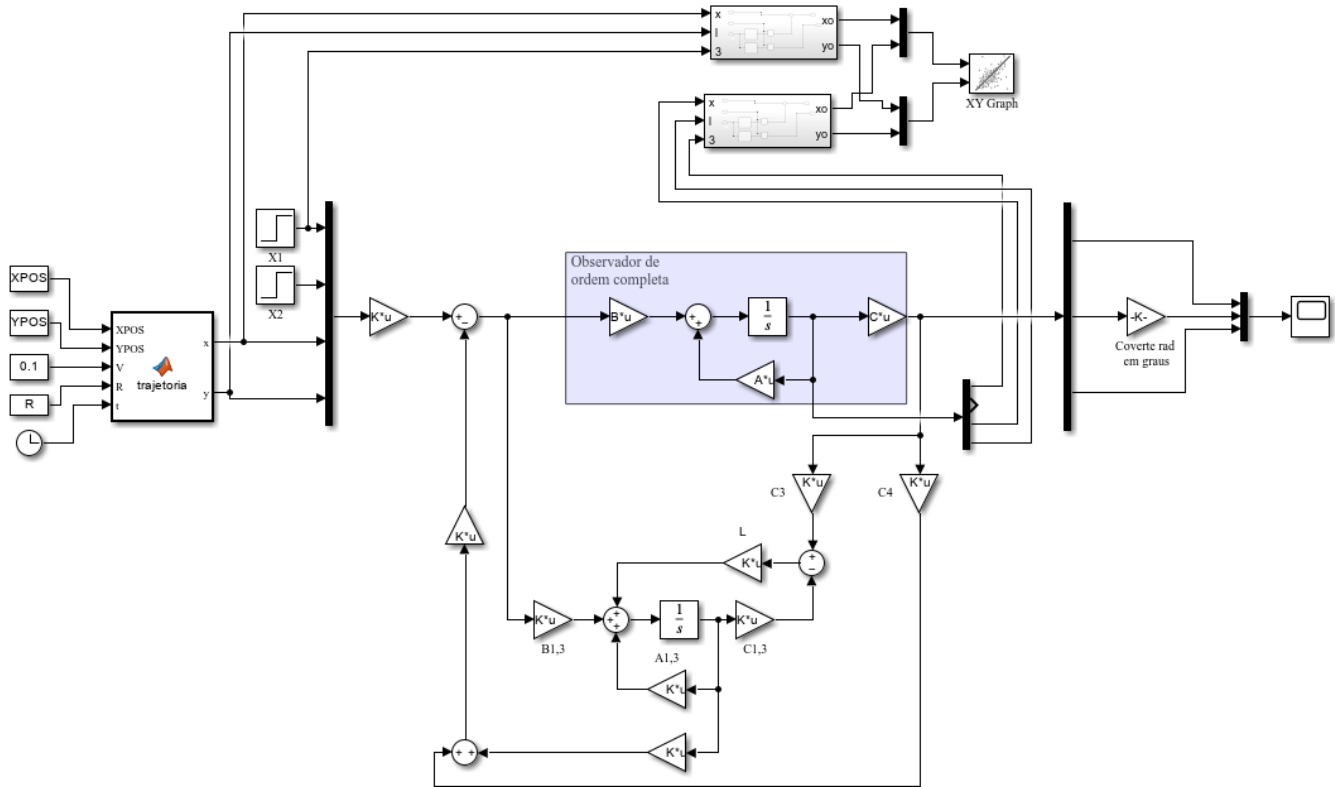


Figura C.1 - Diagrama de blocos do sistema de controlo com realimentação de estados e observador de ordem completa, utilizando o método de colocação de pólos, com referência de uma trajetória pré-definida no Simulink

Apêndice D – Programa em MATLAB para determinar as matrizes para comprimento do cabo dividido em troços

```

clc;clear;

%Valores das constantes
m=0.5;
M=6000;
g=9.8;
dt=0.007;
dx=600;
dl=7000;
T=0.01;

N=14;
index=0;

for L=0.175:0.15:1.975
index=index+1;

%Modelo linear
A=[((-dt*(M+m))/(m*M*(L^2))+dt/(M*(L^2))) 1 0 0;((-g/L)-((m*g)/(M*L))) 0 0 0;0 0 0
0;0 0 0 0];
B=[-1/L 0;0 0;1 0;0 1];
C=[0 0 1 0;1 0 0 0;0 0 0 1];
D=0;
Modelo_Equipamento=ss(A,B,C,D);

%Modelo discreto
sysd=c2d(Modelo_Equipamento,T);
A_disc=sysd.A;
A_disc_matrix(index,1:6)=[A_disc(1,1) A_disc(1,2) A_disc(2,1) A_disc(2,2) A_disc(3,3)
A_disc(4,4)];
B_disc=sysd.B;
B_disc_matrix(index,1:4)=[B_disc(1,1) B_disc(2,1) B_disc(3,1) B_disc(4,2)];
A13_disc=A_disc(1:3,1:3);
B13_disc=B_disc(1:3,1);
C13_disc=[1 0 1];

%Determinar matriz resultante
K_disc=place(A_disc,B_disc,[0.9048,0.9891,0.9512,0.9231]);
L_disc=place(A13_disc',C13_disc',[0.82 0.96 0.89]);
result_matrix(index,1) = L;
result_matrix(index,2:5) = [K_disc(1,1) K_disc(1,2) K_disc(1,3) K_disc(2,4)];
result_matrix(index,6:8) = place(A13_disc',C13_disc',[0.82 0.96 0.89]);

end

```

Apêndice E – Esquema de ligações elétricas

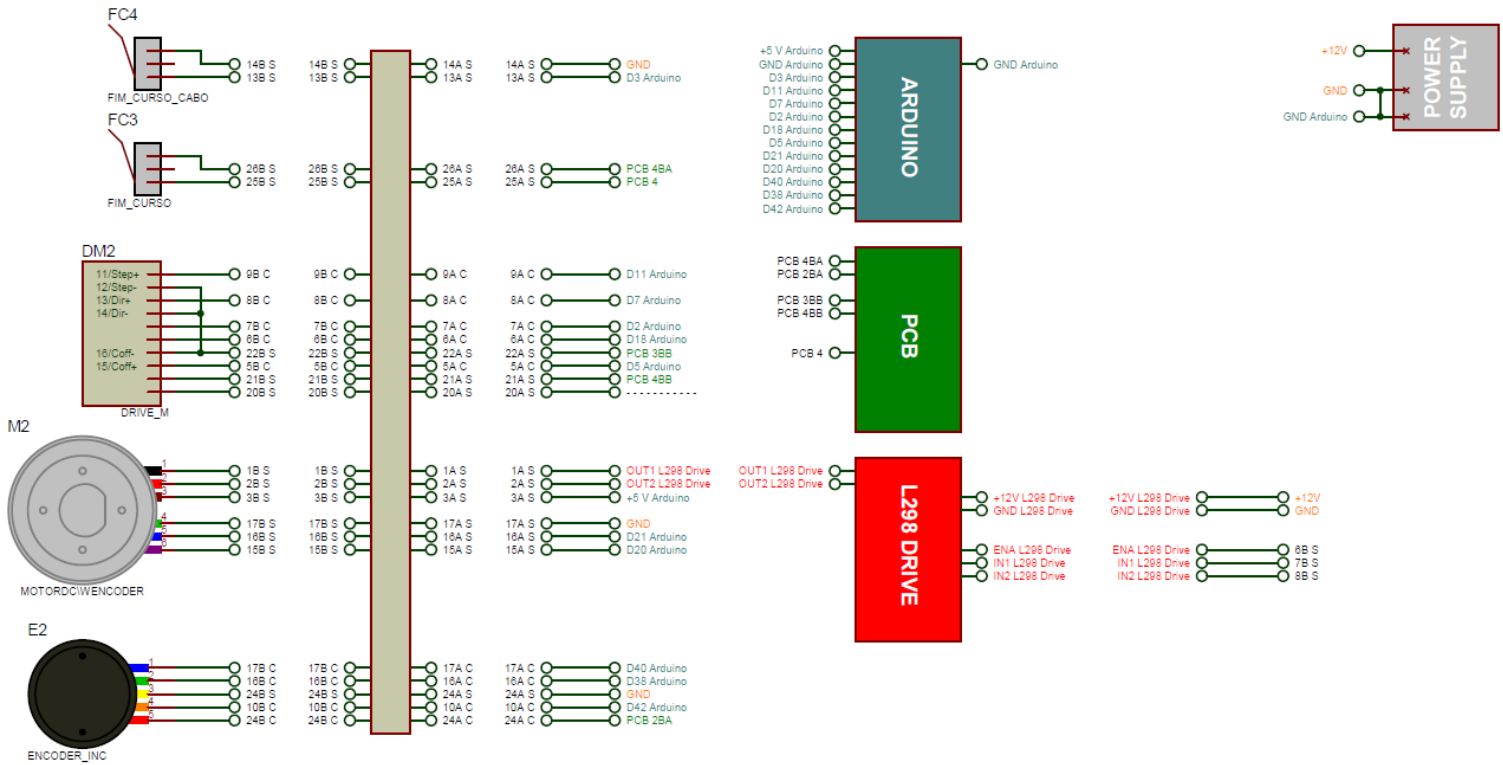


Figura E.1 - Esquema de ligações elétricas

Apêndice F – Grafset de nível 1 do funcionamento do software

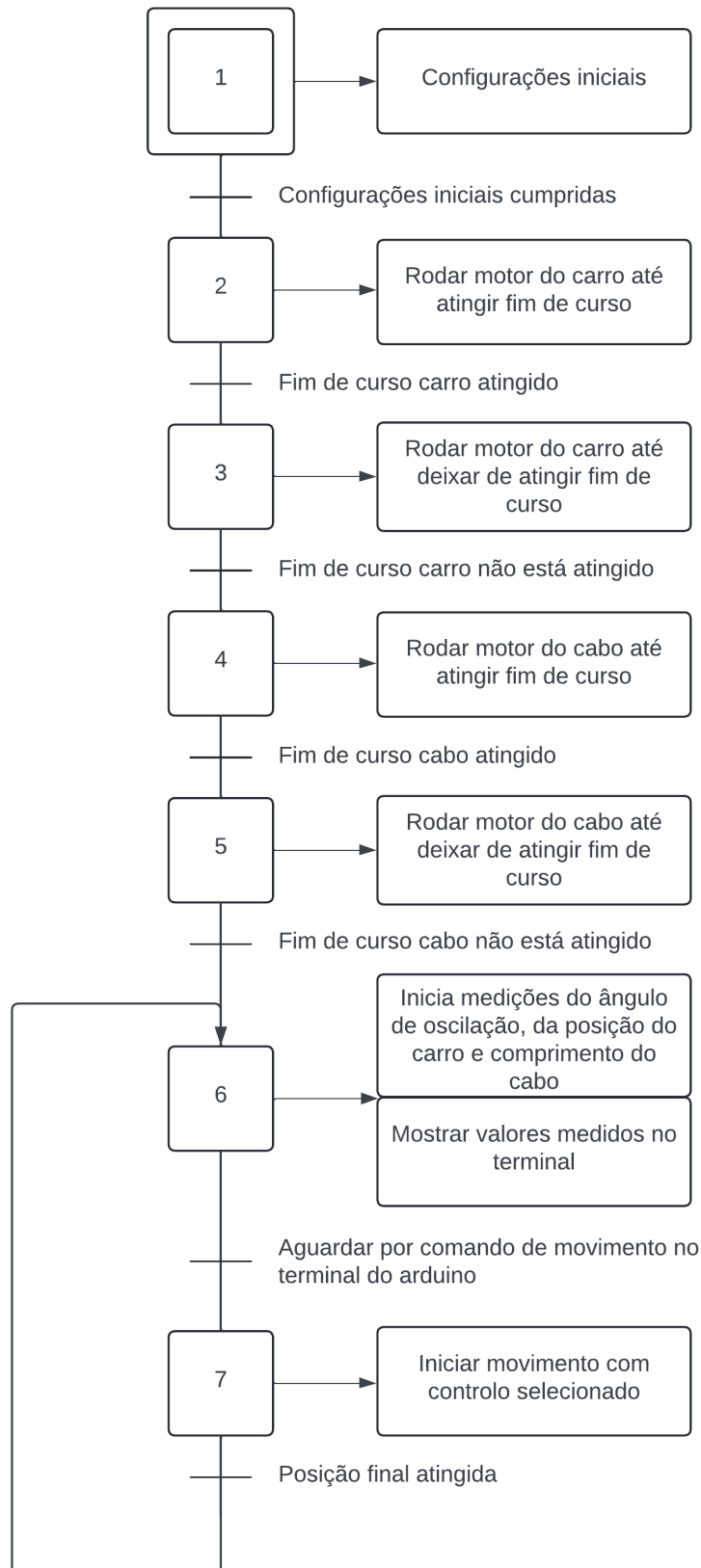


Figura F.1 - Grafset de nível 1 simplificado do funcionamento do software

Apêndice G – Programa *Arduino*

```

//Bibliotecas necessárias

#include <TimerThree.h> //Biblioteca para timer 3
#include <TimerOne.h> //Biblioteca para timer 1
#include <arduino.h> //Biblioteca geral do arduino
#include <CircularBuffer.h>
#include <math.h>

//Definições necessárias

//Pinos de interrupt para ArduinoMega2560 são: 2, 3, 18, 19, 20, 21

#define N 6

#define PinoDirecao 7 //Definição de variável PinoDirecao para o número 1 não
necessita de interrupt, ou seja, pode ser em qualquer pino)

#define PinoSalto 11 //Definição de variável PinoDSalto para o número 4 (Pino
associado em hardware ao canal A do Timer 1)

#define Enable 5 //Definição de variável Enable para o número 5 (não necessita
de interrupt, ou seja, pode ser em qualquer pino)

#define FIM_DE_CURSO 6 //Definição de variável FIM_DE_CURSO para o número 6
(não necessita de interrupt, ou seja, pode ser em qualquer pino)

#define CANAL_A 18 //Canal A do encoder no 3 (pino com interrupt)

#define CANAL_B 2 //Canal B do encoder no 2 (pino com interrupt)

#define TPROSS 10 //Pino para marcar por hardware(osciloscópio)o tempo de
processamento

#define CSn 42 //Pino de comunicação com o encoder de ângulo (chip select)

#define DO 40 //Pino de comunicação com o encoder de ângulo (Data Out)

#define Clk 38 //Pino de comunicação com o encoder de ângulo (Clock)

#define MOV_STOP 0

#define MOV_CTRL 1

#define MOV_NOCTRL 2

#define MOV_CTRL_OBS 3

#define MOV_CTRL_TRAJ 4

#define MOV_CTRL_REDUZ 5

#define MOV_CTRL_L 6

#define MOV_UPDOWN 7

#define FIM_DE_CURSO_CABO 3 //Definição de variável FIM_DE_CURSO_CABO para o
número 3 (não necessita de interrupt, ou seja, pode ser em qualquer pino)

```

```
//definições variáveis pinos para a leitura do teclado
#define col1 48
#define col2 50
#define col3 46
#define lin1 49
#define lin2 44
#define lin3 45
#define lin4 47

#define PI 3.14159265359

//Definir as funções paralelas ao loop principal
void tempo_amostragem(); //Função tempo de amostragem: gera uma flag a partir
do interrupt associado ao timer 3

void CANAL_CONTAR ); //Função canal contar: implementa a lógica de contagem
dos pulsos do encoder

int read_ssi_enc(); //Lê os valores do encoder de ângulo
char read_keyb(void); //Lê o teclado matricial externo

//Definir as variáveis globais necessárias
long dt = 10000;
//Tempo de amostragem em microssegundos, ou seja o tempo que queremos que a
flag passe a 1 no loop geral

long flag = 0;
//Definir uma variável que nos vai servir de flag para o interrupt

long x = 0;
//Definição de uma variável que muda para iniciar o controlo

long contador, A, B, Last_A, Last_B, ref_pos = 39999;
// 17777 é 1 metro

String spc = " "; //definição de uma string para controlar a leitura

int val;

float V = 0.138, graus;

float Posicao=0,saida_real,x_1=0.0,x_2=0.0,x_3=0.0,x_4=0.0,saida_estimada,
saida_total,x_estimado_anterior_1=0.0,x_estimado_anterior_2=0.0,
x_estimado_anterior_3=0.0,referencia,referencia_cabo,entrada,entrada_carro,
entrada_cabo,x_estimado_1,x_estimado_2,x_estimado_3,entrada_anterior,
x_futuro_estimado_1,x_futuro_estimado_2,x_futuro_estimado_3;

int movtype = 0;

int vel_cabo; //Variável de velocidade do motor do cabo

const int botaoDesenrolar = 24; //Pino do outro botão (Descer)
const int botaoEnrolar = 25; //Pino do botão (Subir)
```

```
const int encoderPinA = 20; //Pino digital 20 (interrupção 3) (encoder motor)
const int encoderPinB = 21; //Pino digital 21 (encoder motor)
int In1_Motor_Cabo = 27; //Pino digital Direção In2
int In2_Motor_Cabo = 28; //Pino digital Direção In1
int Pwm_Motor_Cabo = 13; //Pino controlo velocidade
volatile long encoderPos_Motor_Cabo = 0; //Variável para armazenar a posição
do encoder
bool flag_Motor_Cabo = false;
bool ciclo = true;
float comprimento_atual_cabo = 0; //Variável comprimento atual do cabo
float entrada_cabo_desenrolar = 0;
float entrada_cabo_enrolar = 0;
//Variáveis para o controlador com trajetória
float t0, t1, t2, t3, t4, t5, t6, t7, t8, t9, tf;
float XPOS[8] = { 0.4, 0.6, 0.8, 1.4, 1.6, 1.8, 2.25 };
float YPOS[4] = { 0.64, 0.8, 1 };
float VC = 0.1;
float R = 0.2;
float w = VC / R;
float x_trajetoria, y_trajetoria;
float tempoInicial;
float tempoDecorrido;
//Variáveis para controlador com observador de ordem reduzida
float xh = 0;
float xh_previous = 0;
float valor_previous = 0;
float Posicao_previous = 0;
float entrada_previous = 0;
//Matriz Resultante com ganhos K e L
float resultmatrix[13][8] = {
{0.1750000000000000, -0.540090533439580, -0.159472608441493,
0.0383480889656744, 1.093999999999999, 0.150006257234590, 0.09989999569757940,
0.00984766950136563},
{0.3250000000000000, -1.13740181270211, -0.271457563190547, 0.0710871287968287,
1.093999999999999, 0.147407600533714, 0.379024350175551, 0.0182549526999296},
{0.4750000000000000, -1.70198164218469, -0.360609161413544, 0.103851722325331,
1.093999999999999, 0.140648560679208, 0.482422949199753, 0.0266687980080490},
```

```

{0.6250000000000000, -2.25031826426305, -0.426927392691499, 0.136623459926271,
1.0939999999999999, 0.132989541506739, 0.536289892390337, 0.0350844778917052},

{0.7750000000000000, -2.78613361349554, -0.470412253689134, 0.169398191985672,
1.0939999999999999, 0.125001669010067, 0.569335847468980, 0.0435009267426933},

{0.9250000000000000, -3.31073519830860, -0.491063743081737, 0.202174461386403,
1.0939999999999999, 0.116859252397763, 0.591676682799971, 0.0519177703784757},

{1.0750000000000000, -3.82470070276134, -0.488881860240779, 0.234951624473785,
1.0939999999999999, 0.108632437869455, 0.607788872930479, 0.0603348435101760},

{1.2250000000000000, -4.32832484570947, -0.463866604831724, 0.267729352907572,
1.0939999999999999, 0.100354672061588, 0.619958460787399, 0.0687520618209416},

{1.3750000000000000, -4.82177373412607, -0.416017976661831, 0.300507461642586,
1.0939999999999999, 0.0920438574225126, 0.629474765526426, 0.0771693777919367},

{1.5250000000000000, -5.30514811807532, -0.345335975608500, 0.333285838448529,
1.0939999999999999, 0.0837104175969086, 0.637120196865217, 0.0855867626024663},

{1.6750000000000000, -5.77851265633811, -0.251820601602033, 0.366064411296682,
1.0939999999999999, 0.0753608234782532, 0.643397076183660, 0.0940041977567037},

{1.8250000000000000, -6.24191074913300, -0.135471854574713, 0.398843131849254,
1.0939999999999999, 0.0669993001595254, 0.648642672102834, 0.102421670840178},

{1.9750000000000000, -6.69537261123214, 0.00371026549270929, 0.431621966443827,
1.0939999999999999, 0.0586287212942275, 0.653091844767753, 0.110839173210189}

};

//Matriz de Estado A

float Aresultmatrix[13][6] = {

{0.992648593665228,0.00996786744953034,-0.558247093888464,0.997205333070728,
1,1},

{0.997169326546803,0.00998835373510582,-0.301212380798731,0.998493226686841,
1,1},

{0.998348632917887,0.00999346068684482,-0.206198054857488,0.998968725769370,
1,1},

{0.998857888630799,0.00999559533612710,-0.156743995781712,0.999216130767646,
1,1},

{0.999134790752634,0.00999672731426259,-0.126420763449996,0.999367805000126,
1,1},

{0.999306720984870,0.00999741625430181,-0.105927398755422,0.999470301791369,
1,1},

{0.999423080279107,0.00999787494647102,-0.0911510134015091,0.999544201100524,
1,1},

{0.999506728106451,0.00999820021329691,-0.0799922671731841,0.999600005776079,
1,1},

{0.999569598582782,0.00999844186974718,-0.0712675605219753,0.999643636631668,
1,1},

{0.999618494793842,0.00999862795083645,-0.0642588324356440,0.999678685405532,
1,1},

{0.999657563795481,0.00999877535259724,-0.0585051726101438,0.999707457439477,
1,1},

```

```

{0.999689470746236,0.00999889482327447,-0.0536971698806551,0.999731500253468,
1,1},

{0.999716003216434,0.00999899350595370,-0.0496193935205280,0.999751891287983,
1,1},

};

//Matriz de Entrada B

float Bresultmatrix[13][4] = {

{-0.0569592425687448,0.0159695253101277,0.0100000000000000,
0.0100000000000000},

{-0.0307333961080179,0.00463622557895176,0.0100000000000000,
0.0100000000000000},

{-0.0210388646038838,0.00217110364343191,0.0100000000000000,
0.0100000000000000},

{-0.0159929525378034,0.00125419077176732,0.0100000000000000,
0.0100000000000000},

{-0.0128990029861453,0.000815735483708952,0.0100000000000000,
0.0100000000000000},

{-0.0108080175722182,0.000572646712033960,0.0100000000000000,
0.0100000000000000},

{-0.00930034878741490,0.000423998976256322,0.0100000000000000,
0.0100000000000000},

{-0.00816179609248727,0.000326525897078118,0.0100000000000000,
0.0100000000000000},

{-0.00727159408708886,0.000259173358786760,0.0100000000000000,
0.0100000000000000},

{-0.00655647734481079,0.000210698094732838,0.0100000000000000,
0.0100000000000000},

{-0.00596941812095357,0.000174652274939195,0.0100000000000000,
0.0100000000000000},

{-0.00547884647850656,0.000147123148784714,0.0100000000000000,
0.0100000000000000},

{-0.00506278152200187,0.000125624664312697,0.0100000000000000,
0.0100000000000000},

};

float A_1, A_2, A_3, A_4, A_5, B_1, B_2, B_3, K_1, K_2, K_3, K_4, L_1, L_2,
L_3;

int index = 0;

char read_keyb();

char comm;

float ref;

int selection;

```

```
//Definir o setup geral

void setup() {

  Serial.begin(115200); //Começar a ligação à porta série

  pinMode(PinoDirecao, OUTPUT); //Definição do pino de direção como uma saída
do arduino ou seja fornece sinal de 5 ou 0 volts

  pinMode(PinoSalto, OUTPUT); //Definição do pino de salto como uma saída do
arduino ou seja fornece sinal de 5 ou 0 volts

  pinMode(Enable, OUTPUT); //Pino que liga e desliga a habilitação do Driver

  pinMode(FIM_DE_CURSO, INPUT); //Pino que dá a leitura do Fim de curso

  pinMode(CANAL_A, INPUT); //Pino que dá a leitura do Canal A

  pinMode(CANAL_B, INPUT); //Pino que dá a leitura do Canal B

  pinMode(TPROSS, OUTPUT); //Pino para marcar o tempo de processamento

  pinMode(CSn, OUTPUT); //Pino de chipo select para o encoder de ângulo

  pinMode(Clk, OUTPUT); //Pino de clock para o encoder de ângulo

  pinMode(DO, INPUT); //Pino de dados para ler o valor do encoder de ângulo

  //Pinos de leitura do teclado de matriz

  pinMode(lin1, OUTPUT);

  pinMode(lin2, OUTPUT);

  pinMode(lin3, OUTPUT);

  pinMode(lin4, OUTPUT);

  pinMode(col1, INPUT_PULLUP);

  pinMode(col2, INPUT_PULLUP);

  pinMode(col3, INPUT_PULLUP);

  digitalWrite(CSn, HIGH);

  digitalWrite(Clk, HIGH);

  digitalWrite(Enable, HIGH); //Ligar o Enable do controlador assim que o
Arduino Liga

  Timer3.initialize(dt); //Definir o temporizador com o tempo de amostragem
"dt"

  Timer3.attachInterrupt(tempo_amostragem); //interrupt service para a função
tempo de amostragem

  attachInterrupt(digitalPinToInterrupt(CANAL_A), CANAL_CONTAR, CHANGE);

  attachInterrupt(digitalPinToInterrupt(CANAL_B), CANAL_CONTAR, CHANGE);

  pinMode(FIM_DE_CURSO_CABO, INPUT_PULLUP); //Configura botão como entrada com
pull-up

  pinMode(In1_Motor_Cabo, OUTPUT); //Configura pino como saída
```

```

    pinMode(In2_Motor_Cabo, OUTPUT);           //Configura pino como saída
    pinMode(Pwm_Motor_Cabo, OUTPUT);          //Configura pino como saída
    analogWrite(Pwm_Motor_Cabo, 0);           //Velocidade do motor para 0
(parado)
    digitalWrite(In1_Motor_Cabo, HIGH);       //Liga dir IN1
    digitalWrite(In2_Motor_Cabo, HIGH);       //Liga dir IN2
    pinMode(botaoEnrolar, INPUT_PULLUP);      //Configura botão como entrada com
pull-up
    pinMode(botaoDesenrolar, INPUT_PULLUP);   //Configura outro botão como
entrada com pull-up
    pinMode(encoderPinA, INPUT_PULLUP);       //Configura pino como entrada com
pull-up
    pinMode(encoderPinB, INPUT_PULLUP);       //Configura pino como entrada com
pull-up
    attachInterrupt(3, encoderISR, CHANGE);    //Configura a interrupção no pino
20

    home(); //Rotina de referenciamento
    referencia = 2.25; //Degrau de estímulo
    referencia_cabo = 1.0; //Degrau de estímulo

    //Inicialização de variáveis
    entrada = 0;
    entrada_cabo = 0;
    x_futuro_estimado_1 = 0;
    x_futuro_estimado_2 = 0;
    x_futuro_estimado_3 = 0;
    x_1 = 0;
    x_2 = 0;
    x_3 = 0;
    x_4 = 0;
    xh = 0;
    xh_previous = 0;
    valor_previous = 0;
    Posicao_previous = 0;
    entrada_previous = 0;

    Calcular_tis(); //Chama função que calcula valores para o controle com
trajetória
}

```

```
//LOOP GERAL DO SISTEMA

void loop() {

    comm = read_keyb();

    if ((Serial.available() > 0) || (comm != 0)) { //Se ler alguma coisa na
porta permite o funcionamento

        if (comm == 0) comm = Serial.read();

        switch (comm) {

            case 'c':

            case '1':

                configurar_Referencias(); //Chamar função para configuração de
setpoints de posição e comprimento do cabo

                break;

            case 'a':

            case '2':

                //Condições iniciais do movimento controlado

                movtype = MOV_CTRL;

                entrada = 0;

                entrada_cabo = 0;

                x_1 = 0;

                x_2 = 0;

                x_3 = 0;

                x_4 = 0;

                break;

            case 'o':

            case '3':

                //Condições iniciais do movimento controlado com obstáculo

                movtype = MOV_CTRL_OBS;

                entrada = 0;

                entrada_cabo = 0;

                x_1 = 0;

                x_2 = 0;

                x_3 = 0;

                x_4 = 0;

                break;

            case 't':

            case '4':

                //Condições iniciais do movimento controlado com trajetória

                movtype = MOV_CTRL_TRAJ;
```

```
    entrada = 0;
    entrada_cabo = 0;
    x_1 = 0;
    x_2 = 0;
    x_3 = 0;
    x_4 = 0;
    tempoInicial = 0;
    tempoDecorrido = 0;
    break;
case 'r':
case '5':
    //Condições iniciais do controlo com reduzido
    movtype = MOV_CTRL_REDUZ;
    entrada = 0;
    entrada_cabo = 0;
    x_1 = 0;
    x_2 = 0;
    x_3 = 0;
    x_4 = 0;
    xh = 0;
    xh_previous = 0;
    valor_previous = 0;
    Posicao_previous = 0;
    entrada_previous = 0;
    break;
case 'l':
case '9':
    //Condições iniciais do movimento
    movtype = MOV_CTRL_L;
    entrada = 0;
    entrada_cabo = 0;
    x_1 = 0;
    x_2 = 0;
    x_3 = 0;
    x_4 = 0;
    break;
case 'h':
```

```
case 'A':
    //Condições iniciais do movimento de referenciamento
    analogWrite(Pwm_Motor_Cabo, 0); //Velocidade do motor para 0 (parado)
    digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1
    digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2
    home(); //Chama função que realiza o referenciamento
    movtype = MOV_STOP;
    entrada = 0;
    entrada_cabo = 0;
    break;
case 'b':
case '6':
    //Modo de controlo do cabo com recurso a dois botões
    while (1) {
        comm = read_keyb();
        if ((comm == 0) && Serial.available()) comm = Serial.read();
        if ((comm == 'e') || (comm == '7')) { //Sai do modo de botões
            break;
        }
        Serial.print(millis());
        Serial.print(' ');
        Serial.println(encoderPos_Motor_Cabo);
        if (digitalRead(botaoDesenrolar) == LOW) //Se o botão de desenrolar
            estiver pressionado
            {
                digitalWrite(In1_Motor_Cabo, LOW); //Desliga dir IN1
                digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2
                analogWrite(Pwm_Motor_Cabo, 255); //Liga motor a velocidade de
desenrolar
            } else if (digitalRead(botaoEnrolar) == LOW) //Se o botão de enrolar
            estiver pressionado
            {
                digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1
                digitalWrite(In2_Motor_Cabo, LOW); //Desliga dir IN2
                analogWrite(Pwm_Motor_Cabo, 255); //Liga motor a velocidade de
enrolar
            } else {
                digitalWrite(Pwm_Motor_Cabo, 0); //Desliga motor
```

```
        digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1
        digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2
    }
    delay(10);
}
break;
case 'n':
case '8':
    //Condições iniciais do movimento controlado
    movtype = MOV_NOCTRL;
    entrada = 0;
    entrada_cabo = 0;
    flag_Motor_Cabo = false;
    break;
case '\n':
case '\r':
    break;
case '7':
    movtype = MOV_UPDOWN;
    tempoInicial = 0;
    tempoDecorrido = 0;
    break;
case 'B':
default:
    //Ordem de parar o movimento
    movtype = MOV_STOP;
}
x = 1;
while (Serial.available()) Serial.read();
}
if ((flag == 1) && (x == 1)) {
    if (contador <= 266655) {
        digitalWrite(TPROSS, HIGH);

        Posicao = (contador)*0.00005625;
//Converte a contagem de pulsos para distância em metro(1 incremento =
0.00005625m)

        comprimento_atual_cabo = 0.44 + (encoderPos_Motor_Cabo * (-
0.0004347826)); //Converte incremento de encoder para distância em metro
```

```

    val = read_ssi_enc();
//Lê o encoder de ângulo

    valor = ((val - 700) * (2. * 3.1416) / 4096.0);
//Conversão do ângulo em rad

    graus = valor * (180 / 3.1416);

switch (movtype) {
    case MOV_CTRL:
        {
            saida_real = (valor + Posicao); //Cálculo da saída modificada
(para poder usar o observador de Luenberger)

            saida_estimada = (x_1 + x_3);

            saida_total = saida_real - saida_estimada; //Erro de estimação de
saída

            //Aplicação do observador

            x_futuro_estimado_1 = 0.9994 * x_1 + 0.009998 * x_2 - 0.009998 *
entrada + 0.0236 * (saida_total);

            x_futuro_estimado_2 = -0.09799 * x_1 + 0.9995 * x_2 + 0.00049 *
entrada + 0.6477 * (saida_total);

            x_futuro_estimado_3 = x_3 + 0.01 * entrada + 0.1276 *
(saida_total);

            x_1 = x_futuro_estimado_1;
            x_2 = x_futuro_estimado_2;
            x_3 = x_futuro_estimado_3;
            x_4 = comprimento_atual_cabo;

            //Aplicação do controlador de realimentação de estado com duas
entradas

            entrada = -1 * (-1.9908 * (x_1 - 0) - 0.8527 * (x_2 - 0) + 0.0339
* (x_3 - referencia));

            entrada_cabo = 9.5163 * (x_4 - referencia_cabo);

        }

        break;

    case MOV_CTRL_OBS:
        {
            saida_real = (valor + Posicao); //Cálculo da saída modificada
(para poder usar o observador de Luenberger)

            saida_estimada = (x_1 + x_3);

            saida_total = saida_real - saida_estimada; //Erro de estimação de
saída

```

```

//Aplicação do observador

x_futuro_estimado_1 = 0.9994 * x_1 + 0.009998 * x_2 - 0.009998 *
entrada + 0.0171 * (saida_total);

x_futuro_estimado_2 = -0.09799 * x_1 + 0.9995 * x_2 + 0.00049 *
entrada + 0.6332 * (saida_total);

x_futuro_estimado_3 = x_3 + 0.01 * entrada + 0.1310 *
(saida_total);

x_1 = x_futuro_estimado_1;
x_2 = x_futuro_estimado_2;
x_3 = x_futuro_estimado_3;
x_4 = comprimento_atual_cabo;

//Para simular "obstáculo"
if ((x_3 <= 0.9)) {
    referencia_cabo = 0.8;
}
else if ((x_3 > 0.9) && (x_3 <= 1.7)) {
    referencia_cabo = 0.45;
} else if ((x_3 > 1.7)) {
    referencia_cabo = 0.8;
} else if ((x_3 >= 2.25)) {
    entrada_cabo = 0;
}

//Aplicação do controlador de realimentação de estado com duas
entradas
entrada = -1 * (-3.5709 * (x_1 - 0) - 0.4932 * (x_2 - 0) + 0.2185
* (x_3 - referencia));
entrada_cabo = 1.0940 * (x_4 - referencia_cabo);
}
break;

case MOV_CTRL_TRAJ:
{
    saida_real = (valor + Posicao); //Cálculo da saida modificada
(para poder usar o observador de Luenberger)
    saida_estimada = (x_1 + x_3);
    saida_total = saida_real - saida_estimada; //Erro de estimação de
saída

```

```

//Aplicação do observador

x_futuro_estimado_1 = 0.9994 * x_1 + 0.009998 * x_2 - 0.009998 *
entrada + 0.0236 * (saida_total);

x_futuro_estimado_2 = -0.09799 * x_1 + 0.9995 * x_2 + 0.00049 *
entrada + 0.6477 * (saida_total);

x_futuro_estimado_3 = x_3 + 0.01 * entrada + 0.1276 *
(saida_total);

x_1 = x_futuro_estimado_1;
x_2 = x_futuro_estimado_2;
x_3 = x_futuro_estimado_3;
x_4 = comprimento_atual_cabo;

if (tempoInicial == 0) {
    tempoInicial = millis();//Regista o tempo inicial apenas uma vez
}

tempoDecorrido = (millis() - tempoInicial) / 1000;
if (tempoDecorrido <= 100.0) {
    trajetoria(tempoDecorrido);
}

referencia = x_trajetoria;
referencia_cabo = y_trajetoria;

//Aplicação do controlador de realimentação de estado com duas
entradas

entrada = -1 * (-2.2884 * (x_1 - 0) - 0.8438 * (x_2 - 0) + 0.0303
* (x_3 - referencia));

entrada_cabo = 1.0940 * (x_4 - referencia_cabo);
}

break;

case MOV_CTRL_REDUZ:
{
    float g_1 = 2;
    float g_2 = 1;

    xh = ((0.9995 - g_1 * 0.009998) * xh_previous) + ((0.00049 - ((g_1
* (-0.009998)) + (g_2 * 0.01))) * entrada_previous) + g_1 * valor + g_2 *
Posicao + ((-0.097990 - (0.9994 * g_1)) * valor_previous) + ((-1 * g_2) *
Posicao_previous);

    x_1 = valor;
    x_2 = xh;
}

```

```

        x_3 = Posicao;
        x_4 = comprimento_atual_cabo;

        //Aplicação do controlador de realimentação de estado com duas
entradas
        entrada = -1 * (-1.7321 * (x_1 - 0) - 0.1932 * (x_2 - 0) + 0.2185
* (x_3 - 2.25));
        entrada_cabo = 1.0940 * (x_4 - referencia_cabo);

        xh_previous = xh;
        valor_previous = x_1;    //valor;
        Posicao_previous = x_3;  //Posicao;
        entrada_previous = entrada;
    }
    break;

case MOV_CTRL_L:
    {
        referencia_cabo = 1;
        referencia = 2.25;
        variarL();

        saida_real = (valor + Posicao); //cálculo da saida modificada
(para poder usar o observador de Luenberger)
        saida_estimada = (x_1 + x_3);
        saida_total = saida_real - saida_estimada; //erro de estimação de
saída

        //Aplicação do observador
        x_futuro_estimado_1 = A_1 * x_1 + A_2 * x_2 + B_1 * entrada + L_3
* (saida_total);
        x_futuro_estimado_2 = A_3 * x_1 + A_4 * x_2 + B_2 * entrada + L_2
* (saida_total);
        x_futuro_estimado_3 = A_5 * x_3 + B_3 * entrada + L_1 *
(saida_total);
        x_1 = x_futuro_estimado_1;
        x_2 = x_futuro_estimado_2;
        x_3 = x_futuro_estimado_3;
        x_4 = comprimento_atual_cabo;
    }
}

```

```

//Aplicação do controlador de realimentação de estado com duas
entradas
    entrada = -1 * (K_1 * (x_1 - 0) - K_2 * (x_2 - 0) + K_3 * (x_3 -
referencia));
    entrada_cabo = K_4 * (x_4 - referencia_cabo);
}
break;

case MOV_NOCTRL:
{
    float erro = Posicao - referencia;
    if (erro < 0) {
        entrada = 0.4;
    } else {
        entrada = 0;
    }
    float erroCabo = referencia_cabo - comprimento_atual_cabo;
    if ((erroCabo > 0) && (flag_Motor_Cabo == false)) {
        entrada_cabo = -0.26;
    } else if ((erroCabo <= 0) && (flag_Motor_Cabo == false)) {
        entrada_cabo = 0;
    }
}
break;

case MOV_UPDOWN:
{
    if (tempoInicial == 0) {
        tempoInicial = millis(); //Regista o tempo inicial apenas uma
vez
    }
    tempoDecorrido = (millis() - tempoInicial) / 1000;
    float ref = 0.60 + 0.15 * sin(tempoDecorrido);
    float erro = ref - comprimento_atual_cabo;
    entrada_cabo = -erro;
    entrada = 0;
}
break;

```

```
case MOV_STOP:
{
    entrada = 0;
    entrada_cabo = 0;
}
break;
}

//Conversão da velocidade para controlo do cabo
if (entrada_cabo < -0.01) {
    if (entrada_cabo > -0.215) {
        entrada_cabo_desenrolar = -(-0.215);
    } else if (entrada_cabo < -0.23771) {
        entrada_cabo_desenrolar = -(-0.23771);
    } else if ((entrada_cabo >= -0.23771) && (entrada_cabo <= -0.215)) {
        entrada_cabo_desenrolar = -(entrada_cabo);
    }

    vel_cabo = 214550 * pow(entrada_cabo_desenrolar, 4) - 69016 *
pow(entrada_cabo_desenrolar, 3) + 8535.7 * pow(entrada_cabo_desenrolar, 2) -
189.88 * entrada_cabo_desenrolar + 59.793;

    digitalWrite(In1_Motor_Cabo, LOW); //Desliga dir IN1
    digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2
    analogWrite(Pwm_Motor_Cabo, vel_cabo); //Liga motor a velocidade de
desenrolar

} else if (entrada_cabo > 0.01) {
    if (entrada_cabo > 0.1622) {
        entrada_cabo_enrolar = 0.1622;
    } else if (entrada_cabo < 0.082725456) {
        entrada_cabo_enrolar = 0.082725456;
    } else if ((entrada_cabo >= 0.082725456) && (entrada_cabo <= 0.1622))
{
        entrada_cabo_enrolar = entrada_cabo;
    }

    vel_cabo = 438.21 * pow(entrada_cabo_enrolar, 2) + 490.73 *
entrada_cabo_enrolar + 163.81;

    digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1
    digitalWrite(In2_Motor_Cabo, LOW); //Desliga dir IN2
```

```
        analogWrite(Pwm_Motor_Cabo, vel_cabo); //Liga motor a velocidade de
enrolar
    } else if ((entrada_cabo <= 0.01) && (entrada_cabo >= -0.01)) {
        analogWrite(Pwm_Motor_Cabo, 0); //Motor a velocidade 0
        digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1
        digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2
    }

//Aplicação de saturação aos valores de controle

if (entrada > 0.4)
    entrada = 0.4;
else if (entrada < -0.4)
    entrada = -0.4;
else
    entrada = entrada;

//Segurança
if ((entrada < 0) && (digitalRead(FIM_DE_CURSO))) entrada = 0;

//Cálculo da frequência dividindo a velocidade por cada passo de encoder
freq = entrada / .00005625;

//Restrição de velocidade
if (freq >= 7000) { //limitacao de velocidade do motor para 1600 steps
    freq = 7000;
    digitalWrite(PinoDirecao, LOW);
    tone(PinoSalto, freq);
}

//Limitação de amplitude máxima
else if (freq <= -7000) { //Limitação de velocidade do motor para 1600
steps
    freq = 7000;
    digitalWrite(PinoDirecao, HIGH);
    tone(PinoSalto, freq);
} else if (freq >= 31) { //Dependendo da frequência de saída mudar a
direção
    digitalWrite(PinoDirecao, LOW);
```

```

    tone(PinoSalto, freq);
} else if (freq <= -31) {
    freq = -freq; //Dependendo da frequência de saída mudar a direção
    digitalWrite(PinoDirecao, HIGH);
    tone(PinoSalto, freq);
}
//Banda morta de velocidade
else if (freq < 31) {
    noTone(PinoSalto);
} else if (freq > -31) {
    noTone(PinoSalto);
}
} else {
    noTone(PinoSalto);
}

Serial.print(millis());

Serial.print(' ');

Serial.println(graus + spc + val + spc + Posicao + spc + movtype + spc +
entrada + spc + comprimento_atual_cabo + spc + entrada_cabo + spc + vel_cabo);

    flag = 0; //Repõe a flag a 0 para esperar pelo próximo instante de
amostragem

    digitalWrite(TPROSS, LOW);
}
}

//Função encoder. Esta função gere a posição do encoder quando o motor roda
para um lado o valor da posição incrementa quando roda para o outro o valor da
posição diminui
void encoderISR() {
    static int lastStateA = HIGH;
    static int lastStateB = HIGH;
    int stateA = digitalRead(encoderPinA);
    int stateB = digitalRead(encoderPinB);
    if (stateA != lastStateA || stateB != lastStateB) {
        if ((lastStateA == HIGH && lastStateB == LOW) || (lastStateA == LOW &&
lastStateB == HIGH)) {
            encoderPos_Motor_Cabo++;

```

```

    } else {
        encoderPos_Motor_Cabo--;
    }
}

lastStateA = stateA;
lastStateB = stateB;
}

//Rotina de serviço ao interrupt, habilita os cálculos do loop principal.
void tempo_amostragem() {
    flag = 1;
}

//Rotina de atualização da posição a partir dos sinais de quadratura do
encoder
void CANAL_CONTAR() {
    A = digitalRead(CANAL_A);
    B = digitalRead(CANAL_B);

    if ((Last_A == 0 && A == 0 && Last_B == 0 && B == 1) || (Last_A == 0 && A ==
1 && Last_B == 1 && B == 1) || (Last_A == 1 && A == 1 && Last_B == 1 && B ==
0) || (Last_A == 1 && A == 0 && Last_B == 0 && B == 0)) {

        contador--;

    } else if ((Last_A == 0 && A == 0 && Last_B == 1 && B == 0) || (Last_A == 0
&& A == 1 && Last_B == 0 && B == 0) || (Last_A == 1 && A == 1 && Last_B == 0
&& B == 1) || (Last_A == 1 && A == 0 && Last_B == 1 && B == 1)) {

        contador++;

    }

    Last_A = A;
    Last_B = B;
}

//Rotina de referenciação
void home() {
    //Homing do sistema, ou seja, definir o ponto em que o carro começa a 0
    //aproximação rápida

    noTone(PinoSalto); //Se o motor estiver a fazer algo
logo no homing faz notone por segurança

    digitalWrite(PinoDirecao, HIGH); //Direção no sentido do fim de curso

    tone(PinoSalto, 5000); //A função tone mete a frequência
que se quiser ou seja colocar aqui a frequência(velocidade) de aproximação

```

```

while (!digitalRead(FIM_DE_CURSO)) {}; //Enquanto não ler o FIM de curso a
1 ele percorre todo o caminho até ao fim de curso

noTone(PinoSalto); //Faz parar o motor

//Ajuste lento no sentido contrário

delay(1000);

tone(PinoSalto, 2000);

digitalWrite(PinoDirecao, LOW);

while (digitalRead(FIM_DE_CURSO)) {} //Quando o fim de curso estiver a 1
ele anda até ficar a 0

noTone(PinoSalto);

//Guarda posição atual da linha de encoder

Last_A = digitalRead(CANAL_A);

Last_B = digitalRead(CANAL_B);

contador = 0; //Inicializa o contador de posição no ponto inicial

while (digitalRead(FIM_DE_CURSO_CABO)) {

    digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1

    digitalWrite(In2_Motor_Cabo, LOW); //Desliga dir IN2

    analogWrite(Pwm_Motor_Cabo, 255); //Liga motor a velocidade de enrolar

}; //Enquanto não ler o FIM de curso a 1 ele percorre todo o caminho até ao
fim de curso

analogWrite(Pwm_Motor_Cabo, 0); //Desliga o motor

digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1

digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2

delay(200);

while (!digitalRead(FIM_DE_CURSO_CABO)) {

    digitalWrite(In1_Motor_Cabo, LOW); //Desliga dir IN1

    digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2

    analogWrite(Pwm_Motor_Cabo, 80); //Liga motor a velocidade de
deserolar

}; //Quando o fim de curso estiver a 1 ele anda até ficar a 0

digitalWrite(Pwm_Motor_Cabo, 0); //Desliga motor

digitalWrite(In1_Motor_Cabo, HIGH); //Liga dir IN1

digitalWrite(In2_Motor_Cabo, HIGH); //Liga dir IN2

```

```
delay(1750);

encoderPos_Motor_Cabo = 0; //Inicializa o contador de posição no ponto
inicial
}

//rotina de leitura do encoder absoluto do ângulo
int read_ssi_enc() {
    int result = 0;
    digitalWrite(CSn, LOW);
    delayMicroseconds(1);
    digitalWrite(Clk, LOW);
    for (int i = 1; i <= 12; i++) {
        digitalWrite(Clk, HIGH);
        delayMicroseconds(1);
        result = (result << 1) + !digitalRead(DO);
        digitalWrite(Clk, LOW);
        delayMicroseconds(1);
    }
    digitalWrite(Clk, HIGH);
    digitalWrite(CSn, HIGH);
    return result;
}

//Cálculo dos troços da trajetória
void Calcular_tis() {
    t1 = 0;
    t2 = XPOS[0] / VC;
    t3 = t2 + PI / (2 * w);
    t4 = t3 + (YPOS[1] - YPOS[0]) / VC;
    t5 = t4 + PI / (2 * w);
    t6 = t5 + (XPOS[3] - XPOS[2]) / VC;
    t7 = t6 + PI / (2 * w);
    t8 = t7 + (YPOS[1] - YPOS[0]) / VC;
    t9 = t8 + PI / (2 * w);
    tf = t9 + (XPOS[6] - XPOS[5]) / VC;
}
```

```
void trajetoria(float t) {
    if (t < t2) //Troço 1
    {
        x_trajetoria = VC * t;
        y_trajetoria = 0.44;
    } else if (t < t3) //Troço 2
    {
        x_trajetoria = XPOS[0] + R * cos(-PI / 2 + w * (t - t2));
        y_trajetoria = 0.44 + R + R * sin(-PI / 2 + w * (t - t2));
    } else if (t < t4) //Troço 3
    {
        x_trajetoria = XPOS[1];
        y_trajetoria = YPOS[0] + VC * (t - t3);
    } else if (t < t5) //Troço 4
    {
        x_trajetoria = XPOS[1] + R + R * cos(PI - w * (t - t4));
        y_trajetoria = YPOS[1] + R * sin(PI - w * (t - t4));
    } else if (t < t6) //Troço 5
    {
        x_trajetoria = XPOS[2] + VC * (t - t5);
        y_trajetoria = YPOS[2];
    } else if (t < t7) //Troço 6
    {
        x_trajetoria = XPOS[3] + R * cos(PI / 2 - w * (t - t6));
        y_trajetoria = YPOS[2] - R + R * sin(PI / 2 - w * (t - t6));
    } else if (t < t8) //Troço 7
    {
        x_trajetoria = XPOS[4];
        y_trajetoria = YPOS[1] - VC * (t - t7);
    } else if (t < t9) //Troço 8
    {
        x_trajetoria = XPOS[4] + R + R * cos(PI + w * (t - t8));
        y_trajetoria = YPOS[0] + R * sin(PI + w * (t - t8));
    } else if (t < tf) //Troço 9
    {
        x_trajetoria = XPOS[5] + VC * (t - t9);
        y_trajetoria = 0.44;
    }
}
```

```
} else {
    x_trajetoria = XPOS[6];
    y_trajetoria = 0.44;
}
}

void configurar_Referencias() {
    Serial.flush();
    //Faz print no monitor de informação
    Serial.println("\n\n\n-----");
    Serial.println("( Numero Negativo ) - Sair\n");
    Serial.println("      1      ) - Configurar Referencia do Carro");
    Serial.println("      2      ) - Configurar Referencia do Cabo");
    while (1) {
        Serial.flush();
        selection = Serial.parseInt();
        if (selection < 0) {
            //Sair do modo
            break;
        }
        if (selection > 2) {
            Serial.println("\nO valor inserido não é valido!");
            Serial.println("O valor deve inserido deve ser 1 ou 2");
        }
        if (selection == 1) {
            while (1) {
                //Faz print no monitor de informação
                Serial.println("-----");
                Serial.println("(Número Negativo) - Sair\n");
                Serial.println("Inserir Nova Referencia do Carro (m)");
                while (1) {
                    Serial.flush();
                    ref = Serial.parseFloat();
                    if (ref < 0) {
                        break;
                    }
                    if ((ref >= 0.01) && (ref <= 4)) {
```

```

        referencia = ref;

        Serial.println("Nova Referencia do Carro Inserida:" + spc +
referencia + " (m)");

        Serial.println("-----
\n\n\n");

        break;
    }

    if ((ref != 0) && ((ref < 0.01) || (ref > 4))) {
        Serial.println("\nO valor inserido não é valido!");
        Serial.println("O valor deve encontrar-se entre 0.01 (m) e 4(m)");
    }
}

break;
}

break;
}

if (selection == 2) {
    while (1) {
        Serial.println("-----");
        Serial.println("(Numero Negativo) - Sair\n");
        Serial.println("Inserir Nova Referencia do Cabo (m)");
        while (1) {
            Serial.flush();
            ref = Serial.parseFloat();
            if (ref < 0) {
                break;
            }
            if ((ref >= 0.01) && (ref <= 1)) {
                referencia_cabo = ref;

                Serial.println("Nova Referencia do Cabo Inserida:" + spc +
referencia_cabo + " (m)");

                Serial.println("-----
\n\n\n");

                break;
            }
        }

        if ((ref != 0) && ((ref < 0.01) || (ref > 1))) {
            Serial.println("\nO valor inserido não é valido!");
            Serial.println("O valor deve encontrar-se entre 0.01 (m) e 1(m)");
        }
    }
}

```

```
        }
    }
    break;
}
break;
}
}
}

char read_keyb(void) {
    char ret = 0;
    //set first line
    digitalWrite(lin1, LOW);
    digitalWrite(lin2, HIGH);
    digitalWrite(lin3, HIGH);
    digitalWrite(lin4, HIGH);
    //scan first line
    if (digitalRead(col1) == LOW) ret = '1';
    if (digitalRead(col2) == LOW) ret = '2';
    if (digitalRead(col3) == LOW) ret = '3';
    //set second line
    digitalWrite(lin1, HIGH);
    digitalWrite(lin2, LOW);
    //scan second line
    if (digitalRead(col1) == LOW) ret = '4';
    if (digitalRead(col2) == LOW) ret = '5';
    if (digitalRead(col3) == LOW) ret = '6';
    //set third line
    digitalWrite(lin2, HIGH);
    digitalWrite(lin3, LOW);
    //scan third line
    if (digitalRead(col1) == LOW) ret = '7';
    if (digitalRead(col2) == LOW) ret = '8';
    if (digitalRead(col3) == LOW) ret = '9';
    //set fourth line
    digitalWrite(lin3, HIGH);
    digitalWrite(lin4, LOW);
```

```
//scan fourth line
if (digitalRead(col1) == LOW) ret = 'A';
if (digitalRead(col2) == LOW) ret = '0';
if (digitalRead(col3) == LOW) ret = 'B';
//return null if nothing is pressed
if (ret != 0) {
    delay(20);
}
return ret;
}

//Cálculo da classe do comprimento do cabo para o modelo linear por troços
void variarL() {
    if ((comprimento_atual_cabo <= 0.175)) {
        index = 0;
    } else if ((comprimento_atual_cabo > 0.175) && (comprimento_atual_cabo <=
0.325)) {
        index = 1;
    } else if ((comprimento_atual_cabo > 0.325) && (comprimento_atual_cabo <=
0.475)) {
        index = 2;
    } else if ((comprimento_atual_cabo > 0.475) && (comprimento_atual_cabo <=
0.625)) {
        index = 3;
    } else if ((comprimento_atual_cabo > 0.625) && (comprimento_atual_cabo <=
0.775)) {
        index = 4;
    } else if ((comprimento_atual_cabo > 0.775) && (comprimento_atual_cabo <=
0.925)) {
        index = 5;
    } else if ((comprimento_atual_cabo > 0.925) && (comprimento_atual_cabo <=
1.075)) {
        index = 6;
    } else if ((comprimento_atual_cabo > 1.075) && (comprimento_atual_cabo <=
1.225)) {
        index = 7;
    } else if ((comprimento_atual_cabo > 1.225) && (comprimento_atual_cabo <=
1.375)) {
        index = 8;
    }
}
```

```
} else if ((comprimento_atual_cabo > 1.375) && (comprimento_atual_cabo <=
1.525)) {
    index = 9;
} else if ((comprimento_atual_cabo > 1.525) && (comprimento_atual_cabo <=
1.675)) {
    index = 10;
} else if ((comprimento_atual_cabo > 1.675) && (comprimento_atual_cabo <=
1.825)) {
    index = 11;
} else if ((comprimento_atual_cabo > 1.825) && (comprimento_atual_cabo <=
1.975)) {
    index = 12;
}

A_1 = Aresultmatrix[index][0];
A_2 = Aresultmatrix[index][1];
A_3 = Aresultmatrix[index][2];
A_4 = Aresultmatrix[index][3];
A_5 = Aresultmatrix[index][4];
B_1 = Bresultmatrix[index][0];
B_2 = Bresultmatrix[index][1];
B_3 = Bresultmatrix[index][2];
K_1 = resultmatrix[index][1];
K_2 = resultmatrix[index][2];
K_3 = resultmatrix[index][3];
K_4 = resultmatrix[index][4];
L_1 = resultmatrix[index][5];
L_2 = resultmatrix[index][6];
L_3 = resultmatrix[index][7];
}
```