

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

Tony Jun Tanaka

Modelagem de manipuladores robóticos paralelos com elos flexíveis baseada em
redes neurais

São Carlos

2025

Tony Jun Tanaka

Modelagem de manipuladores robóticos paralelos com elos flexíveis baseada em
redes neurais

Monografia apresentada ao Curso de Engenharia Mecatrônica, da Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Engenheiro Mecatrônico.

Orientadora: Profa. Dra. Máira Martins da Silva

São Carlos

2025

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

T161m	<p>Tanaka, Tony Jun</p> <p>Modelagem de manipuladores robóticos paralelos com elos flexíveis baseada em redes neurais / Tony Jun Tanaka; orientadora Máira Martins da Silva. São Carlos, 2025.</p> <p>Monografia (Graduação em Engenharia Mecatrônica) -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2025.</p> <p>1. Manipulador paralelo. 2. Elos flexíveis. 3. Physics-Informed Machine Learning. 4. PINNs. 5. Modelagem dinâmica. I. Título.</p>
-------	---

FOLHA DE AVALIAÇÃO

Candidato: TONY JUN TANAKA

Título:

Modelagem de manipuladores robóticos paralelos
com elos flexíveis baseada em redes neurais

Trabalho de Conclusão de Curso apresentado à
Escola de Engenharia de São Carlos da
Universidade de São Paulo
Curso de Engenharia Mecatrônica

BANCA EXAMINADORA

Professor Maria Martins de Silva
(Orientador)

Nota atribuída: 10 (dez)

Maria M. de Silva
(assinatura)

Professor Kayce Wayhs Lopes

Nota atribuída: 10 (Dez)

Kayce Wayhs Lopes
(assinatura)

Dr. Renan Sanchez Geronzi

Nota atribuída: 10 (Dez)

Renan Sanchez Geronzi
(assinatura)

Média: 10,0 (dez)

Resultado: aprovado

Data: 28 / 11 / 2025.

Este trabalho tem condições de ser hospedado no Portal Digital da Biblioteca da EESC

SIM ☒ NÃO ☐ Visto do orientador Maria M. de Silva

*Este trabalho é dedicado às crianças adultas que,
quando pequenas, sonharam em se tornar cientistas.*

AGRADECIMENTOS

À minha orientadora, Prof^a. Dr^a. Máira Martins da Silva, pela orientação durante todo o desenvolvimento deste trabalho.

Ao Dr. Renan Sanches, pela ajuda e dedicação constante, que contribuiu de forma fundamental para a evolução deste projeto.

Ao Prof. Dr. Rafael Teloli, pela sua disponibilidade e valiosas contribuições de conhecimento e experiência ao longo da pesquisa.

À Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) pelo apoio financeiro, processo nº 2024/07967-9.

Aos meus colegas e amigos pelos momentos de leveza e companheirismo desde o início da faculdade.

Aos meus pais, Ailton e Julia, pelos seus conselhos, incentivos e amparo durante toda a minha vida.

Aos meus tios, Alice e Inácio, que sempre me suportaram incondicionalmente.

À minha namorada Letícia, pelo apoio e paciência que sempre me oferece.

*“Líderes são mestres em encontrar o sentido e
aprendizado em todo tipo de experiência de
vida.”*

(Jucá, 2014, p. 94)

RESUMO

TANAKA, T. J. **Modelagem de manipuladores robóticos paralelos com elos flexíveis baseada em redes neurais**. 2025. Monografia (Trabalho de Conclusão de Curso) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2025.

O aprendizado de máquina é uma área que vem ganhando muito destaque em tempos recentes, sendo que representa uma inovação relevante no campo da modelagem de sistemas dinâmicos ao ser capaz de utilizar dados empíricos para gerar modelos complexos. Este trabalho aplica essa abordagem à modelagem de um manipulador paralelo planar do tipo 3RRR com elos flexíveis, integrando simulações numéricas, experimentos laboratoriais e técnicas avançadas de redes neurais. Inicialmente, utilizou-se o software MSC Adams para a geração de dados virtuais, considerando modelos rígidos e flexíveis, a fim de estabelecer uma base comparativa. Em seguida, foram realizadas medições experimentais com extensômetros e uma câmera, tanto em um elo único quanto no manipulador completo, assegurando a representatividade do ambiente real. A etapa de processamento e modelagem envolveu o emprego de *Physics-Informed Neural Networks* (PINNs), complementadas pela técnica SINDy e por redes *feedforward*, permitindo explorar diferentes estratégias de aprendizado. A validação evidenciou a capacidade das PINNs de reproduzir com elevada precisão os comportamentos observados, mesmo em condições experimentais ruidosas, superando abordagens exclusivamente baseadas em dados. Destaca-se a contribuição deste estudo para a consolidação do *Physics-Informed Machine Learning* (PIML) como ferramenta robusta para sistemas robóticos, ao mesmo tempo em que se reconhece a necessidade de avanços futuros na redução do custo computacional e na integração com estratégias de controle em tempo real.

Palavras-chave: Manipulador paralelo, Elos flexíveis, *Physics-Informed Machine Learning*, PINNs, Modelagem dinâmica.

ABSTRACT

TANAKA, T. J. **Neural Network-Based Modeling of Parallel Robotic Manipulators with Flexible Links**. 2025. Monograph (Undergraduate Thesis) – São Carlos School of Engineering, University of São Paulo, São Carlos, 2025.

Machine learning is an area that has been receiving a lot of emphasis lately, it represents a significant innovation in the modeling of dynamical systems by utilizing empirical data to generate complex models. This work applies the approach to the modeling of a planar 3RRR parallel manipulator with flexible links, integrating numerical simulations, laboratory experiments, and advanced neural network techniques. Initially, the MSC Adams software was employed to generate virtual datasets, considering both rigid and flexible models, thus establishing a comparative basis. Subsequently, experimental measurements were carried out with strain gauges and a camera, both on a single flexible link and on the complete manipulator, ensuring the representativeness of real-world conditions. The processing and modeling stage employed Physics-Informed Neural Networks (PINNs), complemented by the SINDy technique and feedforward neural networks, enabling the exploration of different learning strategies. Validation highlighted the ability of PINNs to reproduce observed behaviors with high accuracy, even under noisy experimental conditions, outperforming purely data-driven approaches. This study contributes to consolidating Physics-Informed Machine Learning (PIML) as a robust tool for robotic systems, while also recognizing the need for future developments in reducing computational cost and integrating the models into real-time control strategies.

Keywords: Parallel manipulator, Flexible links, Physics-Informed Machine Learning, PINNs, Dynamic modeling.

LISTA DE FIGURAS

Figura 1 – Elos cinemáticos do 3RRR com elos flexíveis	30
Figura 2 – Manipulador flexível de elo único.....	31
Figura 3 – Vista completa do pêndulo rígido.....	37
Figura 4 – Vista focada no ponto de fixação do pêndulo rígido.....	38
Figura 5 – Pêndulo flexível durante o movimento oscilatório.....	39
Figura 6 – Instrumentação empregada para o experimento.....	39
Figura 7 – Esquema de comunicação do 3RRR com elos flexíveis.....	42
Figura 8 – Representação do pêndulo físico.....	43
Figura 9 – Características geométricas do manipulador flexível serial.....	44
Figura 10 – Características geométricas do 3RRR.....	46
Figura 11 – Amostra do conjunto de dados utilizados no treinamento e teste da rede neural.....	48
Figura 12 – Simetrização do sinal de deformação em torno e zero.....	49
Figura 13 – Aproximação por série de Fourier com a primeira e segunda derivada.....	50
Figura 14 – Esquemático da Rede Neural do tipo <i>feedforward</i> utilizada.....	52
Figura 15 – Esquemático do PINN utilizado.....	52
Figura 16 – Ilustração esquemática para a aplicação experimental.....	56
Figura 17 – Evolução da solução do PINN do pêndulo rígido.....	58
Figura 18 – Evolução da estimativa do amortecimento do PINN do pêndulo rígido.....	59
Figura 19 – Medidas de deformação e aproximação pela rede neural de Fourier.....	60
Figura 20 – Evolução da função de perda da FNN.....	60
Figura 21 – Evolução dos parâmetros aprendidos pela FNN.....	61
Figura 22 – Evolução da solução do PINN do pêndulo flexível.....	62
Figura 23 – Evolução da estimativa do termo “k” do PINN do pêndulo flexível.....	62
Figura 24 – Evolução das funções de perda do PINN do pêndulo flexível.....	63
Figura 25 – Evolução da solução do PINN do manipulador flexível serial.....	64
Figura 26 – das estimativas dos termos “k” do PINN do manipulador flexível serial.....	65

Figura 27 – Evolução das funções de perda do PINN do manipulador flexível serial.....	65
Figura 28 – Comparação entre as posições reais e o controle do ângulo medido diretamente do manipulador do modelo do PINN.....	66
Figura 29 – Comparação entre as posições reais e o controle do ângulo calculado a partir das posições x e y da câmera do modelo do PINN.....	66
Figura 30 – Comparação entre as posições reais e estimadas pela rede neural.....	67
Figura 31 – Comparação entre as posições da câmera e estimadas pela rede neural do 3RRR.....	68

LISTA DE ABREVIATURAS E SIGLAS

PIML	<i>Physics-Informed Machine Learning</i> (Aprendizado de Máquina Informado por Física)
PINN	<i>Physics-Informed Neural Network</i> (Rede Neural Informada por Física)
PGML	<i>Physics-Informed Neural Network</i> (Rede Neural Informada por Física)
SINDy	<i>Physics-Guided Machine Learning</i> (Aprendizado de Máquina Guiado por Física)
GPU	<i>Graphics Processing Unit</i> (Unidade de Processamento Gráfico)
FNN	<i>Fourier Neural Network</i> (Rede Neural de Fourier)
EESC-USP	Escola de Engenharia de São Carlos – Universidade de São Paulo
MSE	<i>Mean squared error</i> (Erro Quadrático Médio)

SUMÁRIO

1 INTRODUÇÃO.....	24
1.1 Objetivos do trabalho.....	25
2 REVISÃO BIBLIOGRÁFICA.....	28
2.1 Técnicas de controles para manipuladores paralelos.....	28
2.1.1 Manipuladores paralelos com elos flexíveis.....	28
2.1.2 Estratégias de controle alternativas.....	28
2.2 Manipuladores robóticos estudados.....	29
2.2.1 Manipulador 3R̄RR com elos flexíveis.....	29
2.2.2 Manipulador flexível serial.....	30
2.3 Aprendizado de máquina.....	31
2.3.1 Aplicações na engenharia.....	31
2.3.2 Redes neurais artificiais.....	33
2.3.3 Aprendizado de máquina informado por física.....	33
2.4 Softwares utilizados.....	34
2.4.1 MSC Adams.....	34
2.4.2 Google Colab.....	34
2.4.3 MATLAB.....	35
3 METODOLOGIA.....	37
3.1 Modelos virtuais no software MSC Adams.....	37
3.1.1 Pêndulo rígido.....	37
3.1.2 Pêndulo flexível.....	38
3.2 Instrumentação do manipulador paralelo 3R̄RR com elos flexíveis.....	39
3.2.1 Componentes.....	40
3.2.2 Esquema de conexões e funcionamento.....	41
3.3 Modelos físicos e cinemáticos.....	42
3.3.1 Pêndulo Rígido.....	43
3.3.2 Manipulador flexível serial.....	44
3.3.3 Manipulador 3R̄RR.....	46
3.4 Coleta e manipulação de dados.....	46
3.4.1 Coleta e exportação de dados virtuais.....	46
3.4.2 Coleta de dados reais.....	47
3.4.3 Tratamento de dados.....	48
3.5 Modelos de aprendizado de máquina.....	50
3.5.1 Rede neural feedforward.....	51
3.5.2 Rede neural informada por física.....	52
3.5.3 Rede Neural de Fourier.....	55
3.5.4 Validação experimental.....	56
4 RESULTADOS.....	58

4.1 Pêndulos simulados.....	58
4.1.1 Pêndulo rígido.....	58
4.1.2 Pêndulo flexível.....	59
4.2 Manipuladores robóticos.....	63
4.2.1 Manipulador flexível serial.....	63
4.2.2 Manipulador 3RRR.....	67
4.3 Limitações e desafios enfrentados.....	68
4.3.1 Limitações dos experimentos.....	69
4.3.2 Desafios na implementação dos algoritmos de aprendizado de máquina.....	69
4.3.3 Ajustes realizados.....	70
5 CONCLUSÃO.....	72
5.1 Principais conclusões.....	72
5.2 Aspectos positivos.....	72
5.3 Reflexões sobre limitações.....	73
5.4 Perspectiva para melhorias e aplicações futuras.....	73
REFERÊNCIAS.....	76
APÊNDICE.....	80

1 INTRODUÇÃO

O avanço da robótica vem demandando soluções cada vez mais eficientes do ponto de vista energético e dinâmico. Dentro desse contexto, os manipuladores paralelos se tornam uma opção interessante devido à sua eficiência de energia e capacidades dinâmicas altas. Entretanto, o controle desses sistemas é mais complexo devido ao fato deles utilizarem dinâmica acoplada e a falta de uma forma de medir a posição e orientação do efetuador final. Para resolver essas incertezas, técnicas de controle baseado em modelo, visão e processamento de imagem estão sendo usadas em manipuladores paralelos com elos rígidos, algo que já é bem estudado e documentado. Por outro lado, o mesmo não pode ser dito a respeito dos manipuladores com elos flexíveis devido à presença de motores menos potentes e elos mais flexíveis. Uma opção que pode elevar ainda mais a eficiência energética, todavia, isso também aumenta a complexidade do sistema ao introduzir vibrações provenientes da flexibilidade dos componentes.

É possível obter modelos dinâmicos de manipuladores paralelos com elos flexíveis através de diversos métodos, no entanto, a maioria deles fornecem modelos grandes e lentos, sendo inadequados para realizar controle. Visto isso, o aprendizado de máquina informado por física surge como uma alternativa interessante. Este é um novo método que utiliza dados em conjunto com conhecimentos prévios para formar modelos. Essa abordagem, diferentemente de redes neurais profundas, não necessita de um grande banco de dados para funcionar, pois utilizar conhecimentos prévios, como leis físicas e observações, cria restrições teóricas, vieses indutivos e observacionais. Esses fatores aumentam o desempenho do algoritmo de aprendizado, requerendo menos dados para gerar modelos menores e eficientes para controle.

Este novo tipo de abordagem, apesar de ser relativamente nova, já foi utilizado em vários campos de pesquisa e funções relacionadas à automação e robótica. Apesar de todas as pesquisas e avanços, o aprendizado de máquina informado por física possui algumas limitações e continua sob desenvolvimento. Visto isso, este trabalho visa explorar o uso de redes neurais na modelagem de manipuladores complexos, sendo realizado em estágios. Primeiro, sistemas simples construídos no software MSC Adams foram modelados, em seguida foi aplicado a rede neural informada por física em um sistema simplificado do manipulador e por fim, uma rede neural modelou o manipulador 3RRR completo. Esta

estratégia foi adotada para aumentar de forma gradual a complexidade das tarefas realizadas, permitindo uma compreensão progressiva dos métodos utilizados e assegurando que os conceitos fossem consolidados antes da aplicação no manipulador real.

Com este trabalho, espera-se não apenas desenvolver um modelo funcional e eficiente para controle do manipulador 3RRR com elos flexíveis, mas também contribuir para o avanço do uso de técnicas de aprendizado de máquina baseadas em física no contexto da robótica e automação. Os resultados obtidos são promissores, e este trabalho representa um passo importante rumo à integração dessas metodologias em aplicações práticas de engenharia.

1.1 Objetivos do trabalho

O objetivo principal deste trabalho é a aplicação de técnicas de aprendizado de máquina, em especial redes neurais, na modelagem dinâmica de manipuladores com elos flexíveis, avaliando sua viabilidade como ferramenta de suporte ao desenvolvimento de estratégias de controle.

A fim de atingir o objetivo principal, diversos objetivos específicos serão realizados, sendo eles:

- Desenvolver modelos virtuais de pêndulos rígidos e flexíveis no software MSC Adams para geração de dados sintéticos.
- Realizar experimentos com o manipulador serial e com o manipulador 3RRR completo, utilizando extensômetros, câmera e codificadores para coletar dados de treinamento dos algoritmos.
- Implementar uma rede neurais do tipo PINN nos pêndulos simulados e no manipulador flexível, incorporando restrições físicas no processo de aprendizado. E uma do tipo *feedforward* no manipulador 3RRR.
- Analisar a precisão, robustez e potencial de generalização dos modelos obtidos.
- Identificar limitações e propor perspectivas para aplicação das técnicas em controle em tempo real.

Esses objetivos visam proporcionar um entendimento mais aprofundado das técnicas de aprendizado de máquina, avaliando sua viabilidade e investigando sua aplicação em sistemas dinâmicos, com vistas ao aprimoramento da eficiência e da efetividade desses sistemas.

2 REVISÃO BIBLIOGRÁFICA

2.1 Técnicas de controles para manipuladores paralelos

Os manipuladores robóticos se tornaram ferramentas muito importantes nos processos produtivos devido à sua eficiência e adaptabilidade em realizar diversas tarefas. No entanto, a indústria está em uma busca constante por melhorias nos processos produtivos, o que leva a avanços tecnológicos a fim de garantir maior precisão, velocidade, qualidade do produto e redução no tempo de produção. Dentro dos manipuladores robóticos, existem duas categorias principais: manipuladores seriais, que apresentam elos conectados em cadeia simples, e manipuladores paralelos, nos quais o efetuador final é sustentado por múltiplas cadeias cinemáticas. A escolha e controle dos manipuladores paralelos será descrita nesta seção.

2.1.1 Manipuladores paralelos com elos flexíveis

Devido às múltiplas cadeias cinemáticas, que funcionam de forma independente e simultânea, os manipuladores paralelos possuem eficiência de energia e capacidades dinâmicas maiores em relação a manipuladores em série (Ruiz *et al.*, 2018; Carvalho Fontes *et al.*, 2021). Uma forma de elevar ainda mais a eficiência energética e o desempenho dinâmico é diminuindo a inércia dos componentes com elos finos e flexíveis. No entanto, a modelagem deste tipo de sistema é um grande desafio, pois esses dois métodos aumentam a complexidade da dinâmica do sistema.

A dinâmica acoplada dos manipuladores paralelos e a falta de uma forma de medir a posição e orientação do efetuador final diretamente dificulta o seu controle (Paccot; Andreff; Martinet, 2009). Já os elos flexíveis introduzem vibrações e deformações, que também contribuem para uma maior complexidade no controle e modelagem, por isso, esse campo dos manipuladores paralelos com elos flexíveis é pouco explorado (Morlock *et al.*, 2021; Morlock *et al.*, 2022).

2.1.2 Estratégias de controle alternativas

Uma estratégia utilizada para lidar com os problemas citados é o controle baseado em modelo, que apresenta elevado desempenho em manipuladores paralelos com alta não-linearidade, parâmetros com variação no tempo e incertezas (Saied *et al.*, 2018; Bennehar *et al.*, 2017). Além disso, técnicas baseadas em visão e processamento de imagem estão sendo

usadas para resolver essas incertezas (Mohan *et al.*, 2017; Colombo; Carvalho Fontes; Silva, 2019). Também é possível obter modelos dinâmicos de manipuladores paralelos com elos flexíveis através dos métodos dos elementos finitos (Pham *et al.*, 2022; Colombo; Silva, 2022), Lagrangiano e multicorpos flexíveis (Lipinski *et al.*, 2019), modelo de restrição de feixe dinâmico (Ling; Yuan; Zhang, 2024), entre outros. No entanto, esses métodos fornecem modelos grandes e lentos, sendo inadequados para realizar controle (Silva *et al.*, 2010).

Uma alternativa que já foi implementada com sucesso em manipuladores paralelos de elos rígidos e robôs paralelos acionados por cabos foi o controle sem modelo (Wang *et al.*, 2020; Wang *et al.*, 2021). Mas no caso dos elos flexíveis, é necessário compensar a deformação por meio de uma estimativa dela (Bengoa *et al.*, 2017; Bastos, 2022), esse cálculo torna o controle mais complexo e lento. Em razão disso, buscamos o aprendizado de máquina como uma possível alternativa para esse tipo de modelagem.

2.2 Manipuladores robóticos estudados

O sistema principal estudado neste trabalho foi o manipulador 3RRR paralelo com elos flexíveis, também foi utilizada uma versão simplificada do manipulador com apenas um elo. Ambos os sistemas e seus históricos serão expostos nesta seção.

2.2.1 Manipulador 3RRR com elos flexíveis

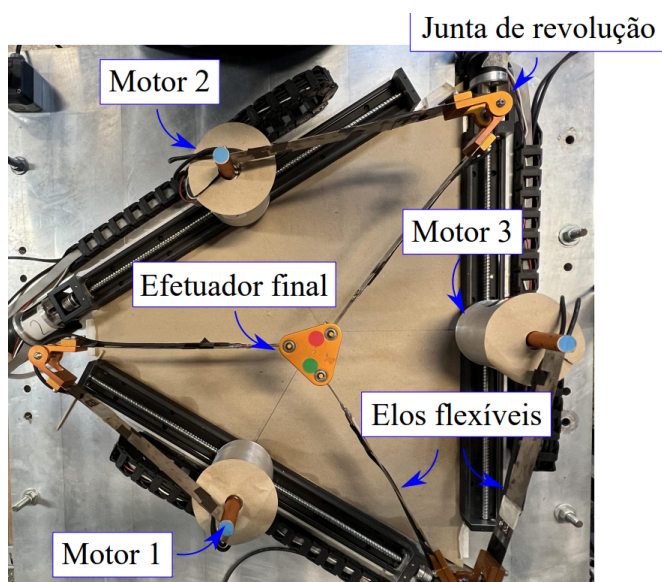
O manipulador robótico planar foi construído no Laboratório de Dinâmica da Escola de Engenharia de São Carlos (EESC-USP) em 2014 e financiado pela FAPESP 2014/01809-0 para a avaliação numérica e experimental de vários níveis de redundância cinemática em manipuladores robóticos de cinemática paralela. Em uma pesquisa posterior, um sistema de servovisão foi utilizado para a implementação de um controle no espaço da tarefa (Colombo; Carvalho Fontes; Silva, 2019). Em outro estudo, os elos foram modificados para serem flexíveis com extensômetros medindo sua deformação em função de realizar um controle híbrido livre de modelo (Silva *et al.*, 2024).

O protótipo era originalmente do tipo 3PRRR, no qual a letra P simboliza uma junta prismática, enquanto R denota uma junta de revolução. As letras sublinhadas são referentes às juntas ativas, enquanto as não sublinhadas estão relacionadas às juntas passivas. Neste trabalho, as juntas passivas são bloqueadas, ou seja, não há movimento translacional dos motores, fazendo com o que o sistema mecânico seja simplificado para um manipulador

paralelo 3RRR. Assim, o manipulador possui 3 cadeias cinemáticas, cada uma com 3 juntas de revolução, sendo a primeira ativa acionada por um motor, a segunda livre entre dois elos flexíveis e a última no efetuador final, onde todos os braços convergem.

Observando a Figura 1, é possível identificar parte da instrumentação empregada no sistema. No que se refere às referências adotadas, utilizaram-se as coordenadas cartesianas e a orientação do efetuador final para a determinar sua posição e ângulo, bem como os ângulos correspondentes a cada motor. A instrumentação e o sistema de coordenadas utilizados serão descritos de forma detalhada na seção de metodologia deste trabalho.

Figura 1 – Elos cinemáticos do 3RRR com elos flexíveis

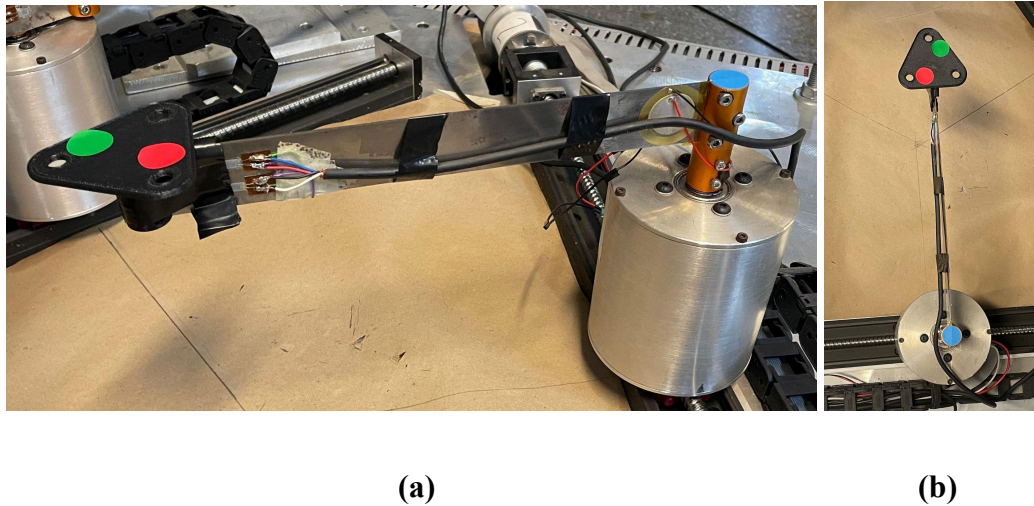


Fonte: Adaptado de SILVA et al., 2024

2.2.2 Manipulador flexível serial

A fim de explorar o uso de PINNs, um tipo de rede neural mais complexo, determinamos que o uso de um sistema mais simples seria a forma mais efetiva de verificar a viabilidade da estratégia. Assim, uma das cadeias cinemáticas do manipulador 3RRR foi desconectada, restando apenas um único elo flexível com uma junta rotativa ativa e o efetuador em sua extremidade, conforme pode ser visto nas Figuras 2 (a) e 2 (b).

Figura 2 – Manipulador flexível de elo único (a) Vista lateral do manipulador (b) Vista superior do manipulador



Fonte: Autoria própria, 2025

2.3 Aprendizado de máquina

O avanço recente de técnicas de aprendizado de máquina permitiu sua aplicação em uma ampla gama de problemas de engenharia. Dentre os diversos métodos disponíveis, esse trabalho foca nas redes neurais artificiais e as informadas por física, que serão descritas nesta seção.

2.3.1 Aplicações na engenharia

Apesar do aprendizado de máquina ser um método relativamente novo, seu uso tem se expandido rapidamente em diversas áreas de engenharia. Uma delas é a engenharia de materiais, especialmente na Indústria 4.0, permitindo prever propriedades mecânicas e reduzir custos experimentais. Redes neurais artificiais foram empregadas com sucesso em estudos sobre o uso de modelos para teste e manufatura de materiais compósitos (Esmaeili; Rizvi, 2023). Em outro estudo, o aprendizado de máquina guiado por física, em inglês *physics-guided machine learning* (PGML), foi efetivo na garantia de estabilidade em processos de usinagem ao criar modelos que evitam o *shatter* e que reduzem a necessidade de medições reais ao utilizar modelos teóricos (Greis *et al.*, 2023). O PGML também se mostrou

capaz de realizar simulações de dinâmica estrutural, onde, comparado com métodos puramente baseados em dados, ele apresentou uma capacidade de generalização melhor e um uso menor de dados. Já em relação à modelagem baseada em física, seu modelo exibiu uma eficiência computacional melhor e a capacidade de identificar física parcialmente desconhecida (Yu; Yao; Liu, 2020).

Uma das áreas que o aprendizado de máquina já é amplamente utilizado é a dos sistemas térmicos e fluídicos, isso se deve ao fato dos fenômenos complexos existentes neste campo. O método SINDy foi empregado para prever o conforto térmico em cabines de veículos elétricos, estimando variáveis como temperatura equivalente homogênea e temperatura média radiante com alta precisão e baixo tempo de processamento, superando limitações dos modelos puramente baseados em física (Warey; Kaushik; Han, 2022). Redes neurais recorrentes e profundas também foram utilizadas na previsão de campos térmicos e em um estudo que aplicou isso na manufatura aditiva a laser, os modelos gerados atingiram alta concordância com simulações numéricas, permitindo otimizar estratégias de deposição e reduzir tensões residuais (Ren *et al.*, 2020). O aprendizado de máquina também já foi utilizado em experimentos dinâmicos, automatizando a seleção de parâmetros em testes de vibração induzida por vórtices e reduzindo drasticamente o número de ensaios necessários para mapear respostas não lineares em sistemas fluido-estruturais (Fan *et al.*, 2019). Este último estudo também cita que a metodologia utilizada também poderia ser aplicada na mecânica dos sólidos e em processos de manufatura, algo que já foi comprovado nos exemplos anteriores.

Por fim, o aprendizado de máquina foi utilizado em sistemas dinâmicos, a área de foco deste trabalho. Uma extensão do método SINDy com parâmetros de controle foi capaz de identificar as equações diferenciais que regem sistemas com parâmetros de controle ajustáveis diretamente dos dados experimentais. Essa metodologia tem sido aplicada com sucesso em sistemas que exibem formação de padrões e bifurcações, permitindo reconstruir dependências não lineares e prever comportamentos complexos mesmo em presença de ruído experimental (Nicolau *et al.*, 2023). A abordagem do PIML também foi explorada na previsão probabilística de respostas em sistemas dinâmicos, combinando modelos físicos com modelos de aprendizado capazes de representar e corrigir discrepâncias entre previsões teóricas e observações experimentais. Essa integração permite que o sistema opere com entradas aleatórias não estacionárias e reduza o custo computacional de análises de confiabilidade, ao

mesmo tempo que apresenta resultados robustos em aplicações como vigas submetidas a carregamentos dinâmicos e escoamentos hipersônicos sobre painéis flexíveis (Subramanian; Mahadevan, 2023).

Em todas as aplicações apresentadas, o aprendizado de máquina baseado em física apresentou resultados confiáveis, onde o método produziu modelos eficientes que previram corretamente os sistemas estudados. Isso demonstra a adaptabilidade e efetividade do método em diversos campos de estudo, reforçando a proposta deste trabalho de utilizá-lo para a modelagem de um manipulador robótico paralelo com elos flexíveis.

2.3.2 Redes neurais artificiais

As redes neurais artificiais são modelos de inteligência artificial inspirados no funcionamento do cérebro humano, que usam nós interconectados em camadas para aprender e processar dados, de forma similar que os neurônios reais são utilizados. Elas funcionam aprendendo padrões complexos através do treinamento com exemplos, ajustando pesos e vieses nas conexões entre os neurônios artificiais para gerar previsões ou decisões precisas. Isso permite que elas realizem o aprendizado profundo, que é capaz de aprender padrões complexos diretamente de grandes volumes de dados (Schmidhuber, 2015), foram estes motivos que levaram à sua escolha para a tarefa de modelagem do trabalho.

A rede neural utilizada foi do tipo *feedforward* com múltiplas camadas ocultas, projetada para mapear relações não lineares entre variáveis de entrada e estados dinâmicos do sistema. A arquitetura proposta é composta pelos estágios de entrada, camadas ocultas e saída. O nome desse tipo de rede se dá pelo fato de que informações são passadas e transformadas de nós de uma camada anterior para os nós de uma próxima camada (Choi *et al.*, 2020).

2.3.3 Aprendizado de máquina informado por física

O aprendizado de máquina informado por física, em inglês *physics-informed machine learning* (PIML), é um novo método que utiliza dados em conjunto com conhecimentos prévios para formar modelos. Essa abordagem, diferentemente de redes neurais profundas, não necessita de um grande banco de dados para funcionar, pois utilizar conhecimentos prévios, como leis físicas e observações, cria restrições teóricas, vieses indutivas e observacionais. Esses fatores aumentam o desempenho do algoritmo de aprendizado,

requerendo menos dados para gerar modelos menores e eficientes para controle (Karniadakis *et al.*, 2021). Outra vantagem desse método é a sua capacidade de integrar perfeitamente dados e equações diferenciais parciais com informações físicas completas ou incompletas (Willard *et al.*, 2022).

O método SINDy é um tipo de PIML que utiliza regressão esparsa para obter sistemas dinâmicos não lineares de dados de medição. O seu funcionamento é baseado na análise de variáveis de estado, onde o algoritmo examina a variação dessas no tempo e, a partir disso, infere equações governantes do sistema, criando modelos inerentemente interpretáveis e generalizáveis. Esse pacote também nos permite configurar o algoritmo, deste modo, podemos ajustar as variáveis de estado segundo as informações adicionais que obtemos pela observação do manipulador e leis físicas (De Silva *et al.*, 2020).

Neste trabalho, utilizamos o modelo rígido do manipulador como base e buscamos representar a flexibilidade através de termos adicionais. Com isso, utilizou-se a estratégia do SINDy como inspiração na forma de identificar a contribuição da flexibilidade na dinâmica do sistema e obter os componentes que a representam na rede neural informada por física do manipulador flexível serial.

2.4 Softwares utilizados

2.4.1 MSC Adams

Os modelos virtuais são uma forma rápida de realizar simulações e experimentos sem desgastar ou desenvolver equipamentos físicos, facilitando a obtenção de dados do sistema. Sendo assim, modelos foram criados no software Adams MSC. A escolha deste software se deve ao fato dele possuir a função de simular corpos flexíveis, sendo um dos mais utilizados no estudo da dinâmica e distribuição de forças de sistemas mecânicos. Ele é empregado tanto na indústria quanto na pesquisa, como em um estudo similar, no qual a simulação de um manipulador planar paralelo 3RRR foi realizado no mesmo (Varedi-Koulaei; Daniali; Farajtabar, 2016).

2.4.2 Google Colab

O Google Colab é uma plataforma baseada em nuvem que suporta a execução de programas em Python. Sua característica principal é a disponibilização de recursos computacionais sem a necessidade de uma infraestrutura local, o que facilita a implementação e o treinamento de redes neurais. Além disso, ele possui integração com bibliotecas de aprendizado de máquina como TensorFlow e PyTorch, para possibilitar a organização e compartilhamento de experimentos em ambiente colaborativo. Devido a esses fatos, o Google Colab foi escolhido como a plataforma para se desenvolver a programação deste trabalho.

2.4.3 MATLAB

O MATLAB é uma plataforma de programação e cálculo numérico amplamente empregada em engenharia e pesquisa científica. Suas ferramentas incluem bibliotecas para análise de sinais, controle, otimização e simulação de sistemas dinâmicos. Nesta pesquisa, o MATLAB foi aplicado na aquisição dos dados de treinamento e posteriormente na implementação do controle dos manipuladores robóticos a partir dos modelos obtidos.

3 METODOLOGIA

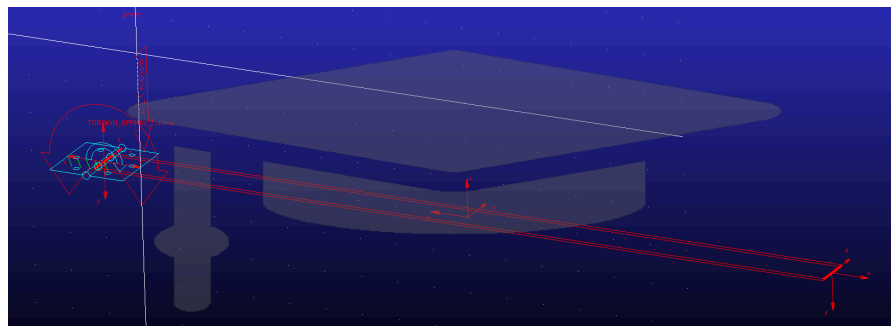
3.1 Modelos virtuais no software MSC Adams

O software MSC Adams foi utilizado como ferramenta de simulação dinâmica para gerar dados sintéticos para a validação inicial das técnicas de PIML aplicadas neste trabalho. Foram modelados dois sistemas, ambos representando a dinâmica de um elo único: um pêndulo rígido, como referência clássica, e um pêndulo flexível, capaz de reproduzir os efeitos de deformação estrutural.

3.1.1 Pêndulo rígido

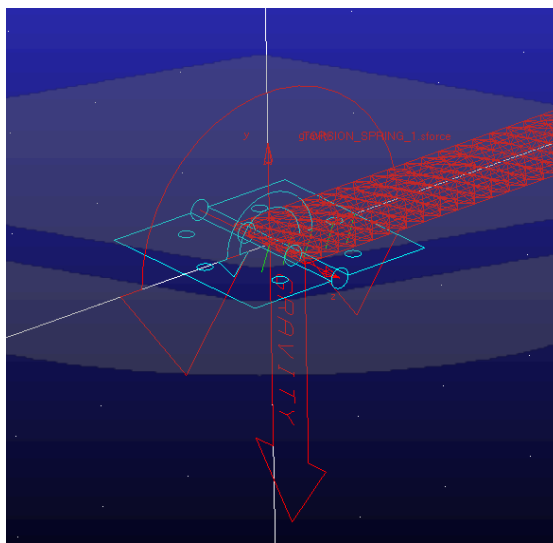
O modelo do pêndulo rígido foi implementado no MSC Adams como um corpo indeformável, preso a uma extremidade por uma junta de revolução, como pode ser visto na Figura 3. O elo é composto por uma barra com dimensões de 1000 mm de comprimento, 50 mm de largura e espessura de 3 mm com massa de 1,17 kg. A fim de simular a resistência do ar e atrito, uma mola de torção com amortecimento de 6 N.mm.s/grau foi colocada no pivô do pêndulo, vista na Figura 4. Os dados registrados foram as coordenadas cartesianas da extremidade livre da barra.

Figura 3 – Vista completa do pêndulo rígido



Fonte: Autoria própria, 2025

Figura 4 – Vista focada no ponto de fixação do pêndulo rígido



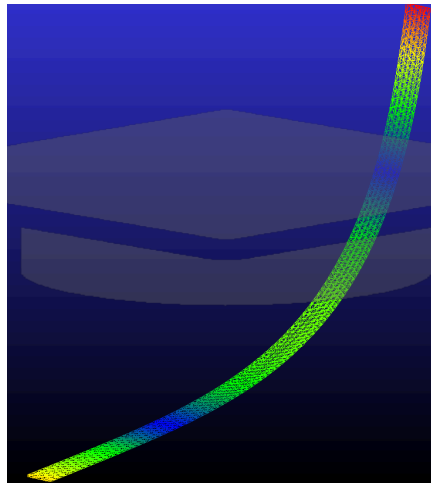
Fonte: Autoria própria, 2025

3.1.2 Pêndulo flexível

O modelo do pêndulo flexível foi criado a partir da mesma geometria, mas utilizando a ferramenta ViewFlex do MSC Adams, que permite a representação de deformações estruturais por meio da integração de elementos finitos ao corpo rígido.

As propriedades mecânicas adotadas para o elo foram: módulo de elasticidade (E) de 7000 N/mm^2 e coeficiente de Poisson (ν) de 0,29. Esses valores foram escolhidos para permitir que a flexibilidade tenha um efeito significativo e seja visualizada durante o movimento do pêndulo. A Figura 5 apresenta a barra durante o movimento, evidenciando sua deformação e a distribuição dos esforços internos.

Figura 5 – Pêndulo flexível durante o movimento oscilatório

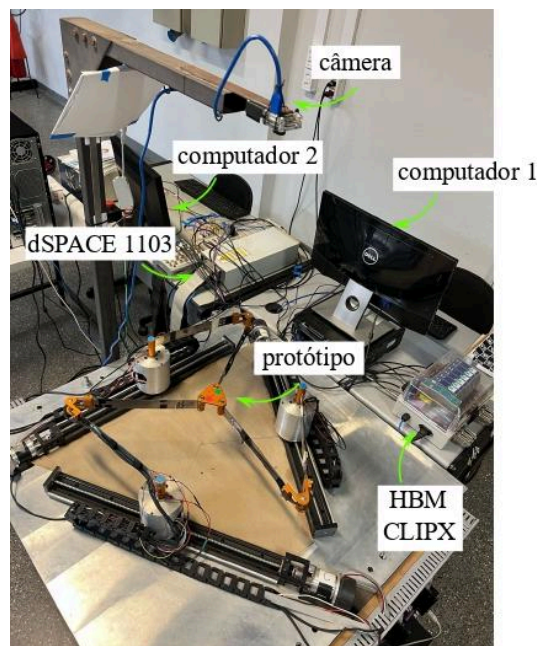


Fonte: Autoria própria, 2025

3.2 Instrumentação do manipulador paralelo 3RRR com elos flexíveis

Esta subseção apresenta a instrumentação e os equipamentos utilizados para a comunicação do manipulador paralelo 3RRR com elos flexíveis. A Figura 6 mostra a instrumentação utilizada para a construção do experimento.

Figura 6 – Instrumentação empregada para o experimento



Fonte: Autoria própria, 2025

3.2.1 Componentes

- Motores de atuação: 3 motores Maxon EC60 plano sem escovas com 100 W de potência de entrega e uma corrente nominal de 2.3 A, acoplados com redutores planetários GP52 C de redução 3,5:1, fornecendo uma rotação nominal de 1200 rpm e torque nominal de 0,82 N.m. Cada motor possui um controlador Maxon EPOS2 50/5 com uma fonte de energia de até 50 Vdc e uma corrente de 5 A. Vale notar que os motores possuem codificadores, que medem a posição angular dos atuadores.
- Elo flexível: fabricado em aço mola AISI 6150. Cada elo apresenta um comprimento de 318,5 mm, largura de 3,5 mm e espessura de 0,7 mm. Devido ao baixo peso e espessura dos elos, eles são flexíveis e a deformação neles gera vibrações no sistema.
- Extensômetros: do tipo HBM 350-E, instalados em configuração de ponte completa na base do elo, utilizados para medir deformações associadas à flexibilidade. O sinal adquirido passa por amplificadores de sinal HBM HB40 CLIPX antes do envio à placa de controle.
- Placa de controle dSPACE 1103: interface central de aquisição e processamento, responsável pela leitura dos sensores, execução das leis de controle e envio dos sinais de comando ao motor.
- Câmera oCam-5CRO-U: utilizada para coletar imagens em tempo real do manipulador paralelo, requer uma interface USB 3.0 e apresenta uma resolução de 640x480 pixels, com uma taxa de quadros máxima de 60 quadros por segundo.
- Computador 1: responsável pelo processamento das imagens obtidas pela câmera para a estimativa da posição do efetuador.
- Computador 2: dedicado ao desenvolvimento do controle em MATLAB/Simulink.

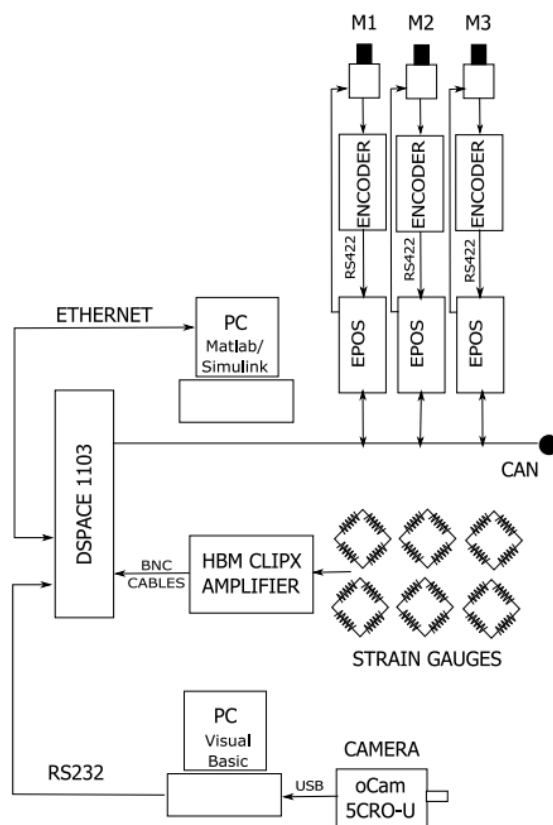
3.2.2 Esquema de conexões e funcionamento

Cada cadeia cinemática do manipulador é acionada por um motor instalado em sua junta ativa, controlado pela placa dSPACE 1103. A comunicação entre os controladores EPOS e o dSPACE 1103 é conduzida via protocolo CAN a uma taxa de transmissão de 250 kbit/s, e a Ethernet é usada para estabelecer a comunicação entre a placa e o computador.

A câmera é fixada acima do protótipo a fim de obter imagens de todo o espaço de trabalho. Como mencionado, as imagens, capturadas pela câmera, são processadas pelo Computador 1 para determinar a posição do efetuador final. Note que, o manipulador paralelo é considerado plano, uma vez que o movimento é restrito ao longo das direções x e y , enquanto $z = 0$. Após as imagens serem capturadas pela câmera, o algoritmo de processamento de imagens (Geronel et al. (2024)) é usado para estimar a posição linear e angular do efetuador final.

Devido ao baixo peso dos elos e consequentemente a alta flexibilidade, características como vibração indesejada pode ser ocasionada no protótipo, gerada principalmente pela deformação dos elos. Assim, os extensômetros são fixados em cada elo, visando coletar em tempo real a deformação de cada elo flexível. O sinal coletado é enviado para amplificadores de sinal HBM HB40 CLIPX, e então enviados a dSPACE 1103. Portanto, torna-se importante observar que todo o sinal medido é coletado através da câmera e dos extensômetros, codificados e levados para a dSPACE, conforme pode ser observado no esquema da Figura 7.

Figura 7 – Esquema de comunicação do 3RRR com elos flexíveis



Fonte: Adaptado de SILVA et al., 2024

A dSPACE envia os dados adquiridos ao computador 2 que projeta a estratégia de controle, realizada em Matlab/SIMULINK, a fim de gerar um sinal de controle apropriado. Este sinal é então encaminhado para os controladores dos motores, através da dSPACE, realizando o movimento do protótipo 3RRR.

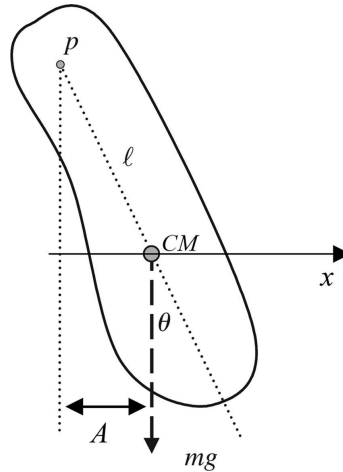
3.3 Modelos físicos e cinemáticos

A formulação de modelos físicos e cinemáticos serve como referência para a análise dos sistemas estudados e para a incorporação das leis da dinâmica nas redes neurais. Nesta seção, são apresentados os modelos e sistemas de referência do pêndulo rígido, manipulador flexível serial e manipulador paralelo 3RRR.

3.3.1 Pêndulo Rígido

O primeiro modelo, ilustrado na Figura 8, descreve o movimento de um pêndulo simples subamortecido rígido.

Figura 8 – Representação do pêndulo físico



Fonte: (Bonventi Júnior; Aranha, 2015)

Utilizando as seguintes equações baseadas em um estudo sobre oscilações amortecidas (Bonventi Júnior; Aranha, 2015):

$$\frac{d^2\theta}{dt^2} + \frac{b}{I_p} \frac{d\theta}{dt} + \frac{mgl}{I_p} \theta = 0 \quad (1)$$

A Equação 1 é uma equação diferencial ordinária de segunda ordem que descreve o movimento de um pêndulo amortecido, onde θ é a posição angular do pêndulo, b é o coeficiente de amortecimento angular, l é a distância do pivô ao centro de massa, g é a aceleração da gravidade, m é a massa total da barra e I_p é o momento de inércia.

$$\theta(t) = Ae^{-bt/2I_p} \cos(\omega t + \phi) \quad (2)$$

A solução da Equação 1, apresentada na Equação 2, é análoga ao caso do sistema massa-mola amortecido. Como o movimento se inicia com amplitude A máxima, temos que a fase $\phi = 0$. Por fim, a frequência natural de oscilação harmônica do pêndulo é definida pela

Equação 3, a frequência de oscilação com amortecimento é dada na Equação 4 e o momento de inércia da barra é relacionado pela Equação 5.

$$\omega_0 = \sqrt{mgl/I_p} \quad (3)$$

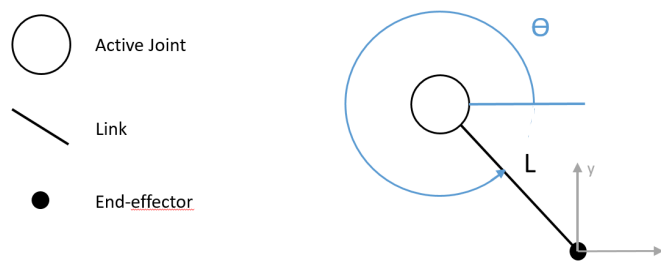
$$\omega = \sqrt{\omega_0^2 - (b/2I_p)^2} \quad (4)$$

$$I_p = m[(h^2 + w^2)/12 + l^2] \quad (5)$$

3.3.2 Manipulador flexível serial

O segundo modelo cobre o manipulador de único elo, sendo uma simplificação do 3RRR original. Este modelo é ilustrado na Figura 9, apresentando as características geométricas do sistema.

Figura 9 – Características geométricas do manipulador flexível serial



Fonte: Autoria própria, 2025

A posição angular da junta ativa é definida por θ , e a distância do centro da junta ativa até o efetuador final é L , o comprimento do elo. Vale notar que este θ representa o ângulo da extremidade do manipulador, diferente do θ dos codificadores. Dada a simplicidade do manipulador, as posições x e y do efetuador final podem ser descritas pelas seguintes relações:

$$x = L \cos (\theta) \quad (6)$$

$$y = L \sin (\theta) \quad (7)$$

Também podemos realizar cinemática inversa para resolver θ e determinar as restrições do sistema:

$$\tan(\theta) = \frac{y}{x} \quad (8)$$

$$\sqrt{x^2 + y^2} = L \quad (9)$$

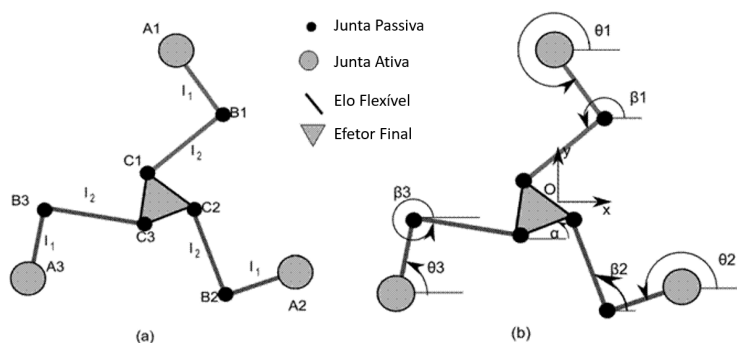
As Equações 6, 7 e 8 descrevem o comportamento rígido de um manipulador 1, e a Equação 9 representa sua restrição. A partir disso, técnicas de aprendizado de máquinas são utilizadas, com o objetivo de através de uma equação simplificada de um elo rígido, obter as equações de um elo com estrutura flexível. Assim, permite-se utilizar o aprendizado de máquina como uma extensão da modelagem matemática do sistema mecânico, possibilitando compensar efeitos de incertezas, defeitos de produção e erros operacionais.

3.3.3 Manipulador 3RRR

Conforme foi descrito na seção 2.2.1 , o manipulador paralelo 3RRR com elos flexíveis é composto por três cadeias cinemáticas idênticas, cada uma formada por uma junta

de revolução ativa e duas passivas, interligando a base fixa ao efetuador final. Podemos observar as referências do sistema na Figura 10.

Figura 10 – Características geométricas do 3RRR



Fonte: Adaptado de Ruiz et al, 2018

Os ângulos θ indicam a posição angular das juntas ativas com os motores e a posição do efetuador final é dada pelas coordenadas x e y , enquanto a sua orientação se dá pelo ângulo α . Como aplicamos apenas uma rede neural do tipo *feedforward* neste modelo, não foi necessário criar um modelo matemático do manipulador 3RRR completo. Por este motivo, utilizaremos apenas o sistema de referências na modelagem para o aprendizado de máquina.

3.4 Coleta e manipulação de dados

A coleta e processamento de dados é uma etapa essencial para assegurar a qualidade das informações utilizadas no treinamento das redes neurais e nos métodos complementares de aprendizado, fazendo com que os modelos obtidos sejam consistentes. O processo incluiu tanto a exportação de dados sintéticos gerados em simulação, quanto o tratamento de dados experimentais coletados em bancada.

3.4.1 Coleta e exportação de dados virtuais

Os modelos do pêndulo rígido e flexível, desenvolvidos no MSC Adams, foram configurados para registrar as posições cartesianas x e y da extremidade livre em ambos os pêndulos e, no caso do flexível, a deformação no nódulo com maior valor medido. A duração das simulações foi de 10 segundos, realizadas com taxa de amostragem de 10 e 1000 medições por segundo, sendo a maior quantidade de medições empregada para aumentar a precisão dos dados. Os dados obtidos foram exportados para uma tabela no Google Sheets,

onde as coordenadas x e y foram utilizadas para calcular a posição angular θ do pêndulo por meio da função ATAN. Para utilizar esses dados no Google Colab, foi empregada a biblioteca Pandas, que permite inserir os dados de uma coluna da tabela em uma variável no Python.

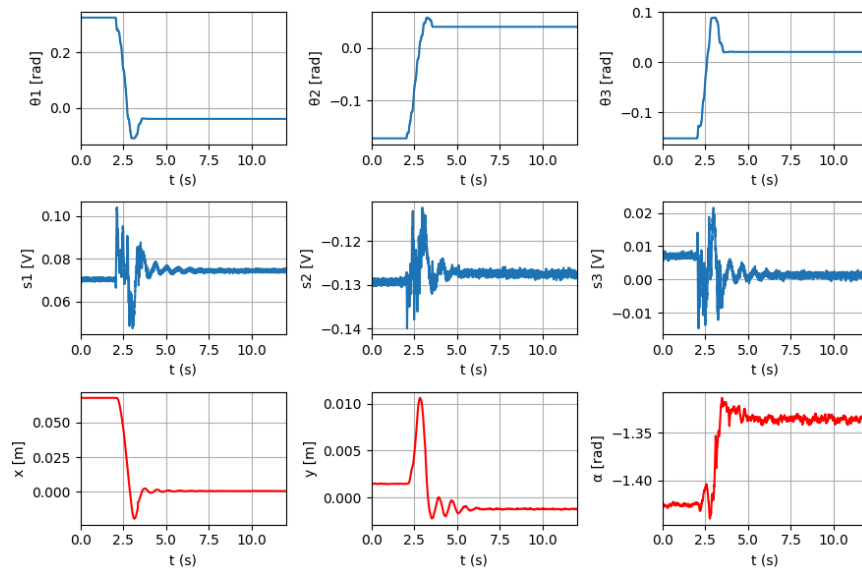
Através dessa primeira abordagem, foi possível compreender o funcionamento e as características de um elo rígido e flexível, permitindo a investigação experimental do sistema proposto.

3.4.2 Coleta de dados reais

Para a coleta dos dados reais, foi utilizado a estratégia sem modelo, através da utilização de um controlador de modos deslizantes. Através dessa estratégia, é possível combinar as imagens do efetuador final e a deformação medida nos extensômetros, a fim de estabelecer a posição angular e linear do efetuador. Assim, o controle desenvolvido é dividido em dois laços: o laço da câmera, que estima a posição do efetuador final, e o laço de deformação, responsável por compensar a flexibilidade do protótipo. A câmera, fixada acima do protótipo, é utilizada para coletar as imagens em tempo real, e através da utilização do algoritmo de processamento de imagens, a posição do efetuador final é estimada. Neste caso, o controlador de modos deslizantes é empregado devido a sua robustez e simplicidade. Baseado no erro entre os estados desejado e real, calculados em cada iteração, a lei de controle correspondente pode ser calculada. Em seguida, baseada no erro entre as deformações obtidas pelos extensômetros e o valor do sinal desejado, uma lei de controle é estabelecida.

Assim, diferentes trajetórias, definidas como entrada do tipo degrau, são utilizadas para a coleta dos dados. O experimento é executado diversas vezes, a fim de estabelecer um conjunto de dados. Com isso, o banco de dados foi dividido entre treinamento, teste e validação. Os dados treinados e testados não serão os mesmos utilizados para a validação. Vale notar que o procedimento descrito foi utilizado tanto no manipulador 3RRR completo quanto em sua versão simplificada. Temos na Figura 11 um exemplo de conjunto de dados, com as entradas do sistema em azul e as saídas em vermelho.

Figura 11 – Amostra do conjunto de dados utilizados no treinamento e teste da rede neural



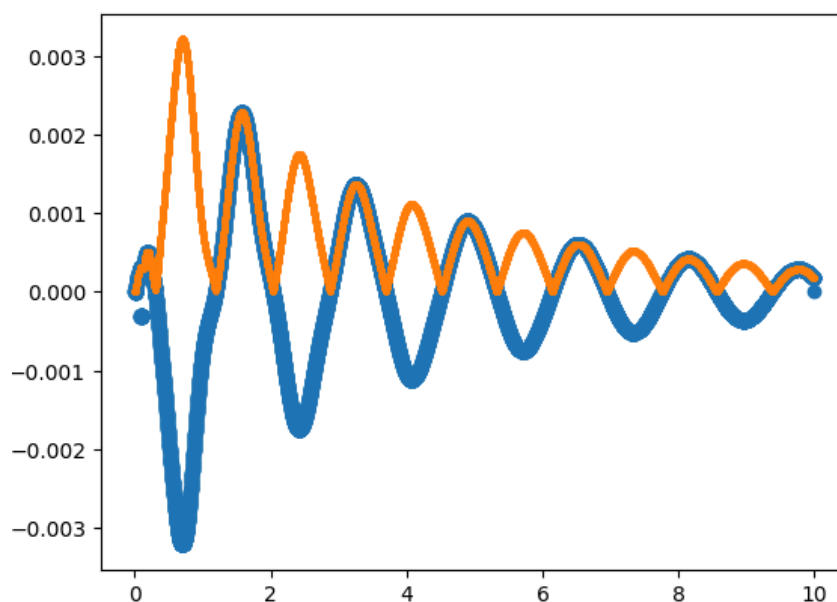
Fonte: Autoria própria, 2025

Observe que foram necessários três sensores diferentes para adquirir os sinais: uma câmera, codificadores e extensômetros. Isso aumenta a complexidade do protótipo, emprega diferentes premissas de manipuladores paralelos com enlaces rígidos e exige um tempo de processamento de custo superior. As redes neurais artificiais são projetadas visando reconhecer padrões complexos de sistemas mecânicos com uma quantidade menor de dados e, possivelmente, como uma forma de retirar a necessidade de sensores adicionais.

3.4.3 Tratamento de dados

O tratamento dos dados buscou filtrar ruídos, eliminar inconsistências e preparar os sinais para a etapa de treinamento das redes neurais. Com exceção do caso do pêndulo rígido, todos os conjuntos de dados foram normalizados utilizando o valor máximo das entradas e saídas. Eles também passaram por um processo de centralização, removendo partes iniciais e finais dos dados que possuíam inconsistências que poderiam afetar o processo de aprendizado. É importante destacar que os valores da deformação gerados pelo MSC Adams são absolutos e apenas positivos, assim, como forma de assimilar a condição real, utiliza-se uma função que altera o sinal da deformação a cada meio período, simétrico em torno de zero. Observa-se na Figura 12, a comparação do sinal original em laranja com o simétrico em azul.

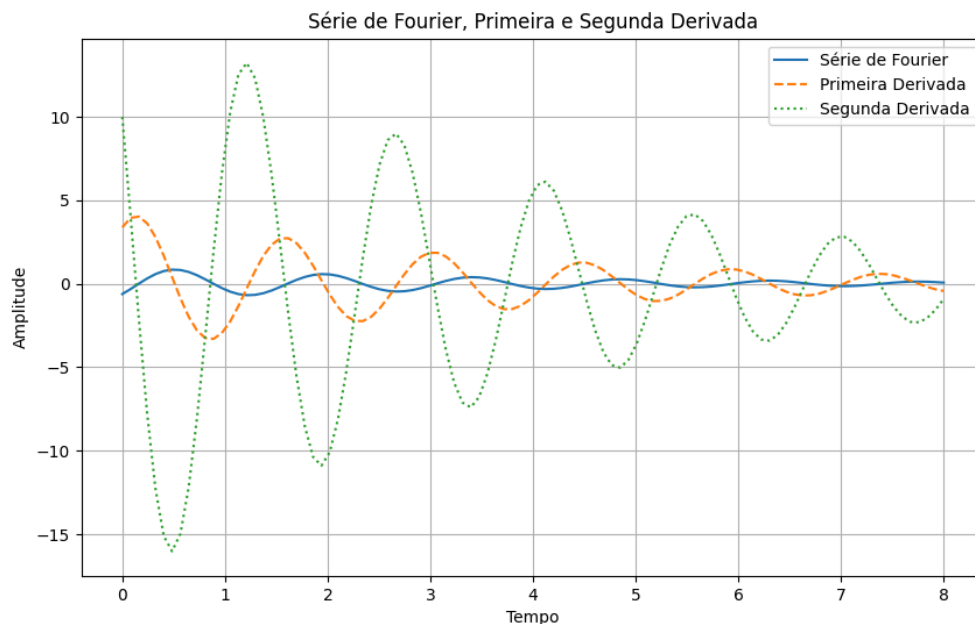
Figura 12 – Simetrização do sinal de deformação em torno e zero



Fonte: Autoria própria, 2025

Adotamos outras estratégias no processamento das medidas de deformação, como a análise do espectro de frequência, que auxiliou na identificação dos componentes principais do sinal e na separação de vibrações relevantes de ruídos de alta frequência, levando a escolha de filtros digitais. Incluindo o Bandstop de segunda ordem (Notch), os filtros passa-alta e passa-baixa de segunda ordem para remover ruídos e preservar o comportamento dinâmico essencial do sistema. Outro método utilizado foi uma aproximação por série de Fourier através de um Rede Neural de Fourier, a fim de filtrar pontos discrepantes e facilitar a diferenciação, o funcionamento desta rede neural será explicada em detalhe na seção 3.5.4. Após a aproximação ser encontrada, a função `autograd.grad` da biblioteca Pytorch foi aplicada para obter a primeira e segunda derivada da deformação, que foram utilizadas na investigação do treinamento dos modelos. A derivação dos sinais de deformação pode ser observada na Figura 13.

Figura 13 – Aproximação por série de Fourier com a primeira e segunda derivada



Fonte: Autoria própria, 2025

No entanto, as estratégias discutidas de filtragem e aproximação foram empregadas apenas no pêndulo flexível e o modelo do manipulador serial com 1 conjunto de dados de treinamento. Para a análise do manipulador 3RRR completo e do modelo final do manipulador serial, os dados reais utilizados não possuíam ruído significativo e não necessitavam de um filtro ou da aproximação pela série de Fourier, assim, as medidas das deformações foram apenas normalizadas e centralizadas. Fora isso, estes últimos casos utilizam múltiplos conjuntos de dados no seu treinamento, fazendo com que o tempo de processamento aumente significativamente, necessitando a simplificação descrita.

Esse conjunto de procedimentos assegurou que os dados utilizados nas etapas de treinamento fossem consistentes, livres de distorções e representassem com fidelidade a dinâmica real e simulada do sistema, aumentando a confiabilidade dos modelos obtidos.

3.5 Modelos de aprendizado de máquina

Esta seção visa explicar como os métodos de aprendizado de máquina foram utilizados na construção de modelos capazes de representar a dinâmica de sistemas rígidos e flexíveis, assim como as configurações utilizadas neles.

3.5.1 Rede neural *feedforward*

A rede neural responsável por modelar o manipulador 3RRR é do tipo *feedforward* com múltiplas camadas ocultas, projetada para mapear relações não lineares entre variáveis de entrada e estados dinâmicos do sistema. Ela é composta por três estágios: entrada, camadas ocultas e saída. As entradas correspondem às posições angulares dos motores ($\theta_1, \theta_2, \theta_3$) e aos sinais dos extensômetros (s_1, s_2, s_3), enquanto a saída representa a posição da extremidade do efetuator (x, y, α). Com relação às camadas ocultas, há neurônios que irão processar informações e aprender uma tarefa específica. A rede neural recebe uma entrada u e a propaga pelas camadas ocultas l , aplicando os pesos W e viés b , conforme a Equação 10.

$$u^{l+1} = W^l u^l + b^l, \text{ para } l = 1, \dots, L - 1 \quad (10)$$

Em seguida, um operador de soma realiza a soma dos sinais ponderados de cada sinapse l do neurônio. A função de ativação é então aplicada a cada camada oculta para limitar a saída do neurônio a um determinado intervalo e melhorar as respostas não lineares do algoritmo (Garpeilli et al., 2023).

$$ReLU(u) = \max(0, u) \quad (11)$$

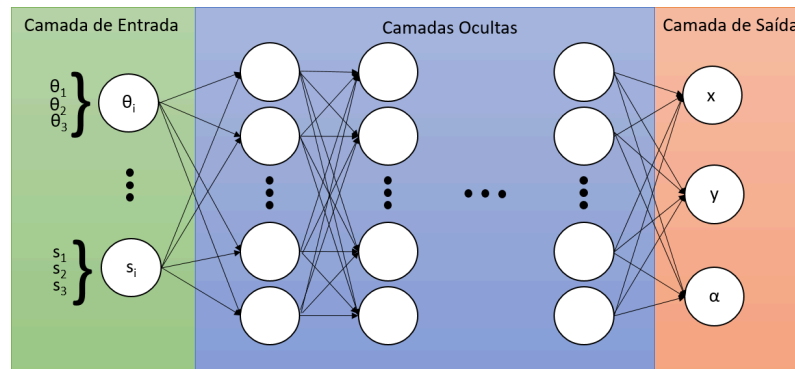
A função ReLU (*Rectified Linear Unit*) é definida na Equação 11, que retorna u se u for maior do que zero, caso contrário, retorna zero. O procedimento utilizado na etapa de aprendizado minimiza a função objetivo para modificar os valores dos pesos e vieses a fim de alcançar a minimização desejada. Para realizar esse processo, uma função de perda deve ser definida para comparar a saída da rede com o valor esperado. No nosso caso, o erro é estimado pela técnica do Erro Quadrático Médio (MSE), conforme a Equação 12.

$$L = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (12)$$

As variáveis y_i e \hat{y}_i representam os valores reais e estimados pela rede neural, respectivamente, e N é o comprimento dos vetores. O processo de aprendizado é implementado usando o método do gradiente. A função de perda (Equação 12) é usada para propagar os erros da camada de saída até a primeira camada de cada neurônio, as derivadas parciais do MSE são calculadas com relação aos pesos e vieses correspondentes. Com base nas derivadas, calcula-se o gradiente da função de perda, e o uso do método da descida do gradiente permite a atualização dos valores de pesos e vieses para realizar o aprendizado da

rede neural. Para a implementação, os dados foram divididos em dois conjuntos: o primeiro relativo aos dados para treino e teste, enquanto o segundo para validação. O primeiro grupo foi ainda dividido em 90% dos dados para treino e 10% para teste. A Figura 14 ilustra a rede neural adotada para o trabalho.

Figura 14 – Esquemático da Rede Neural do tipo *feedforward* utilizada

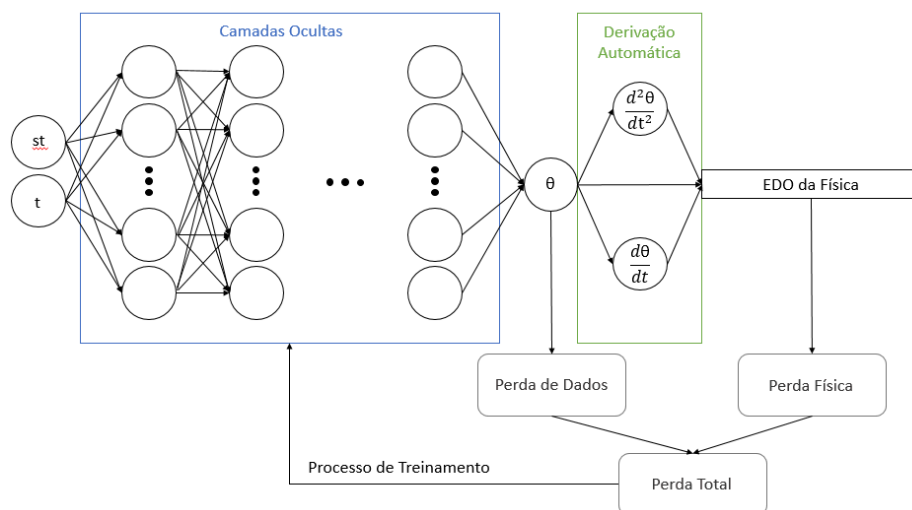


Fonte: Autoria própria, 2025

3.5.2 Rede neural informada por física

Nos casos do manipulador flexível serial, pêndulo rígido e pêndulo flexível, utilizamos redes neurais informadas por física, do inglês *physics-informed neural networks* (PINNs), para a modelagem de sua dinâmica, seu funcionamento pode ser observado na Figura 15.

Figura 15 – Esquemático do PINN utilizado



Fonte: Autoria própria, 2025

A Figura 15 destaca a principal diferença entre um PINN e uma rede neural comum: o fato do PINN possuir duas funções de perda. Enquanto uma rede neural considera apenas a força de perda, obtida através dos dados medidos, o PINN inclui uma função de perda adicional, relativo à física do sistema. Isso auxilia na robustez da rede neural, especialmente em condições com baixo número de dados coletados. Um dos parâmetros de ajuste na arquitetura da rede neural é a relação de peso no treinamento entre essas duas funções de perda.

A configuração final do PINN foi obtida após múltiplos testes com os diferentes sistemas modelados. Ela buscou permitir que a rede aprendesse tanto as relações empíricas quanto às restrições impostas pela física enquanto equilibra precisão, robustez a ruídos experimentais e viabilidade computacional.

A arquitetura do PINN também é do tipo *feedforward*, possuindo 5 camadas ocultas, 40 neurônios por camada oculta e função de ativação da tangente hiperbólica (\tanh), escolhida por ser adequada para sinais oscilatórios, facilitando a representação de dinâmicas periódicas. No entanto, a função de perda se diferencia da rede neural anterior por possuir dois componentes, a perda física ($loss1$) e a perda de dados ($loss2$). Associadas, respectivamente, à satisfação da equação diferencial parcial que descreve o sistema e a diferença entre as saídas previstas pela rede e as observações experimentais. O peso das perdas foi configurado como igual, conforme demonstrado na Equação 13.

$$loss = 10^{-1} \cdot loss1 + 10^{-1} \cdot loss2 \quad (13)$$

Foram utilizadas duas variáveis de entrada para o PINN, o tempo (t_{physics}) e a deformação medida no extensômetro (st), a saída foi definida como uma única variável dependente, o ângulo de deslocamento (θ). A deformação foi escolhida como uma das entradas do PINN, visto que ela está relacionada à flexibilidade e é possivelmente uma forma de representá-la. A sua derivada foi utilizada apenas no caso do pêndulo flexível, pois presumimos que ela representaria a flexibilidade de forma mais efetiva. Porém, não observamos mudanças significativas com o valor original da deformação, por este motivo que a derivação não foi utilizada no modelo do manipulador flexível serial.

Além dos pesos e vieses da rede, foram definidos parâmetros físicos “ k ” com a função de capturar a flexibilidade do sistema. Eles representam constantes na formulação da equação

física e sua identificação foi uma parte integral do processo de aprendizado, em linha com o conceito do método SINDy abordado anteriormente. As Equações 14 e 15 representam a física utilizada no caso do pêndulo flexível e manipulador serial flexível, respectivamente.

$$\frac{d^2\theta}{dt^2} + \frac{b}{I_p} \frac{d\theta}{dt} + \frac{mgl}{I_p} \theta + k \frac{dst}{dt} = 0 \quad (14)$$

$$\frac{d^2\theta}{dt^2} + k1.st + k2.st^2 + k3.sen(st) + k4^{st} = 0 \quad (15)$$

A escolha dos múltiplos parâmetros físicos “k” foi realizada visando modelar componentes distintos da dinâmica do sistema. Enquanto k1 descreve um termo de crescimento linear no tempo, k2 e k4 introduzem um comportamento polinomial e exponencial, representando a não linearidade do sistema. Por fim, o termo k3 modela um comportamento oscilatório, relacionado aos modos vibratórios naturais do elo flexível.

Em relação ao algoritmo de otimização, utilizamos o Adam com uma taxa de aprendizado inicial de 1.10^{-3} , devido à sua eficiência em problemas não lineares. Durante o treinamento, foi adotada a política de taxa de aprendizado OneCycleLR, que consiste em variar a taxa de aprendizado em um único ciclo de treinamento. Inicialmente, ocorre um aumento gradual da taxa de aprendizado até um valor máximo, em seguida, a taxa é reduzida progressivamente até valores pequenos. Esta foi a política de taxa de aprendizado que forneceu os melhores resultados, ela foi configurada com uma taxa máxima de aprendizagem de 0,01 e um número total de passos de 5001.

O PINN também possui uma função de *early stopping*, a fim de interromper o treinamento antes do final do número de passos, caso não ocorra uma evolução significativa, assim conservando recursos e permitindo que modelos sejam obtidos em menos tempo. No caso, foi utilizado um tamanho de lote de 4001, tolerância de convergência de 1.10^{-5} , paciência de *early stopping* de 100 (com interrupção após 20 verificações consecutivas sem melhoria) e um *clipping* de gradiente de 1,0. Por fim, o PINN utilizou a combinação de 9 conjuntos de dados para o seu treinamento e um décimo conjunto para a sua validação,

assegurando que os resultados obtidos possuam maior confiabilidade e capacidade de generalização.

3.5.3 Rede Neural de Fourier

A rede neural de Fourier (FNN) foi utilizada como um método de aprendizado de máquina auxiliar no processamento dos dados do pêndulo flexível e nos primeiros testes do manipulador flexível serial. Ao contrário das redes *feedforward* tradicionais, que utilizam funções de ativação convencionais, como ReLU ou Tanh, as FNNs empregam funções harmônicas periódicas como parte de suas ativações ou como *features* adicionais na entrada da rede.

$$x(t) = \sum_{n=1}^N (a_n \sin(2\pi f_n t) + b_n \cos(2\pi f_n t)) e^{-\lambda_n t} \quad (16)$$

A FNN foi implementada conforme a Equação 16, visando identificar os coeficientes “a” e “b”, além da frequência “f” e do fator de decaimento da série de Fourier “λ”. O coeficiente “N” é o parâmetro que define o número de termos da série de Fourier, assim, podemos ajustar a aproximação baseado na complexidade do sinal original. No caso do pêndulo flexível, utilizou-se $N = 1$, pois isso resultava em uma série mais simples e adequada para a derivação.

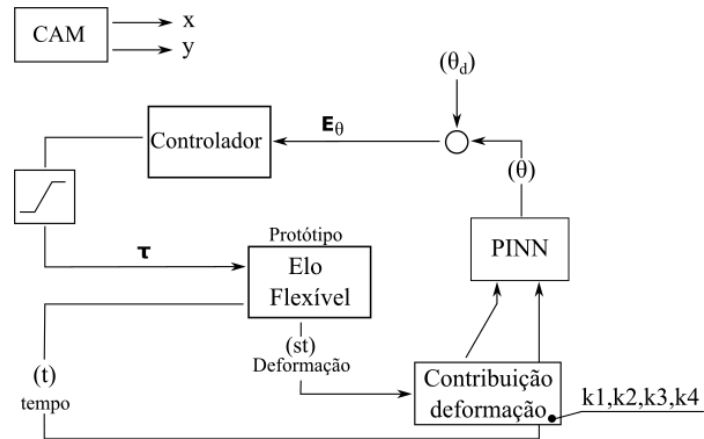
Essa configuração permitiu que a rede aproximasse a deformação por funções harmônicas de senos e cossenos, filtrando ruídos de alta frequência, mantendo apenas os componentes relevantes da dinâmica. Outro benefício da utilização da FNN foi a possibilidade de calcular derivadas de primeira e segunda ordem com maior facilidade, evitando a amplificação de ruídos típica da diferenciação numérica.

3.5.4 Validação experimental

Para realizar uma avaliação experimental do modelo obtido, ele foi empregado no controle do manipulador flexível de elo único. A esquemática da aplicação experimental é ilustrada na Figura 16. O PINN utiliza o vetor do tempo (t) e a deformação (st), multiplicadas respectivamente pelos seus coeficientes (conforme visto na Eq. 15), como entradas para treinar a rede neural. Em seguida, os pesos e os vieses aprendidos são utilizados para a

reconstrução do modelo que foi utilizado para realizar o controle do sistema. Neste caso, o método proposto é implementado no protótipo a fim de avaliar como o método responde ao elo flexível e a sua eficácia na tarefa.

Figura 16 – Ilustração esquemática para a aplicação experimental



Fonte: Autoria própria, 2025

4 RESULTADOS

Nessa seção, serão apresentados os resultados dos testes nos diferentes modelos estudados, ela foi estruturada em duas partes. A primeira trata dos pêndulos simulados, incluindo o pêndulo rígido e o pêndulo flexível. A segunda aborda os sistemas reais do manipulador flexível serial e manipulador 3RRR. Também foi realizada uma descrição das limitações e dificuldades encontradas na obtenção dos resultados.

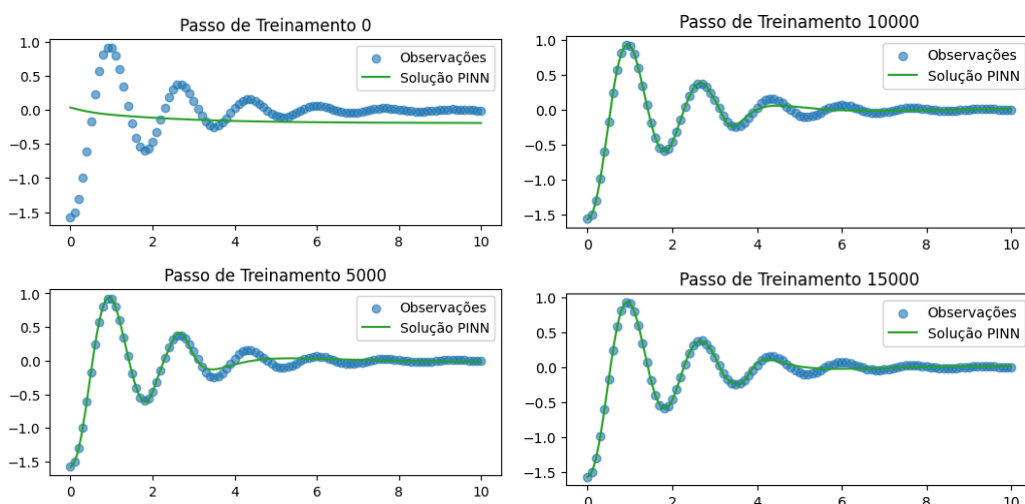
4.1 Pêndulos simulados

Os pêndulos simulados serviram de base para a validação e familiarização inicial dos métodos de aprendizado de máquina. Os resultados obtidos permitiram avaliar as diferenças entre o comportamento de um sistema rígido e de um sistema flexível, realidade mais próxima do sistema experimental com elos flexíveis.

4.1.1 Pêndulo rígido

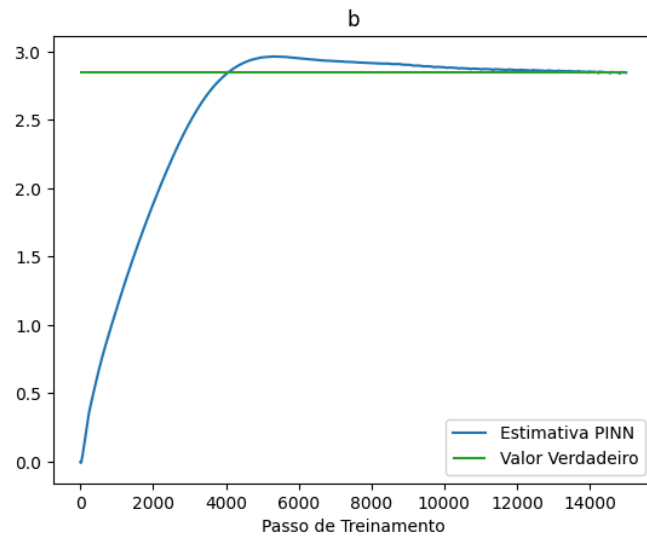
Na simulação do pêndulo rígido, o elo foi modelado como um corpo indeformável preso a uma junta de revolução. As Figuras 17 e 18 ilustram os resultados obtidos.

Figura 17 – Evolução da solução do PINN do pêndulo rígido



Fonte: Autoria própria, 2025

Figura 18 – Evolução da estimativa do amortecimento do PINN do pêndulo rígido



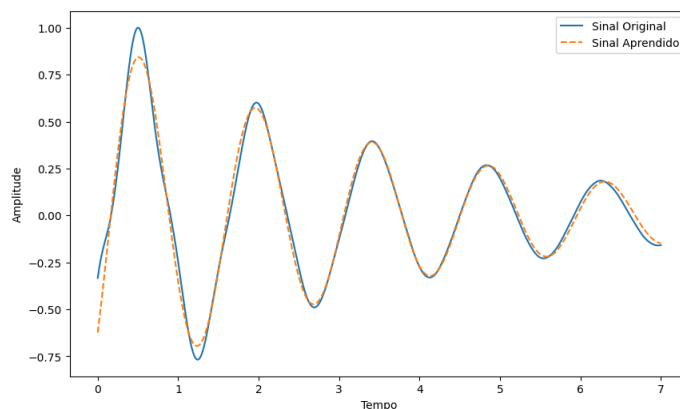
Fonte: Autoria própria, 2025

Conforme mostrado nas Figuras 17 e 18, percebe-se que a rede neural é capaz de reconhecer padrões complexos, através do comportamento do sistema mecânico. Essa primeira abordagem garante uma validação da viabilidade da estratégia empregada nesta etapa da pesquisa e permitirá uma avaliação do impacto da flexibilidade no sistema.

4.1.2 Pêndulo flexível

Na simulação do pêndulo flexível, foi utilizado o recurso ViewFlex do MSC Adams, que integra elementos finitos ao corpo multicorpos para representar deformações elásticas. Este caso também utilizou a FNN para aproximar a deformação por uma série de Fourier, observável na Figura 19.

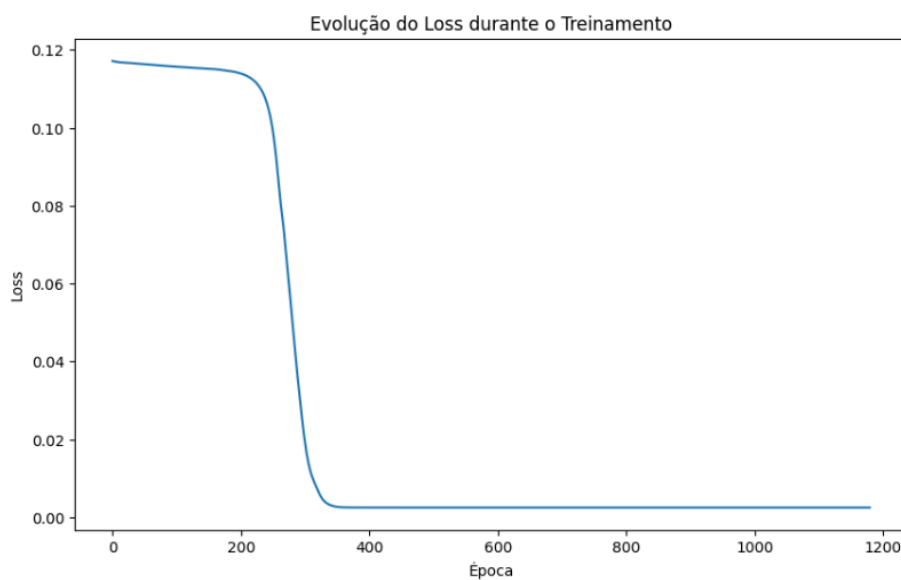
Figura 19 – Medidas de deformação e aproximação pela rede neural de Fourier



Fonte: Autoria própria, 2025

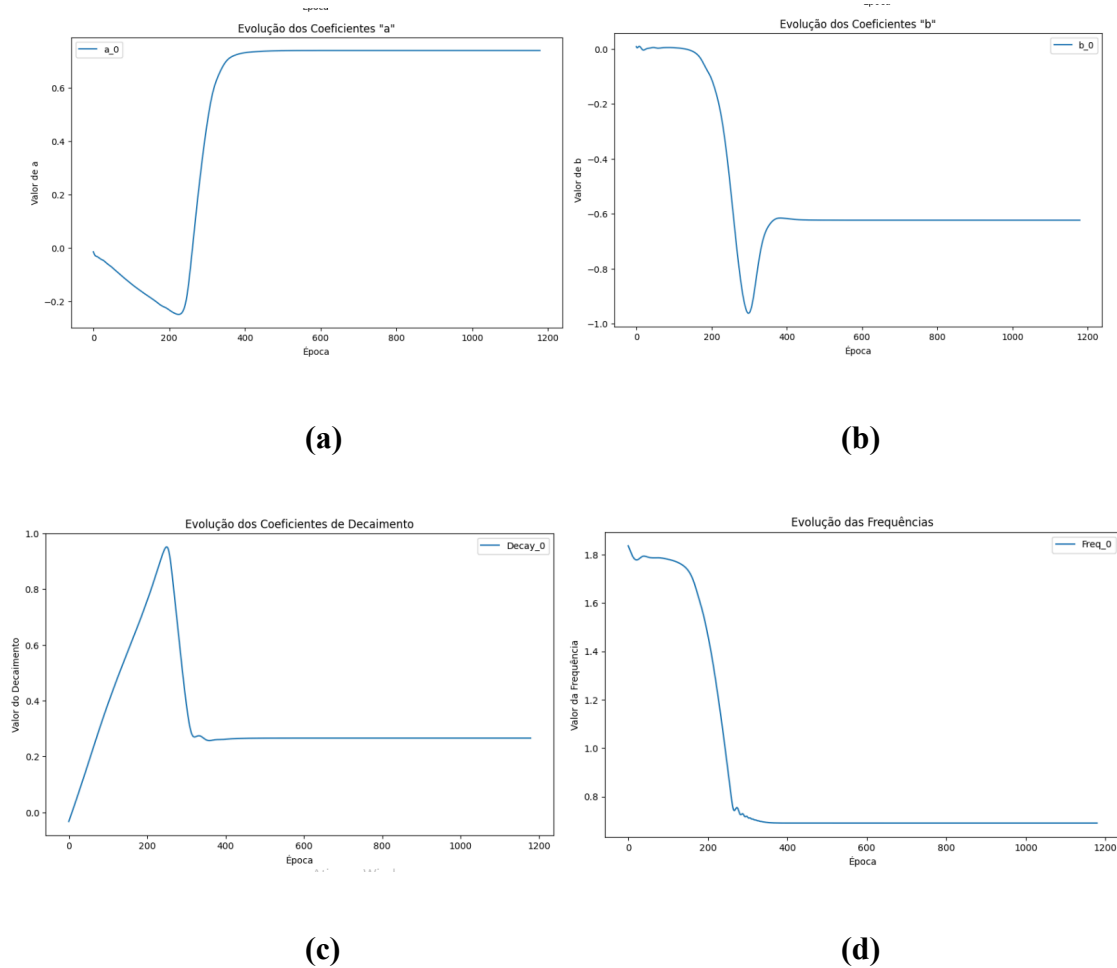
Analisando as Figuras 20 e 21, podemos notar que a função de perda e todos os coeficientes da série de Fourier convergiram de forma adequada, indicando que o treinamento ocorreu com sucesso e que a aproximação obtida é precisa.

Figura 20 – Evolução da função de perda da FNN



Fonte: Autoria própria, 2025

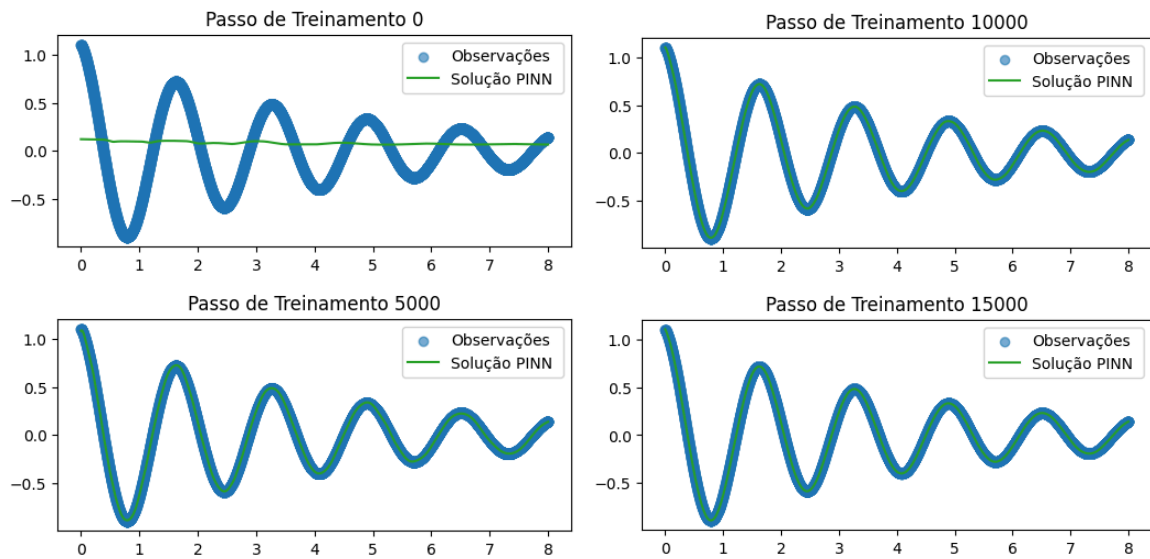
Figura 21 – Evolução dos parâmetros aprendidos pela FNN (a) Evolução do coeficiente a (b) Evolução do coeficiente b (c) Evolução do coeficiente de decaimento (d) Evolução da frequência



Fonte: Autoria própria, 2025

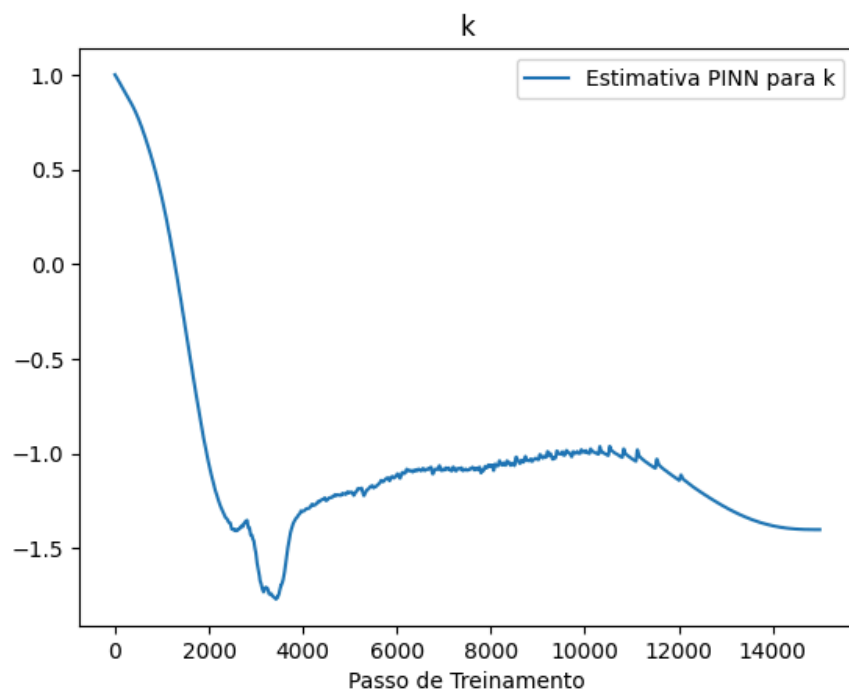
Ao observar a Figura 22, nota-se que a solução obtida segue bem as medidas observadas. No entanto, as Figuras 23 e 24 mostram que tanto o termo “k” quanto as funções de perda não convergiram adequadamente. Com isso, através do princípio da técnica de aprendizado de máquina SINDy, busca-se avaliar as contribuições em que a flexibilidade acarreta no sistema dinâmico. Por isso, no modelo do manipulador flexível serial, utilizaram-se múltiplos coeficientes com termos distintos, a fim de melhorar o processo de treinamento de rede neural e melhorar a qualidade do modelo obtido.

Figura 22 – Evolução da solução do PINN do pêndulo flexível



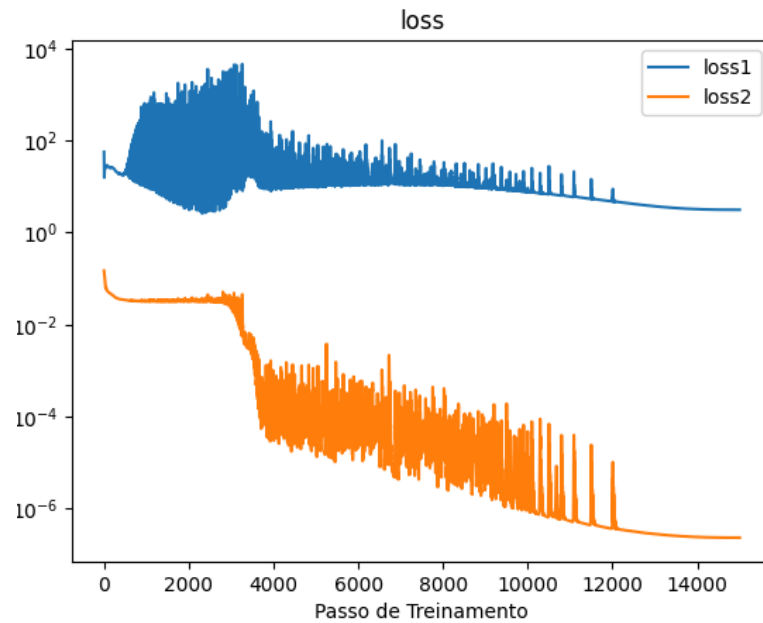
Fonte: Autoria própria, 2025

Figura 23 – Evolução da estimativa do termo “k” do PINN do pêndulo flexível



Fonte: Autoria própria, 2025

Figura 24 – Evolução das funções de perda do PINN do pêndulo flexível



Fonte: Autoria própria, 2025

4.2 Manipuladores robóticos

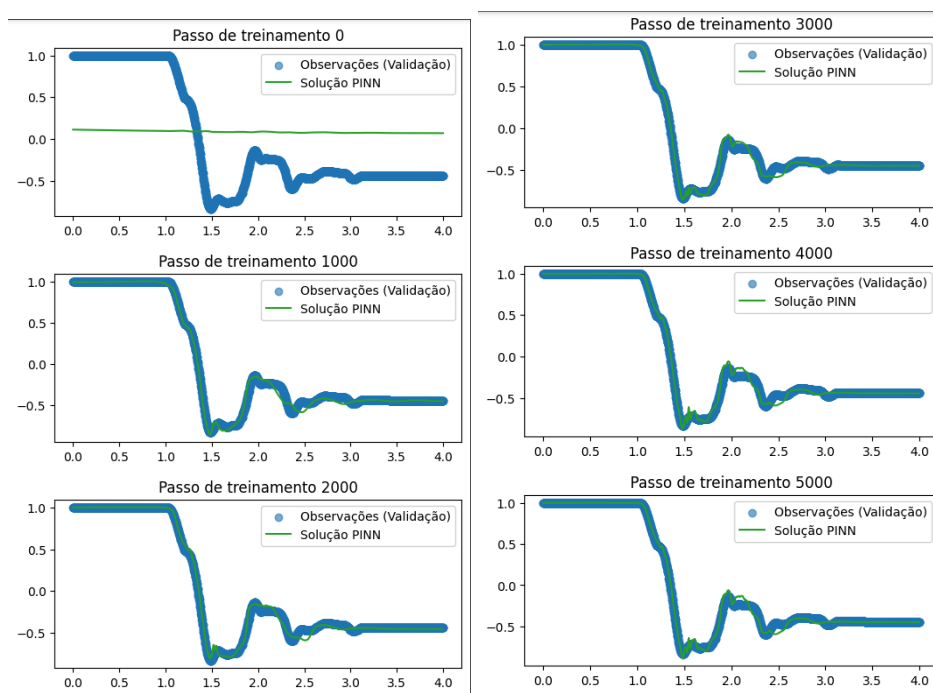
Após a validação inicial por meio das simulações de pêndulos, passou-se à análise dos manipuladores reais. Essa etapa serviu como o teste final da eficácia das redes neurais em sistemas complexos e condições experimentais.

4.2.1 Manipulador flexível serial

O manipulador flexível serial utilizou a ideia de parâmetros esparsos, inspirada no método SINDy, para definir os termos k_1 , k_2 , k_3 e k_4 da Equação 15, que foi empregada como função de perda física da rede neural. A saída deste PINN é o ângulo θ da extremidade livre do manipulador, calculado pelas coordenadas x e y do efetuador, incluindo a contribuição dos sistemas rígido e flexível. Dez trajetórias foram coletadas do protótipo, a fim de construir a base de treinamento e validação. Nesse modelo, foram utilizados nove conjuntos de dados experimentais para o treinamento da rede e um décimo conjunto, não incluído no processo de aprendizagem, foi reservado exclusivamente para a etapa de validação.

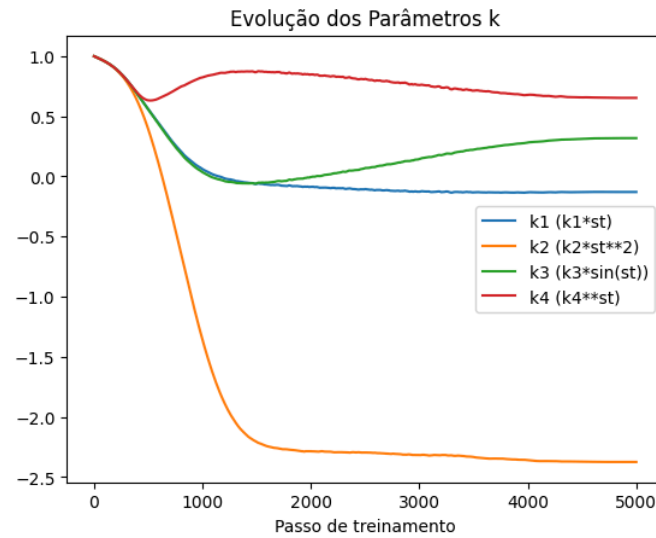
Diferente do caso do pêndulo flexível, a inserção de múltiplos coeficientes para a obtenção da contribuição do sinal do extensômetro, o procedimento adotado apresenta boa aderência ao conjunto de dados de validação, conforme pode ser observado na Figura 25. As Figuras 26 e 27 indicam que tanto os coeficientes “k” quanto às funções de perda convergiram adequadamente. Note que em cada coeficiente adotado, há uma contribuição distinta, o que destaca como a flexibilidade pode estar relacionada à física do sistema.

Figura 25 – Evolução da solução do PINN do manipulador flexível serial



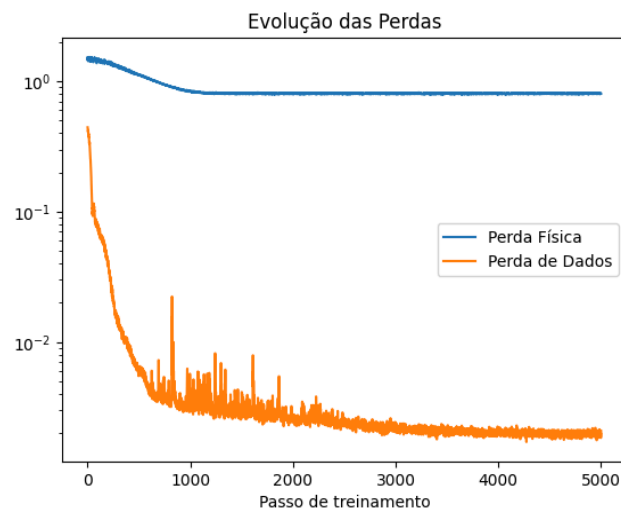
Fonte: Autoria própria, 2025

Figura 26 – Evolução das estimativas dos termos “k” do PINN do manipulador flexível serial



Fonte: Autoria própria, 2025

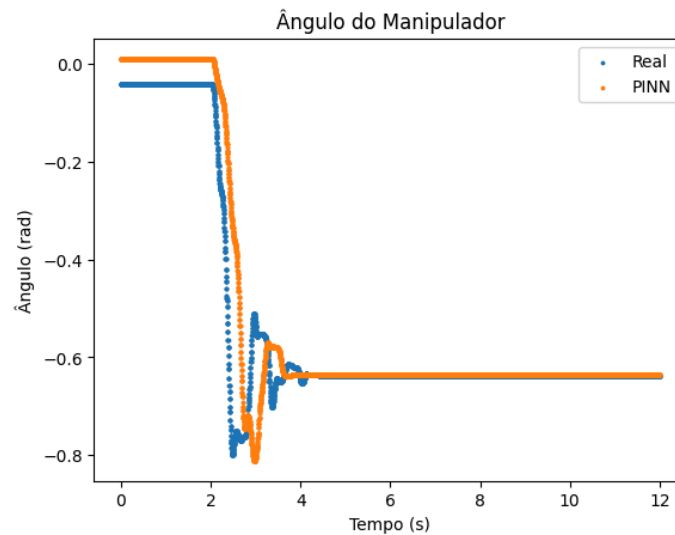
Figura 27 – Evolução das funções de perda do PINN do manipulador flexível serial



Fonte: Autoria própria, 2025

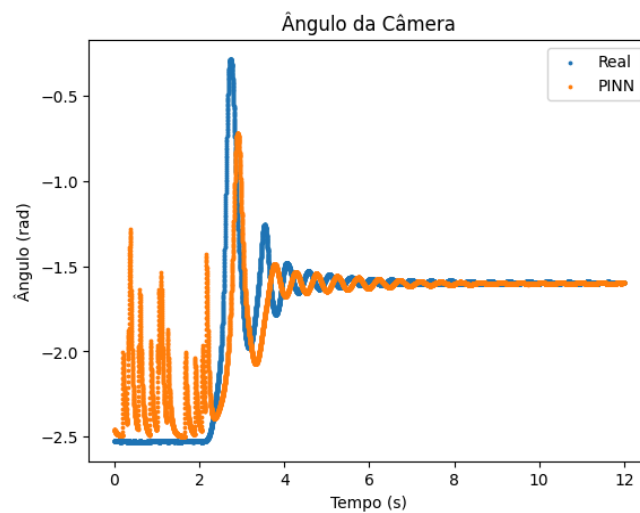
Como última abordagem, foi realizada uma validação experimental do modelo obtido, as Figura 28 e 29 mostram a comparação teórica e experimental da metodologia proposta. Note que neste caso, o sinal teórico é dito por “Real”, enquanto o sinal experimental é descrito como “PINN”.

Figura 28 – Comparação entre as posições reais e o controle do ângulo medido diretamente do manipulador do modelo do PINN



Fonte: Autoria própria, 2025

Figura 29 – Comparação entre as posições reais e o controle do ângulo calculado a partir das posições x e y da câmera do modelo do PINN



Fonte: Autoria própria, 2025

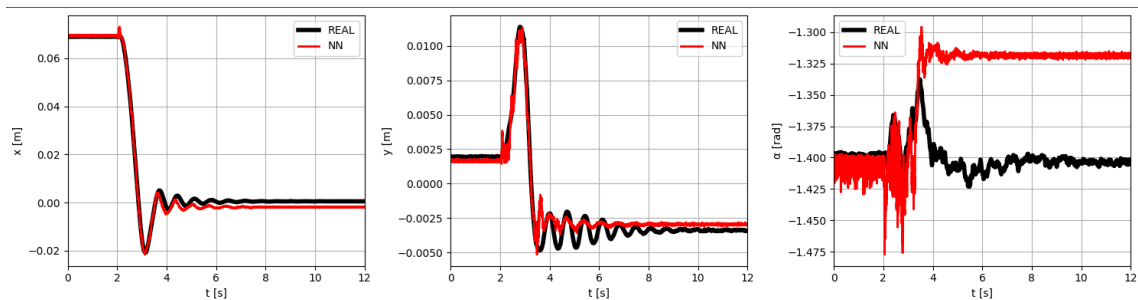
Conforme visto nas Figuras 28 e 29, a técnica do PINN apresenta um desempenho positivo, na aplicação experimental. Através da Figura 29 se observa oscilações, ocasionadas pela câmera, no período estacionário do elo, não possuindo qualquer interferência no sinal de medida. Portanto, vê-se que a abordagem adotada pode ser utilizada em aplicações práticas em controle de sistemas flexíveis. Através da utilização da metodologia empregada

anteriormente, busca-se avaliar a aplicação das técnicas de aprendizagem no manipulador paralelo 3RRR, conforme visto na seção abaixo.

4.2.2 Manipulador 3RRR

Para o manipulador 3RRR, utilizou-se uma rede neural do tipo *feedforward*, cuja entrada é composta pelas três posições angulares dos motores e pelos sinais obtidos dos extensômetros instalados nos elos. A Figura 30 apresenta a comparação entre os dados reais obtidos experimentalmente e os valores estimados pela rede neural para as variáveis x , y e α do efetuador final. Observa-se que a rede proposta foi capaz de capturar, de maneira satisfatória, a dinâmica não linear do manipulador paralelo 3RRR com elos flexíveis.

Figura 30 – Comparação entre as posições reais e estimadas pela rede neural



Fonte: Autoria própria, 2025

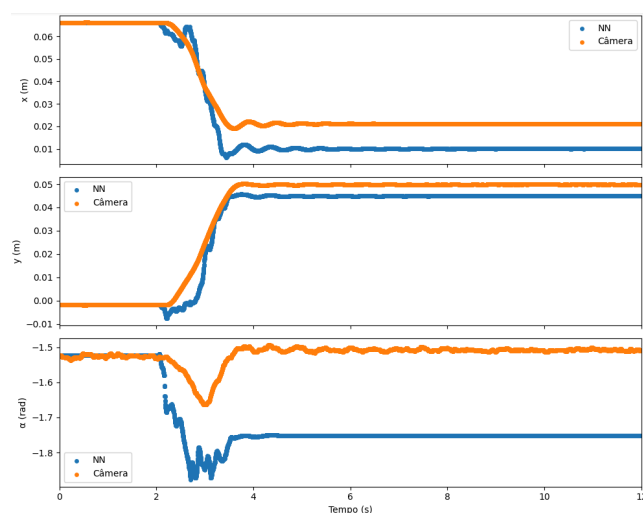
A rede neural demonstrou bom desempenho na estimativa das componentes de posição do manipulador, com destaque para as coordenadas x e y , cujas respostas acompanharam de forma bem próxima os dados reais. As principais discrepâncias ocorreram nas fases de transição e amortecimento, o que era esperado, dada a complexidade das oscilações resultantes da flexibilidade estrutural. Por outro lado, a variável angular apresentou maior sensibilidade a erros, especialmente na resposta inicial, devido ao acoplamento entre a orientação e as deformações dos elos, além da influência de ruídos nos sensores. Ainda assim, a rede foi capaz de reproduzir coerentemente a tendência geral da resposta dinâmica.

De modo geral, a arquitetura *feedforward* com múltiplas camadas ocultas, combinada com a minimização baseada no MSE e com o uso de funções de ativação ReLU, mostrou-se eficaz no mapeamento não linear entre os sinais de entrada e a resposta do sistema. Os resultados obtidos indicam que a rede neural desenvolvida possui potencial para modelar

sistemas com significativa flexibilidade estrutural e acoplamento dinâmico, como o manipulador 3RRR.

Como forma de abordar experimentalmente o método proposto da rede neural *feedforward*, também foi realizada uma abordagem similar ao de controle (para o manipulador serial) utilizando o modelo obtido, conforme ilustrado na Figura 31. Nesse experimento, a fim de avaliar a validação do técnica proposta, as posições x , y e o ângulo α estimados pelo modelo NN foram comparados com os valores medidos pela câmera.

Figura 31 – Comparação entre as posições da câmera e estimadas pela rede neural do 3RRR



Fonte: Autoria própria, 2025

Através da comparação, pode-se notar que o controle realizado conseguiu utilizar da técnica de aprendizado de máquinas para estimar a dinâmica do manipulador paralelo. Devido à complexidade e graus de liberdade adicionais, em comparação ao serial, houve uma discrepância um pouco mais relevante. No entanto, o modelo proposto foi capaz de capturar satisfatoriamente o comportamento do sistema mecânico, o que valida a estratégia proposta e abre espaço para futuras discussões de melhorias.

4.3 Limitações e desafios enfrentados

Durante o desenvolvimento da pesquisa, algumas limitações e desafios foram identificados tanto na etapa experimental quanto na implementação computacional dos

modelos de aprendizado de máquina. Os principais pontos enfrentados e os ajustes realizados para contorná-los serão descritos nesta seção.

4.3.1 Limitações dos experimentos

Os experimentos realizados com o elo flexível e com o manipulador 3RRR apresentaram algumas restrições práticas devido a motivos de segurança e limitações físicas. A principal delas foi em relação à limitação dos testes a pequenas amplitudes e baixas velocidades, restringindo a variabilidade dos dados e reduzindo a capacidade da rede de ser generalizada para condições dinâmicas mais complexas. Fora isso, o uso de extensômetros em elos flexíveis dos manipuladores apresentou ruído devido à sensibilidade do sensor, o que dificultou a obtenção de dados limpos para treinamento e validação das redes neurais.

4.3.2 Desafios na implementação dos algoritmos de aprendizado de máquina

A implementação dos algoritmos de aprendizado de máquina foi a área em que ocorreram os maiores desafios, a maioria deles foram atribuídos à complexidade desse tipo de algoritmo. Existem muitos parâmetros para serem controlados e técnicas diferentes que podem ser utilizadas, isso faz com que o ajuste das redes neurais seja difícil e muitas vezes não intuitivo. Em alguns casos, a rede neural apresentou dificuldade em convergir de forma estável, colocando em dúvida a confiabilidade do modelo obtido. O balanceamento das funções de perda dos PINNs também foi complexo, pois pesos inadequados levavam a sobreajuste ou perda da consistência física. No caso do pêndulo flexível, a função da perda física precisou ter sua influência reduzida drasticamente para alcançar convergência, indicando possíveis inadequações no modelo.

Além destas dificuldades, esse tipo de algoritmo possui alto custo computacional, devido à necessidade de calcular derivadas automáticas e termos da equação diferencial a cada iteração. O que restringiu o número de experimentos de calibração de hiperparâmetros e impôs simplificações nas equações dinâmicas utilizadas. Embora o Google Colab tenha fornecido acesso a GPU, o tempo disponível de execução e a memória limitaram o número de testes em larga escala. O cálculo das derivadas da deformação foi outro desafio, pois a diferenciação direta dos sinais experimentais amplificava ruídos, comprometendo a qualidade dos dados utilizados.

4.3.3 Ajustes realizados

Em relação às restrições de movimento do manipulador, o experimento para coleta de dados foi repetido diversas vezes com diferentes trajetórias a fim de estabelecer um banco de dados seguro. Os sinais ruidosos foram filtrados, normalizados e aproximados por uma FNN, possibilitando calcular derivadas de forma analítica e reduzir os efeitos do ruído. Sobre a implementação dos algoritmos, utilizamos várias técnicas para melhorar a qualidade do treinamento e modelos obtidos, como a política de aprendizado One Cycle, que foi implementada a fim de melhorar a convergência e acelerar o treinamento. Também foram realizados diversos testes com os pesos das funções de perda com o objetivo de equilibrar a influência dos dados e da física no treinamento das PINNs. A validação cruzada e experimental foram outros métodos muito importantes para avaliar o desempenho dos modelos em diferentes conjuntos de dados, garantir sua robustez e comprovar a sua eficácia.

Esses ajustes permitiram que os modelos de aprendizado de máquina atingissem desempenho satisfatório, mesmo diante das limitações experimentais e computacionais enfrentadas.

5 CONCLUSÃO

O presente trabalho investigou a aplicação de redes neurais artificiais e aprendizado de máquina informado por física na modelagem de manipuladores com elos flexíveis, utilizando tanto simulações virtuais quanto dados experimentais obtidos em bancada. A pesquisa contemplou desde modelos simplificados até a aplicação em um sistema paralelo complexo, permitindo avaliar a robustez e a aplicabilidade da abordagem.

5.1 Principais conclusões

Baseado nos resultados obtidos, foi possível concluir que os métodos de aprendizado de máquina demonstraram alto potencial na modelagem de sistemas mecânicos com dinâmicas complexas, como os manipuladores robóticos. As redes implementadas foram eficazes tanto em configurações simples, como os pêndulos simulados, quanto em sistemas reais de maior complexidade, como o manipulador paralelo 3RRR com elos flexíveis. Esse desempenho, em conjunto com a validação experimental, reforça a robustez das técnicas aplicadas e evidencia que o aprendizado de máquina pode capturar fenômenos não lineares e de acoplamento dinâmico típicos desses sistemas, tornando-se uma alternativa sólida aos métodos analíticos tradicionais.

No caso específico do manipulador de elo único, a estrutura baseada em PINN, com perda física inspirada no método SINDy, possibilitou a identificação bem-sucedida de parâmetros físicos a partir de dados reais, como amortecimento e flexibilidade. Já para o manipulador 3RRR completo, a rede *feedforward* mostrou um desempenho satisfatório na estimativa das variáveis de saída, reproduzindo o comportamento real mesmo sob ruído experimental. Esses resultados consolidam o PIML como uma ferramenta promissora para a modelagem de manipuladores robóticos flexíveis, estabelecendo uma base teórica e experimental sólida para o avanço de futuras pesquisas em modelagem e controle em tempo real.

5.2 Aspectos positivos

Entre os principais aspectos positivos, temos a validação em diferentes sistemas, essa metodologia se mostrou eficaz para aumentar gradativamente a complexidade dos modelos estudados, permitindo um aprendizado e aplicação mais efetiva das técnicas utilizadas. A integração entre dados e modelos físicos do PIML é outro ponto de destaque, pois ele foi capaz de conciliar informações experimentais com equações diferenciais, levando a um

aprendizado estável, que não depende de grandes bases de dados, e modelos mais consistentes e simples. O uso das FNN também foi notável por explorar um uso diferente das redes neurais, além de sua aproximação ser muito eficaz na melhoria da representação da deformação, possibilitando calcular derivadas analíticas mais precisas, mitigando os efeitos de ruídos experimentais. O uso da estratégia de identificação de parâmetros físicos inspirada no método SINDy foi muito benéfico ao trazer interpretabilidade ao modelo, aproximando-o de representações físicas. A sobreposição entre as trajetórias experimentais e as previsões da rede mostraram a aplicação prática e uma validação da eficiência dos modelos obtidos, consolidando o PIML como uma ferramenta promissora para aplicações em robótica e controle de sistemas flexíveis.

Finalmente, o projeto possibilitou o domínio de diversas ferramentas e áreas como MSC Adams, MATLAB, bibliotecas de aprendizado de máquina em Python e os experimentos em bancada, ampliando a base interdisciplinar da pesquisa.

5.3 Reflexões sobre limitações

Apesar dos resultados promissores, algumas limitações apontam para oportunidades de aprimoramento em pesquisas futuras, a principal delas foi a exploração de arquiteturas alternativas. A área de aprendizado de máquina é muito vasta e existem outros métodos a serem explorados que podem oferecer vantagens adicionais no desempenho em relação às redes neurais utilizadas. O ajuste dos parâmetros do algoritmo também é outro campo que pode ser investigado com mais profundidade, a configuração atual é satisfatória, mas certamente existem áreas que podem ser aprimoradas a fim de melhorar o desempenho do treinamento. A capacidade computacional foi outro fator limitante deste trabalho, uma vez que o uso do Google Colab impôs restrições de tempo de execução e memória. Essas limitações impactaram o treinamento das redes neurais, pois apresentavam um custo computacional elevado e restringiram o número de testes realizados. Por fim, um número maior de ensaios com trajetórias maiores e variadas enriqueceria a base de dados e beneficiaria a generalização dos modelos obtidos.

5.4 Perspectiva para melhorias e aplicações futuras

Durante a execução deste trabalho, ocorreram desafios tanto na etapa experimental quanto na implementação computacional dos modelos de aprendizado de máquina. As

limitações identificadas ao longo do processo influenciaram os resultados, mas também trouxeram aprendizados importantes e indicaram caminhos promissores para aprimoramentos futuros da pesquisa na aplicação de aprendizado de máquina em manipuladores robóticos.

Baseado nas reflexões das limitações na seção anterior, as perspectivas de aprimoramento envolvem principalmente o avanço das arquiteturas de rede, a otimização dos processos de treinamento e o fortalecimento da infraestrutura experimental e computacional. A estudo de arquiteturas alternativas, como as redes convolucionais, redes recorrentes e o uso de autoencoders, pode ampliar a capacidade de generalização e reduzir o custo de convergência das redes. Além disso, técnicas de ajuste automático de hiperparâmetros e métodos de regularização mais sofisticados poderiam contribuir para um aprendizado mais estável e eficiente.

Outro ponto fundamental é a ampliação da base experimental, que foi limitada por a restrições práticas dos manipuladores reais. Deste modo, o uso de sensores com maior precisão e uma revisão da implementação do modelo permitiria que ensaios com maior amplitude e velocidade sejam executados com segurança. Juntando isso com testes de diferentes condições de carga proporcionaria uma representação mais ampla do comportamento dinâmico do sistema, favorecendo a robustez e a generalização dos modelos obtidos. Em paralelo, a adoção de uma infraestrutura computacional mais potente, com o uso de GPUs dedicadas ou servidores de alto desempenho, superaria as limitações impostas pelo ambiente de desenvolvimento utilizado. Possibilitando explorar redes mais complexas, realizar múltiplos experimentos de forma simultânea, comparar o desempenho de outras funções de ativação e perda.

Ademais, visamos aplicar o PINN na modelagem do manipulador paralelo 3RRR com elos flexíveis completo e realizar o seu controle em tempo real, este era um dos objetivos iniciais deste trabalho, porém, a complexidade da tarefa superou as expectativas e não foi possível realizá-la no escopo do projeto. No entanto, com a base de conhecimento estabelecida, essa tarefa se torna possível e a sua realização consolidaria o PIML como uma ferramenta prática e confiável para o controle de sistemas robóticos flexíveis. Por fim, espera-se que este trabalho contribua não apenas para a compreensão dos manipuladores flexíveis, mas também para o avanço das aplicações de aprendizado de máquina informado por física na engenharia.

REFERÊNCIAS

- BASTOS, G. A non-inherent parametric estimation for dynamical equivalence of flexible manipulators. **Optimal Control Applications and Methods**, v. 43, n. 3, p. 825–841, 2022.
- BENNEHAR, M.; CHEMORI, A.; BOURI, M.; JENNI, L.; PIERROT, F. A new RISE-based adaptive control of PKMs: design, stability analysis and experiments. **International Journal of Control**, v. 91, n. 3, p. 593–607, 2017.
- BENGOA, P.; ZUBIZARRETA, A.; CABANES, I.; MANCISIDOR, A.; PINTO, C.; MATA, S. Virtual sensor for kinematic estimation of flexible links in parallel robots. **Sensors**, v. 17, n. 9, 1934, 2017.
- BONVENTI JÚNIOR, W.; ARANHA, N. Estudo das oscilações amortecidas de um pêndulo físico com o auxílio do “Tracker”. **Revista Brasileira de Ensino de Física**, v. 37, n. 2, p. 2504-1–2504-9, 2015.
- CARVALHO FONTES, J. V. de; COLOMBO, F. T.; SILVA, N. B. F. da; SILVA, M. M. da. Model-based joint and task space control strategies for a kinematically redundant parallel manipulator. **Robotica**, v. 40, n. 5, p. 1570–1586, 2021.
- CHEN, Q.; ZI, B.; SUN, Z.; LI, Y.; XU, Q. Design and development of a new cable-driven parallel robot for waist rehabilitation. **IEEE/ASME Transactions on Mechatronics**, v. 24, n. 4, p. 1497–1507, 2019.
- CHOI, R.; COYNER, A.; KALPATHY-CRAMER, J.; CHIANG, M.; CAMPBELL, J. Introduction to machine learning, neural networks, and deep learning. **Translational Vision Science & Technology**, v. 9, n. 2, p. 14, 2020.
- COLOMBO, F. T.; CARVALHO FONTES, J. V. de; SILVA, M. M. da. A visual servoing strategy under limited frame rates for planar parallel kinematic machines. **Journal of Intelligent & Robotic Systems**, v. 96, n. 1, p. 95–107, 2019.
- COLOMBO, F. T.; SILVA, M. M. da. Deriving suitable models for model-based control design of a parallel manipulator with flexible components. **Journal of Vibration and Control**, 2022.
- DE SILVA, B. M.; et al. PySINDy: A Python package for the sparse identification of nonlinear dynamical systems from data. **Journal of Open Source Software**, v. 5, n. 49, p. 2104, 2020.
- ESMAEILI, H.; RIZVI, R. Machine learning predictions and benchmarking of non-linear mechanical behavior of polymer composites. **Behavior and Mechanics of Multifunctional Materials XVII**. Proceedings of SPIE, v. 12484, p. 124840I, 18 abr. 2023.
- FAN, D.; et al. A robotic intelligent towing tank for learning complex fluid-structure dynamics. **Science Robotics**, v. 4, e.aay5063, 2019.
- GERONEL, R. S.; OLIVEIRA, G. C. de; SILVA, M. M. da. Dual-loop control strategies for vibration attenuation of a flexible parallel manipulator. **Journal of Vibration and Control**, p. 1–13, 2024.

GREIS, N. P.; NOGUEIRA, M. L.; BHATTACHARYA, S.; et al. Stability modeling for chatter avoidance in self-aware machining: an application of physics-guided machine learning. **Journal of Intelligent Manufacturing**, v. 34, p. 387–413, 2023.

KARNIADAKIS, G. E.; KEVREKIDIS, I. G.; LU, L.; PERDIKARIS, P.; WANG, S.; YANG, L. Physics-informed machine learning. **Nature Reviews Physics**, v. 3, p. 422–440, 2021.

LING, M.; YUAN, L.; ZHANG, X. Geometrically nonlinear analysis of compliant mechanisms using a dynamic beam constraint model (DBCM). **Mechanism and Machine Theory**, v. 191, 105489, 2024.

LIPINSKI, K.; BOBROWSKI, K.; WITTBRODT, E. Rigid finite elements and multibody modeling in analyses of a robot shaped elastic/plastic deformations of a beam. **Advances in Mechanism and Machine Science**. Cham: Springer, 2019. p. 2731–2740.

MORLOCK, M.; BURKHARDT, M.; SEIFRIED, R.; EBERHARD, P. End-effector trajectory tracking of flexible link parallel robots using servo constraints. **Multibody System Dynamics**, v. 56, n. 1, p. 1–28, 2022.

MORLOCK, M.; MEYER, N.; PICK, M.-A.; SEIFRIED, R. Real-time trajectory tracking control of a parallel robot with flexible links. **Mechanism and Machine Theory**, v. 158, 104220, 2021.

MOHAN, S.; MOHANTA, J. K.; HUESING, M.; CORVES, B. Dual-loop motion control for geometric errors and joint clearances compensation of a planar 2-PRP+1-PPR manipulator. **Mechanisms, Transmissions and Applications**. Cham: Springer, 2017. p. 171–180.

NICOLAOU, Z. G.; HUO, G.; CHEN, Y.; BRUNTON, S. L.; KUTZ, J. N. Data-driven discovery and extrapolation of parameterized pattern-forming dynamics. **Physical Review Research**, v. 5, n. 4, p. L042017, nov. 2023.

PACCOT, F.; ANDREFF, N.; MARTINET, P. A review on the dynamic control of parallel kinematic machines: theory and experiments. **International Journal of Robotics Research**, v. 28, n. 3, p. 395–416, 2009.

PHAM, M.-N.; CHAMPLIAUD, H.; LIU, Z.; BONEV, I. A. Parameterized finite element modeling and experimental modal testing for vibration analysis of an industrial hexapod for machining. **Mechanism and Machine Theory**, v. 167, 104502, 2022.

REN, K.; CHEW, Y.; ZHANG, Y. F.; FUH, J. Y. H.; BI, G. J. Thermal field prediction for laser scanning paths in laser-aided additive manufacturing by physics-based machine learning. **Computer Methods in Applied Mechanics and Engineering**, v. 362, 112734, 2020.

RUIZ, A. G.; SANTOS, J. C.; CROES, J.; DESMET, W.; SILVA, M. M. da. On redundancy resolution and energy consumption of kinematically redundant planar parallel manipulators. **Robotica**, v. 36, n. 6, p. 809–821, 2018.

SAIED, H.; CHEMORI, A.; RAFEI, M. E.; FRANCIS, C.; PIERROT, F. From non-model-based to model-based control of PKMs: a comparative study. **Mechanism, Machine, Robotics and Mechatronics Sciences**. Cham: Springer, 2018. p. 153–169.

SCHMIDHUBER, J. Deep learning in neural networks: an overview. **Neural Networks**, v. 61, p. 85–117, 2015.

SILVA, M. M. da; OLIVEIRA, L. P. de; BRÜLS, O.; MICHELIN, M.; BARADAT, C.; TEMPIER, O.; CAIGNY, J. D.; SWEVERS, J.; DESMET, W.; BRUSSEL, H. V. Integrating structural and input design of a 2-DOF high-speed parallel manipulator: a flexible model-based approach. **Mechanism and Machine Theory**, v. 45, n. 11, p. 1509–1519, 2010.

SILVA, M. M. da; COLOMBO, F. T.; OLIVEIRA, G. C. de; OLIVEIRA, L. P. R. de. Hybrid vision/strain-based control strategy for a parallel manipulator with flexible links. **Mechanism and Machine Theory**, v. 191, 105508, 2024.

SUBRAMANIAN, A.; MAHADEVAN, S. Probabilistic physics-informed machine learning for dynamic systems. **Reliability Engineering & System Safety**, v. 230, 108899, 2023.

VAREDI-KOULAEI, S. M.; DANIALI, H. M.; FARAJTABAR, M. Reducing the undesirable effects of joints clearance on the behavior of the planar 3-RRR parallel manipulators. **Nonlinear Dynamics**, v. 86, p. 1007–1022, 2016.

WAREY, A.; KAUSHIK, S.; HAN, T. Physics-guided sparse identification of nonlinear dynamics for prediction of vehicle cabin occupant thermal comfort. **SAE Technical Paper**, n. 2022-01-0159, 2022.

WANG, Y.; LIU, L.; YUAN, M.; DI, Q.; CHEN, B.; WU, H. A new model-free robust adaptive control of cable-driven robots. **International Journal of Control, Automation and Systems**, v. 19, n. 9, p. 3209–3222, 2021.

WANG, Y.; ZHU, K.; CHEN, B.; JIN, M. Model-free continuous nonsingular fast terminal sliding mode control for cable-driven manipulators. **ISA Transactions**, v. 98, p. 483–495, 2020.

WILLARD, J.; JIA, X.; XU, S.; STEINBACH, M.; KUMAR, V. Integrating physics-based modeling with machine learning: a survey. **ACM Computing Surveys**, v. 55, n. 1, p. 1–34, 2022.

YU, Y.; YAO, H.; LIU, Y. Structural dynamics simulation using a novel physics-guided machine learning method. **Engineering Applications of Artificial Intelligence**, v. 96, 103947, 2020.

APÊNDICE

A seguir está o repositório no Google Drive que reúne os códigos desenvolvidos no Google Colab utilizados para o processamento dos dados, construção e comparação dos modelos de aprendizado de máquina empregados neste projeto. Também estão incluídos os conjuntos de dados utilizados, os modelos criados no software MSC Adams e os artigos e projetos relacionados ao trabalho.

https://drive.google.com/drive/folders/1z2eHkfA-_Npqmw8BEI9Xn6JZ24r4VKsz?usp=sharing