

Universidade Federal de Santa Catarina
Centro Tecnológico de Ciências Exatas e Educação
Departamento de Engenharia de Controle, Automação e
Computação



Daniel Victor Krepsky

Desenvolvimento de software para síntese de geração de função
em mecanismos de seis barras via otimização

Blumenau

2025

Daniel Victor Krepsky

**Desenvolvimento de software para síntese de geração
de função em mecanismos de seis barras via
otimização**

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como parte dos requisitos necessários para a obtenção do Título de Engenheiro de Controle e Automação.

Orientador: Prof. Dr. Leonardo Mejia Rincon

Coorientador: Prof. Dr. Daniel Martins

Universidade Federal de Santa Catarina
Centro Tecnológico de Ciências Exatas e Educação
Departamento de Engenharia de Controle, Automação e Computação

Blumenau
2025

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Krepsky, Daniel Victor

Desenvolvimento de software para síntese de geração de função em mecanismos de seis barras via otimização / Daniel Victor Krepsky ; orientador, Leonardo Mejia Rincon, coorientador, Daniel Martins, 2025.

97 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Campus Blumenau, Graduação em Engenharia de Controle e Automação, Blumenau, 2025.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Síntese de mecanismos. 3. Geração de função. 4. Mecanismos de seis barras. 5. Otimização. I. Rincon, Leonardo Mejia. II. Martins, Daniel. III. Universidade Federal de Santa Catarina. Graduação em Engenharia de Controle e Automação. IV. Título.

Daniel Victor Krepsky

Desenvolvimento de software para síntese de geração de função em mecanismos de seis barras via otimização

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Engenheiro de Controle e Automação.

Comissão Examinadora

Prof. Dr. Leonardo Mejia Rincon
Universidade Federal de Santa Catarina
Orientador

Prof. Dr. Ebrahim Samer El Youssef
Universidade Federal de Santa Catarina

Prof. Dr. Esdras Salgado da Silva
Universidade Federal de Santa Catarina

Blumenau, 17 de dezembro de 2025

Dedico este trabalho aos meus pais e à minha noiva, que estiveram presentes nesta etapa tão importante da minha vida e que, com certeza, estarão nas demais que virão.

Agradecimentos

Agradeço primeiramente aos meus pais, Rosane e Raphael, que me proporcionaram amor, carinho e educação ao longo de toda a minha vida, moldando com orgulho a pessoa que sou hoje.

Agradeço à minha noiva, Giordana, que conheci logo no início da graduação, há quatro anos, pelo companheirismo, pela paciência, pelo amor e por todo o apoio nos momentos em que mais precisei. O estresse do dia a dia, as entregas desafiadoras, os longos períodos de estudo e as dificuldades profissionais e pessoais tornaram-se mais leves, graças à sua companhia e incentivo.

Agradeço ao meu professor e orientador, Leonardo Mejia, que sempre dedicou parte de seu tempo para ajudar, ensinar e compartilhar sua vasta sabedoria sobre robótica e mecanismos, despertando em mim uma verdadeira paixão por essas áreas do conhecimento.

Por fim, mas não menos importante, agradeço aos amigos que conheci ao longo desta jornada acadêmica, que estiveram ao meu lado desde o início, apoiando e motivando uns aos outros.

*“Nobody ever figures out what life is all about, and it doesn’t matter. Explore the world.
Nearly everything is really interesting if you go into it deeply enough.”*
(Richard P. Feynman)

Resumo

Este trabalho apresenta o desenvolvimento de um programa computacional voltado à síntese de geração de função de mecanismos de seis barras, abrangendo todas as suas topologias clássicas. A proposta visa simplificar o processo de projeto e ampliar as possibilidades de aplicação de mecanismos complexos por meio de otimização numérica baseada no algoritmo de Evolução Diferencial (DE).

Inicialmente, foram revisados os fundamentos da teoria de mecanismos, destacando a análise e síntese cinemática, bem como conceitos de algoritmos evolutivos aplicados à engenharia mecânica. Em seguida, implementou-se o programa em linguagem Python, empregando programação orientada a objetos e uma interface gráfica de usuário (GUI) que permite inserir parâmetros, visualizar resultados e exportar dados de forma intuitiva.

Os estudos de caso contemplaram a síntese de mecanismos capazes de reproduzir funções matemáticas, como logarítmica e séries de Fourier, além de trajetórias articulares humanas aplicadas à modelagem de uma prótese de mão. Os resultados demonstraram a eficiência do método proposto, com baixo erro médio e boa estabilidade geométrica, confirmando a aplicabilidade do programa como ferramenta de apoio a pesquisas e projetos na área de Engenharia.

Palavras-Chave: 1. Síntese de mecanismos. 2. Geração de função. 3. Mecanismos de seis barras. 4. Otimização. 5. Interface Gráfica de Usuário

Abstract

This work presents the development of a computational program aimed at the function generation synthesis of six-bar mechanisms, encompassing all their classical topologies. The proposal seeks to simplify the design process and expand the applicability of complex mechanisms through numerical optimization based on the Differential Evolution (DE) algorithm.

Initially, the fundamentals of mechanism theory were reviewed, highlighting kinematic analysis and synthesis, as well as evolutionary algorithms applied to mechanical engineering. Subsequently, the program was implemented in the Python language, employing object-oriented programming and a graphical user interface (GUI) that enables users to input parameters, visualize results, and export data intuitively.

The case studies included the synthesis of mechanisms capable of reproducing mathematical functions, such as logarithmic and Fourier-series functions, in addition to human joint trajectories applied to the modeling of a prosthetic hand. The results demonstrated the efficiency of the proposed method, with low average error and good geometric stability, confirming the applicability of the program as a support tool for research and design in the field of Engineering.

Keywords: 1. Mechanism synthesis. 2. Function generation. 3. Six-bar mechanisms. 4. Optimization. 5. Graphical User Interface

Lista de figuras

Figura 1 – Mecanismo de Svoboda	17
Figura 2 – Mecanismo de quatro barras de um alicate de pressão	21
Figura 3 – Pares cinemáticos inferiores	22
Figura 4 – Pares cinemáticos superiores	23
Figura 5 – Classificação de cadeias cinemáticas	24
Figura 6 – Representações de um mecanismo de 4 barras	25
Figura 7 – Exemplo de uma porta com motor no eixo de rotação	25
Figura 8 – Mecanismo com dois circuitos	26
Figura 9 – Mecanismo de 4 barras	28
Figura 10 – Exemplos reais de mecanismos de 4 barras	29
Figura 11 – Mecanismos de 6 barras	29
Figura 12 – Geração de movimento de um Watt I	30
Figura 13 – Exemplos reais de mecanismos 6 barras	30
Figura 14 – Mecanismo que calcula $\log_{10} x$	34
Figura 15 – Montagens do mecanismo Watt II	37
Figura 16 – Defeitos de Circuito (a), Ramificação (b) e Ordem (c) em mecanismos .	38
Figura 17 – Exemplo de função multimodal	39
Figura 18 – Fluxograma de estruturação do capítulo	44
Figura 19 – Fluxograma de etapa funcional	47
Figura 20 – Tela principal da Interface	53
Figura 21 – Tela de configuração	55
Figura 22 – UML do programa	57
Figura 23 – Tela inicial com dados inseridos	58
Figura 24 – Primeira tela de gráficos	58
Figura 25 – Segunda tela de gráficos	59
Figura 26 – Tabela com os dados exportados	59
Figura 27 – Gráfico da função logarítmica 4.1	61
Figura 28 – Gráfico da função dada pela série de fourier 4.2	61
Figura 29 – Gráfico da função log com os pontos de precisão	63
Figura 30 – Watt I para 5 pontos de precisão da função \ln 4.1	63
Figura 31 – Watt II para 5 pontos de precisão da função logarítmica mostrada na Eq. 4.1	65
Figura 32 – Stephenson I para 5 pontos de precisão da função \ln 4.1	65
Figura 33 – Stephenson II para 5 pontos de precisão da função \ln 4.1	65
Figura 34 – Stephenson III para 5 pontos de precisão da função \ln 4.1	66

Figura 35 – Gráfico da função de Série de Fourier com os pontos de precisão	67
Figura 36 – Watt I para 20 pontos de precisão da função da série de Fourier 4.2 . .	68
Figura 37 – Watt II para 20 pontos de precisão da função da série de Fourier 4.2 .	68
Figura 38 – Stephenson I para 20 pontos de precisão da função da série de Fourier 4.2	68
Figura 39 – Stephenson II para 20 pontos de precisão da função da série de Fourier 4.2	69
Figura 40 – Stephenson III para 20 pontos de precisão da função da série de Fourier 4.2	69
Figura 41 – Curvas das articulações	73
Figura 42 – Mecanismo Watt II para articulação do quadril	74
Figura 43 – Mecanismo Stephenson III para articulação do joelho	74
Figura 44 – Mecanismo Stephenson II para articulação do tornozelo	74
Figura 45 – Projeto simplificado de uma perna humanóide com mecanismos	75
Figura 46 – Trajetória simulada da ponta do dedo obtida pela síntese de função. . .	78
Figura 47 – Protótipo final da prótese de mão modelado em <i>SolidWorks</i>	79
Figura 48 – Protótipo final da prótese de mão segurando uma lixa	80

Lista de tabelas

Tabela 1 – Resumo visual dos mecanismos e suas equações principais	33
Tabela 2 – Coeficientes da série de Fourier de $f(\theta_{in})$	62
Tabela 3 – Ângulos de entrada e saída da otimização	63
Tabela 4 – Ângulos de entrada, ângulos de saída desejados, obtidos e erro absoluto (em graus)	64
Tabela 5 – Parâmetros geométricos otimizados e tempo de otimização para a função ln	64
Tabela 6 – Ângulos de entrada e saída da otimização com 20 pontos	66
Tabela 7 – Erro absoluto médio do ângulo de saída (em graus)	67
Tabela 8 – Parâmetros geométricos otimizados e tempo para a função da série de Fourier	67
Tabela 9 – Coeficientes de Fourier utilizados por Plecnik e McCarthy (2016) para a síntese das funções articulares	72
Tabela 10 – Parâmetros geométricos otimizados e tempo para as curvas das articulações	72
Tabela 11 – Pares de ângulos de entrada e saída utilizados na síntese de função. . .	77
Tabela 12 – Limites geométricos adotados na otimização.	77
Tabela 13 – Parâmetros geométricos otimizados para o dedo indicador.	78

Lista de Siglas e Abreviaturas

CROS	<i>Brazilian Conference on Robotics</i>
DE	<i>Differential Evolution</i>
DH	<i>Denavit-Hartenberg</i>
DoF	<i>Degree of Freedom</i>
EMG	<i>Eletromiográfico</i>
GDL	<i>Grau de Liberdade</i>
GUI	<i>Graphical User Interface</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IFTOMM	<i>International Federation for the Promotion of Mechanism and Machine Science</i>
POO	<i>Programação Orientada a Objetos</i>
UFSC	<i>Universidade Federal de Santa Catarina</i>

Lista de Símbolos

x, y	<i>Coordenadas cartesianas</i>
M	<i>Mobilidade do mecanismo</i>
λ	<i>Ordem do espaço de trabalho</i>
f_i	<i>Conectividade do i-ésimo par cinemático (graus de liberdade permitidos pelo par)</i>
c_i	<i>Restrição imposta pelo i-ésimo par cinemático (número de restrições)</i>
j	<i>Índice (ou contador) usado para referir um par/junta específico</i>
\bar{j}	<i>Número total de pares cinemáticos considerados na soma da Eq. de mobilidade</i>
n	<i>Número de elos (barras) do mecanismo</i>
L_1	<i>Comprimento do elo 1</i>
L_2	<i>Comprimento do elo 2</i>
L_3	<i>Comprimento do elo 3</i>
L_4	<i>Comprimento do elo 4</i>
L_5	<i>Comprimento do elo 5</i>
L_6	<i>Comprimento do elo 6</i>
L_8	<i>Comprimento do elo 8</i>
L_9	<i>Comprimento do elo 9</i>
θ_I	<i>Ângulo de entrada</i>
θ_O	<i>Ângulo de saída</i>
θ_{Od}	<i>Ângulo de saída desejado</i>
μ_1	<i>Primeiro ângulo de qualidade de transmissão</i>
μ_2	<i>Segundo ângulo de qualidade de transmissão</i>
$f(\theta_I)$	<i>Função desejada para síntese</i>
a_0, a_i	<i>Coefficientes da série de Fourier</i>
b_i	<i>Coefficientes da série de Fourier</i>
F_{OBJ}	<i>Valor da função objetivo</i>
p_j	<i>Indivíduo da população</i>
p'_j	<i>Indivíduo mutado/recombinado</i>
lb, rb	<i>Limites inferior e superior</i>

Sumário

1	INTRODUÇÃO	16
1.1	Contextualização	16
1.2	Objetivos	18
1.2.1	Objetivo Geral	18
1.2.2	Objetivos Específicos	18
1.2.3	Estrutura do Trabalho	19
2	REVISÃO DE LITERATURA	20
2.1	Teoria de Mecanismos	20
2.1.1	Mobilidade	25
2.1.2	Mecanismos Planares com 1 GDL	27
2.1.2.1	Mecanismo de 4 barras	28
2.1.2.2	Mecanismos de 6 barras	28
2.1.3	Análise Cinemática de Mecanismos de 6 barras	30
2.1.4	Síntese de Mecanismos	32
2.1.4.1	Síntese de Geração de Função	34
2.1.4.2	Metodologias para Síntese de Mecanismos	36
2.1.5	Configurações de Montagem de Mecanismos	36
2.1.6	Defeitos de Síntese de Mecanismos	37
2.2	Algoritmos de Otimização	37
2.2.1	Algoritmo Evolução Diferencial	41
2.2.2	Aplicações da Evolução Diferencial em Mecanismos	42
3	METODOLOGIA	44
3.1	Visão Geral da Solução	45
3.2	Requisitos do Projeto	45
3.3	Solução Proposta	47
3.3.1	Requisitos Funcionais	47
3.3.1.1	Inserção dos Parâmetros de Entrada	47
3.3.1.2	Modelagem Cinemática e Formulação da Função Objetivo	48
3.3.1.3	Aplicação das Restrições de Qualidade de Transmissão	49
3.3.1.4	Execução do Algoritmo de Otimização	50
3.3.1.5	Condição de Parada e Exibição dos Resultados	50
3.3.1.6	Configurações do algoritmo	51
3.3.2	Requisitos Não Funcionais e de Interface	51

3.3.2.1	Definição da Interface Gráfica de Usuário (GUI)	52
3.3.3	Integração da GUI com a Aplicação Funcional	56
3.3.4	Funcionamento Completo	56
4	RESULTADOS	60
4.1	Testes e Validação	60
4.1.1	Teste da Função Logarítmica	62
4.1.2	Teste da Função de Série de Fourier	66
4.1.3	Discussão dos Testes	69
4.2	Estudos de Caso	70
4.2.1	Síntese de Mecanismos para Juntas de Quadril, Joelho e Tor- nozelo Humano	71
4.2.2	Síntese de Mecanismos para uma Prótese de Mão	76
5	CONCLUSÕES	81
	REFERÊNCIAS BIBLIOGRÁFICAS	82
	APÊNDICES	86
	APÊNDICE A – MODELO WATT I	87
	APÊNDICE B – MODELO WATT II	89
	APÊNDICE C – MODELO STEPHENSON I	91
	APÊNDICE D – MODELO STEPHENSON II	93
	APÊNDICE E – MODELO STEPHENSON III	95

1 Introdução

1.1 Contextualização

A construção e o aperfeiçoamento de máquinas e mecanismos têm desempenhado um papel essencial no desenvolvimento das civilizações ao longo da história. Desde as soluções engenhosas criadas por povos antigos, como egípcios e chineses, aplicadas na agricultura, no transporte e na construção de grandes estruturas, até os avanços consolidados durante a Revolução Industrial e a era tecnológica atual, o uso de mecanismos foi determinante para o progresso humano. Ao possibilitar a execução de tarefas complexas com maior eficiência, reduzir o esforço físico e ampliar a capacidade produtiva, máquinas e mecanismos não apenas simplificaram atividades cotidianas, mas também sustentaram transformações científicas, econômicas e sociais que moldaram a sociedade moderna. (MARTINS; MURAI, 2019).

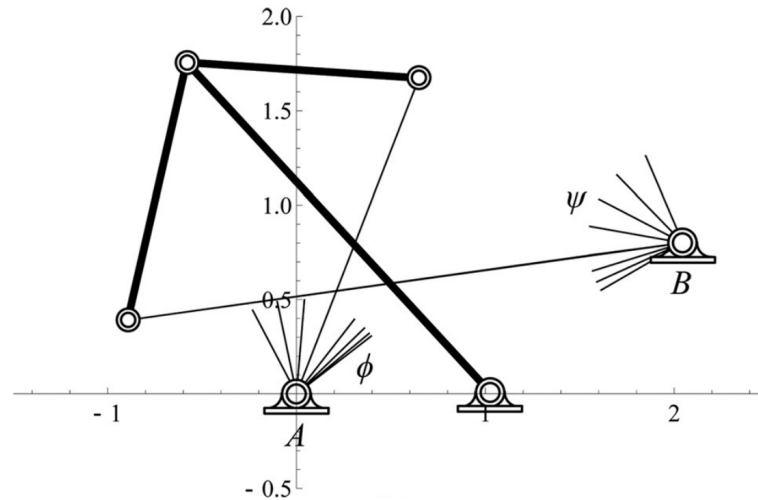
Dentro desse contexto, ganha destaque a síntese de mecanismos para geração de função, cujo objetivo é projetar arranjos mecânicos capazes de estabelecer uma relação funcional entre o movimento de entrada e o movimento de saída. Diferentemente da síntese de trajetória ou de movimento, a geração de função busca atender a requisitos matemáticos que definem a correspondência entre variáveis cinemáticas, garantindo que o mecanismo desempenhe corretamente a função desejada. (ERDMAN; SANDOR, 1984).

Essa síntese foi amplamente empregada no século XX no projeto de mecanismos conhecidos como computadores mecânicos (PLECNIK; MCCARTHY, 2014), capazes de realizar cálculos para determinar distâncias, ângulos, valores de funções e outros parâmetros. Como exemplo de aplicação, destaca-se o mecanismo logarítmico de Svoboda, utilizado para o cálculo da função $\log x$, apresentado na Figura 1.

Atualmente, a síntese continua sendo empregada nas indústrias automobilística e aeroespacial devido às suas características adimensionais, isto é, à capacidade de o mecanismo manter as relações entre ângulos de entrada e de saída mesmo quando escalado, o que facilita o desenvolvimento de projetos sujeitos a restrições de espaço que variam ao longo do tempo. Um exemplo de aplicação são os mecanismos de contrabalanceamento de carga, utilizados em porta-malas de veículos e em portas de aeronaves, com o objetivo de reduzir o esforço exigido do usuário.

Para mecanismos de quatro barras, existem soluções analíticas para a síntese de geração de função com 3, 4 e 5 pares de precisão, sendo abordada a de 3 pares em (FREUDENSTEIN, 1954) e as de 4 e 5 pares em (ERDMAN; SANDOR, 1984) e (HARTENBERG; DENAVIT, 1964). Entretanto, para mecanismos de seis barras, que apresentam complexidade significativamente maior, as soluções analíticas tornam-se mais extensas,

Figura 1 – Mecanismo de Svoboda



Fonte: (PLECNIK; MCCARTHY, 2014)

tornando necessária a utilização de métodos numéricos, como a homotopia planar (PLECNIK, 2015), que encontra todas as soluções. Contudo, essa abordagem nem sempre é viável devido ao elevado custo computacional e ao tempo exigido para filtrar mecanismos com defeitos.

Portanto, à medida que a complexidade dos mecanismos aumenta e as exigências de desempenho se tornam mais rigorosas, os métodos clássicos de síntese de geração de função se mostram limitados para atender simultaneamente múltiplas restrições e critérios de projeto. Nesse contexto, a síntese de mecanismos por otimização surge como uma alternativa eficaz, pois permite explorar amplos espaços de soluções em busca de um ótimo global, em vez de se dedicar à obtenção de todas as soluções possíveis, o que torna o processo significativamente mais rápido.

Para que essa abordagem seja implementável, o método de otimização precisa ser simples de implementar, fácil de usar, confiável e capaz de convergir de forma robusta para o ótimo verdadeiro, sem exigir tempo computacional excessivo. O Differential Evolution (DE) (PRICE; STORN; LAMPINEN, 2006) exemplifica essas características, sendo um algoritmo de otimização global desenvolvido em 1995 que conquistou reputação por sua eficácia e robustez. Embora não seja uma solução universal, o DE demonstra ser uma ferramenta poderosa para engenheiros que buscam otimizar projetos complexos, como os de mecanismos, de maneira eficiente.

De modo geral, este trabalho apresenta o desenvolvimento de um programa para síntese de geração de função de mecanismos de seis barras por meio de técnicas de otimização, com o objetivo de simplificar o processo de projeto e aumentar sua eficiência. Ao longo do estudo, serão abordados conceitos fundamentais de mecanismos e algoritmos, definidas estratégias de análise e otimização, implementada uma interface em Python utilizando programação orientada a objetos e conduzidos estudos de caso para validar o desempe-

nho do programa. Estabelecendo-se, dessa forma, uma base sólida para demonstrar a relevância e a aplicabilidade da ferramenta desenvolvida.

1.2 Objetivos

Com base na contextualização abordada anteriormente, a seguir serão descritos os objetivos deste trabalho.

1.2.1 Objetivo Geral

Desenvolver um programa computacional voltado à síntese de geração de função para todas as topologias de mecanismos de seis barras, utilizando métodos de otimização aplicados a N pares de ângulos de precisão, com o propósito de simplificar o processo de projeto e auxiliar no desenvolvimento de mecanismos mais eficientes.

1.2.2 Objetivos Específicos

Com o intuito de alcançar o objetivo principal deste trabalho, faz-se necessário cumprir os seguintes objetivos específicos:

- Realizar uma pesquisa bibliográfica que permita compreender e sistematizar os principais conceitos relacionados a mecanismos e a algoritmos de otimização.
- Definir a estratégia mais adequada de análise de posição de mecanismos e selecionar o algoritmo de otimização a ser adotado.
- Desenvolver e implementar os modelos cinemáticos necessários para a análise de posição dos mecanismos de seis barras, garantindo a consistência matemática requerida para o processo de síntese.
- Projetar e implementar uma interface gráfica em Python que aprimore a usabilidade do programa por usuários externos, incluindo recursos de visualização gráfica do processo construtivo e exportação de dados para planilhas.
- Empregar programação orientada a objetos no desenvolvimento da interface, de modo a assegurar escalabilidade e facilitar adaptações para novos mecanismos.
- Aplicar uma metodologia de síntese de mecanismos na condução de estudos de caso, com o objetivo de verificar a validade e a aplicabilidade do programa desenvolvido.

1.2.3 Estrutura do Trabalho

Esta monografia está organizada em cinco capítulos. O Capítulo 1 corresponde à Introdução, na qual são apresentados a contextualização do tema, a formulação do problema, os objetivos geral e específicos, além da estrutura do trabalho.

O Capítulo 2 apresenta a Revisão de Literatura, reunindo os conceitos fundamentais relacionados à teoria de mecanismos, à análise e síntese cinemática, às metodologias de síntese, bem como a discussão de algoritmos de otimização aplicados e de trabalhos correlatos.

No Capítulo 3, são descritos os requisitos necessários para o desenvolvimento do programa, abrangendo os procedimentos adotados, as técnicas empregadas e a implementação proposta, desde as funcionalidades internas até a criação da interface.

O Capítulo 4 apresenta os Resultados, incluindo a análise dos estudos de caso e a validação da abordagem implementada.

Por fim, o Capítulo 5 reúne as Conclusões, destacando as principais contribuições da pesquisa, suas limitações e as possibilidades de continuidade em trabalhos futuros.

2 Revisão de Literatura

O presente capítulo tem como objetivo apresentar os fundamentos que dão suporte a este trabalho, indo além de simples definições e fórmulas. Ele conduz o leitor por uma jornada estruturada em dois eixos complementares: (i) a teoria de mecanismos, abordando mobilidade, cadeias cinemáticas, a análise/síntese de mecanismos de quatro e seis barras, entre outros, essenciais para compreender como movimentos simples podem se transformar em movimentos complexos; e (ii) os algoritmos de otimização, com destaque para abordagens estocásticas e, em especial, a Evolução Diferencial, cuja simplicidade e eficiência a tornam especialmente adequada para problemas de síntese dimensional. Dessa forma, este capítulo não apenas situa o leitor no estado da arte, mas também evidencia o espaço onde a contribuição deste trabalho se insere, mostrando como conceitos teóricos se conectam à prática e à busca por soluções eficientes.

2.1 Teoria de Mecanismos

Quando falamos em **mecanismo**, a primeira imagem que surge pode ser a de uma engrenagem girando ou de uma alavanca em movimento (HUNT, 1990). De fato, mecanismos são conjuntos de partes rígidas, chamadas elos (ou barras), conectadas entre si por juntas que permitem movimento. Apesar de sua aparência simples, esses arranjos escondem uma função poderosa: transformar um tipo de movimento em outro, com propósito específico. Formalmente, um mecanismo pode ser definido como um dispositivo mecânico projetado para transferir movimento e/ou força de uma origem até uma saída específica (ERDMAN; SANDOR, 1984). Para construí-lo, é necessário combinar os elos de maneira adequada, vinculando-os entre si e a um referencial fixo, de modo que o conjunto funcione harmoniosamente.

Embora a definição formal possa parecer abstrata, é possível perceber o conceito de transferir movimento e/ou força de maneira intuitiva em ferramentas do dia a dia. Um exemplo clássico é o alicate de pressão, projetado para multiplicar a força aplicada pelo usuário, permitindo prender objetos com maior facilidade. Por trás dessa simplicidade aparente, o alicate oculta um mecanismo de quatro barras (Figura 2): ao apertar as extremidades mais distantes das alças, a força transmitida à boca do alicate é significativamente ampliada, devido à vantagem mecânica proporcionada pela disposição dos elos e das juntas. Para compreender e projetar mecanismos como o do alicate, é necessário primeiro formalizar os conceitos fundamentais que regem o funcionamento desses sistemas, os quais serão apresentados a seguir.

As obras e referências utilizadas neste trabalho para a apresentação e explicação dos

Figura 2 – Mecanismo de quatro barras de um alicate de pressão



Fonte: <http://www.mrbillington.com/linkages.html>

conceitos relacionados a mecanismos adotam definições que estão em conformidade com a terminologia estabelecida pelo dicionário da *International Federation for the Promotion of Mechanism and Machine Science* (IFToMM). Assim, garante-se que as descrições conceituais empregadas estejam alinhadas aos padrões internacionais da área.

Para conectar elos entre si e formar um mecanismo, é necessário juntá-los de alguma forma. Esse vínculo é feito por meio do **acoplamento**, que consiste em uma conexão entre dois elos, restringindo os movimentos possíveis de um em relação ao outro e impondo restrições mecânicas (MARTINS; MURAI, 2019). O acoplamento pode ocorrer de forma **direta**, com os dois elos conectados por um elemento de ligação, ou de forma **indireta**, quando a conexão entre eles se dá por meio de um ou mais elos intermediários.

O modo como dois elos se conectam, restringindo seus movimentos relativos, é definido como um **par cinemático**. Em algumas literaturas, par cinemático e **junta** são descritos como equivalentes, porém, o dicionário IFToMM define a junta como a manifestação física de um par cinemático.

Dependendo da geometria do contato, um par cinemático pode ser classificado como **inferior** ou **superior**. Ele é considerado inferior quando um dos elementos não apenas se ajusta à superfície do outro, mas também o envolve. Já o par superior ocorre quando existe apenas um ponto ou uma linha de contato entre os elementos, sem envolvimento (TSAI, 2000). A descrição dos tipos de pares inferiores e superiores podem ser vistos abaixo. As Figuras 3 e 4 apresentam a representação visual dos diferentes tipos de pares cinemáticos.

Pares Cinemáticos Inferiores:

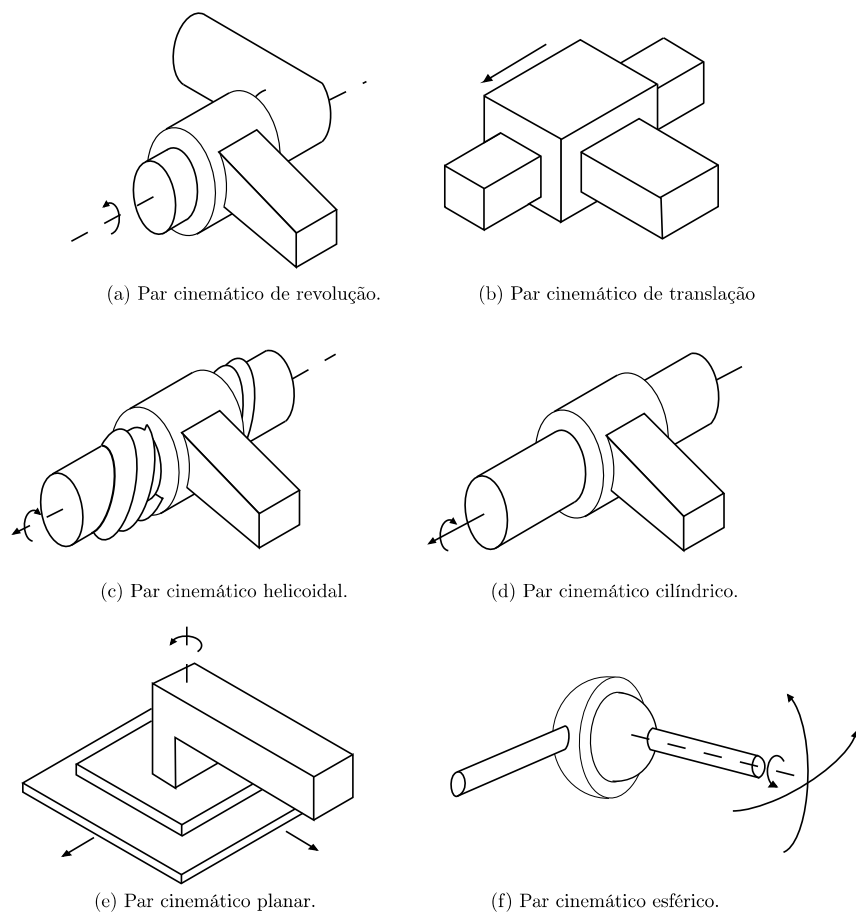
- **Par de Revolução (R):** permite apenas rotação relativa entre dois elos em torno de um eixo fixo.
- **Par Prismático (P):** permite apenas translação relativa ao longo de uma direção.
- **Par Cilíndrico (C):** permite rotação e translação ao longo de um mesmo eixo.

- **Par Esférico (S):** permite rotações em três direções independentes, com centro comum.
- **Par Helicoidal (H):** combinação de rotação e translação acopladas por passo geométrico fixo.
- **Par Plano (E):** permite duas translações no plano e uma rotação perpendicular ao plano.

Pares Cinemáticos Superiores:

- **Par de Contato Pontual:** ocorre quando o contato entre os elos se dá em ponto, como em mecanismos came–seguidor.
- **Par Linear:** estabelecido pelo contato entre dentes de engrenagens, permitindo transmissão rotacional com relação fixa entre velocidades angulares.

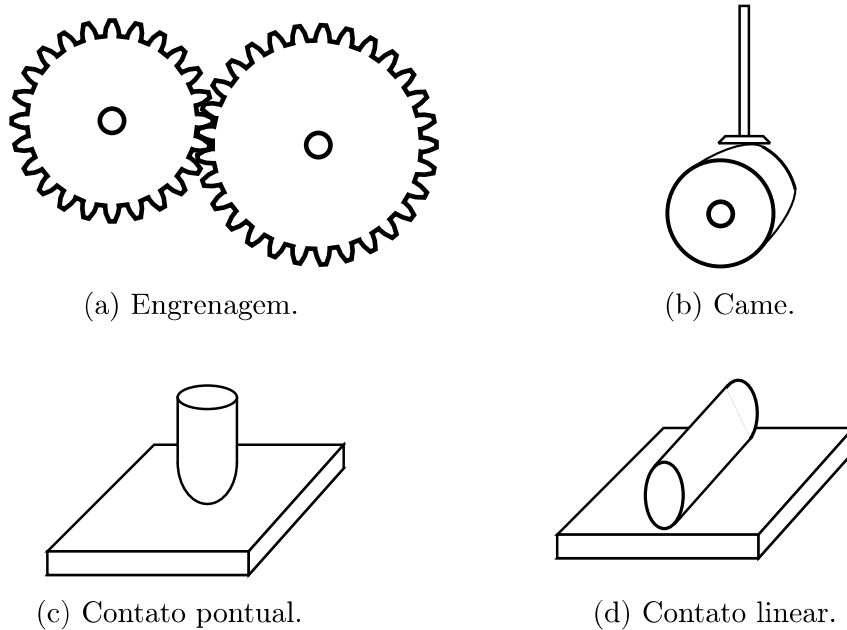
Figura 3 – Pares cinemáticos inferiores



Fonte: Martins e Murai (2019)

A conexão direta entre os elos, obtida por meio de pares cinemáticos, permite a formação de estruturas mais complexas conhecidas como **cadeias cinemáticas**. Uma cadeia

Figura 4 – Pares cinemáticos superiores

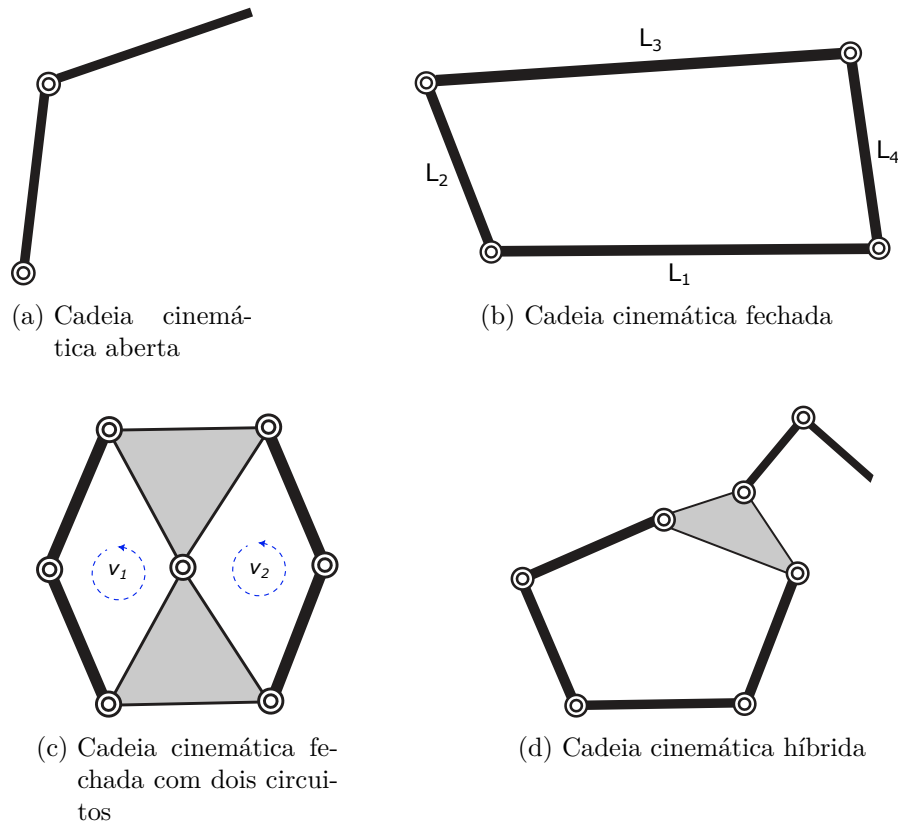


Fonte: Martins e Murai (2019)

cinemática pode ser entendida como um sistema de elos interconectados que possibilita o movimento relativo de qualquer elo em relação aos demais, sendo esse movimento limitado pelas restrições mecânicas do conjunto (RAO; DUKKIPATI, 2007). Segundo Martins e Murai (2019), as cadeias cinemáticas podem ser classificadas em **abertas** (ou seriais), **fechadas** (ou paralelas) e **híbridas** (ou mistas), como ilustrado na Figura 5.

- **Cadeia cinemática aberta:** sistema composto por elos conectados entre si, onde existe apenas um caminho possível para ir de um corpo a outro. Para que a cadeia seja considerada aberta, é necessário que haja pelo menos dois elos que estejam conectados exclusivamente a outro corpo rígido, sem formar um ciclo fechado (TSAI, 1999).
- **Cadeia cinemática fechada:** sistema composto por corpos rígidos conectados entre si, onde há pelo menos dois caminhos distintos para ir de um corpo a outro. Nessa configuração, cada corpo rígido está ligado a pelo menos dois outros corpos rígidos, formando um ciclo fechado (TSAI, 1999). Este tipo de cadeia pode apresentar apenas um circuito, como mostra a Figura 5b, ou mais de um circuito, que pode ser visualizado na cadeia da Figura 5c, que possui dois circuitos v_1 e v_2 .
- **Cadeia cinemática híbrida:** Cadeia cinemática híbrida: sistema composto por corpos rígidos e juntas que apresentam simultaneamente segmentos de cadeia aberta e ao menos um ciclo fechado. (TSAI, 1999). Um exemplo de cadeia cinemática híbrida pode ser visto na Figura 5d

Figura 5 – Classificação de cadeias cinemáticas



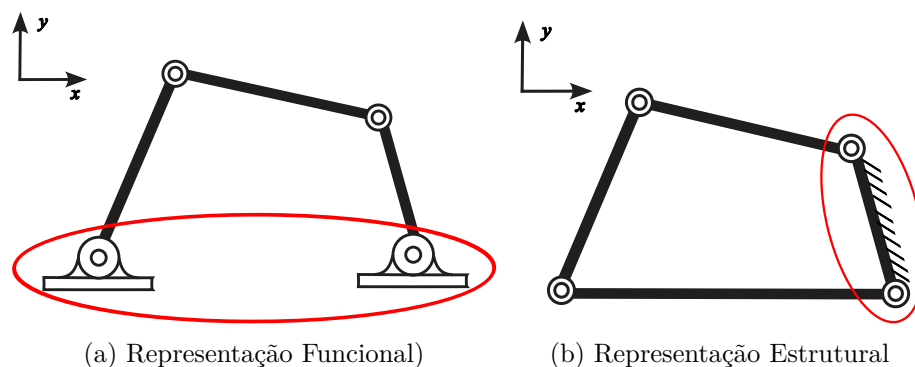
Fonte: Adaptado de Martins e Murai (2019) e Plecnik (2015)

As cadeias cinemáticas fechadas apresentam várias vantagens em relação às abertas. Entre elas, destacam-se: a necessidade de menos atuadores (ou motores), já que nem todas as juntas exigem movimentação independente; a não acumulação de erros de posição e de carga ao longo dos elos, garantindo maior precisão; e a possibilidade de operar com velocidades e acelerações mais elevadas, sem exigir estruturas excessivamente robustas. Por outro lado, a principal desvantagem das cadeias fechadas é que elas possuem um espaço de trabalho menor em comparação com as cadeias abertas (TSAI, 1999).

Para que uma cadeia cinemática fechada se torne, de fato, um mecanismo, é necessário realizar um último passo: fixar um de seus elos em relação a um sistema de referência (e.g. sistema cartesiano). Dependendo de qual elo é escolhido como fixo no sistema referencial, o comportamento do mecanismo pode se modificar, especialmente quando os elos apresentam dimensões distintas entre si. Além disso, o elo fixo, isto é, o referencial, pode ser representado graficamente de diferentes maneiras. Nesta monografia, serão utilizadas duas formas de representação: a **representação funcional**, na qual o elo fixo é omitido e as juntas conectadas a ele são ilustradas apenas como mancais; e a **representação estrutural**, que exhibe explicitamente o elo fixo, geralmente por meio de linhas tracejadas que destacam seu caráter estático (MARTINS; MURAI, 2019).

A Figura 6 apresenta ambas as representações aplicadas a um mesmo mecanismo de

Figura 6 – Representações de um mecanismo de 4 barras

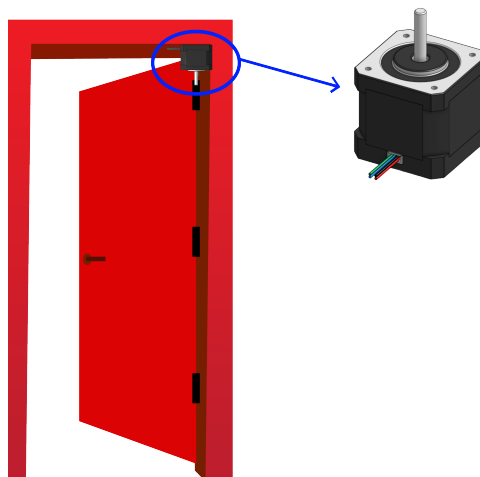


(a) Representação Funcional

(b) Representação Estrutural

Fonte: Adaptado de Plecnik (2015)

Figura 7 – Exemplo de uma porta com motor no eixo de rotação



Fonte: Elaborado pelo autor

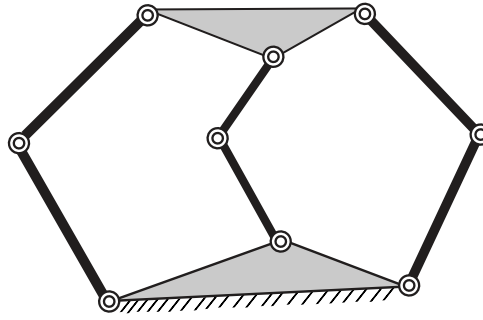
quatro barras, com o elo fixo circulado em vermelho. Conforme discutido no texto, a escolha de qual elo é fixado resulta em diferentes comportamentos cinemáticos.

2.1.1 Mobilidade

A **mobilidade** (M) de um mecanismo é uma das primeiras questões a se considerar ao iniciar um estudo mais detalhado sobre seu funcionamento. A mobilidade se refere ao número de parâmetros independentes, também chamados de **graus de liberdade**, necessários para especificar completamente o comportamento do mecanismo ou cadeia cinemática no espaço (TSAI, 2000). Em termos simples, ela determina quantos motores ou atuadores precisamos definir nas juntas para saber exatamente como o mecanismo inteiro vai se mover. Por exemplo, uma porta presa a uma dobradiça (Figura 7) precisaria apenas de um motor para especificar o movimento completamente, pois ela só pode rotacionar em torno de um eixo.

No caso da porta, é intuitivo concluir que apenas um motor é suficiente para defi-

Figura 8 – Mecanismo com dois circuitos



Fonte: Adaptado de Plecnik (2015)

nir completamente seu movimento. Entretanto, ao analisar mecanismos mais complexos, como o mecanismo de dois circuitos ilustrado na Figura 8, essa determinação se torna menos evidente. Nesse contexto, a aplicação da equação de mobilidade torna-se fundamental. Por meio dela, é possível, a partir de uma formulação simples, identificar de maneira rápida e precisa os graus de liberdade de um mecanismo, mesmo quando sua estrutura apresenta múltiplos elos e juntas. A aplicação da equação de mobilidade permite identificar, de maneira simples e objetiva, que o mecanismo de três circuitos da Figura 8 demanda três motores para que todo o seu movimento seja completamente especificado.

Na maioria dos casos, é possível chegar nesta expressão geral que calcula a mobilidade de um mecanismo a partir do seu número de elos, número de juntas e tipos de juntas. Para entender como cada junta contribui para essa mobilidade, é necessário considerar dois conceitos fundamentais: a **conectividade** (f_i) e a **restrição** (c_i) de cada par cinemático. A conectividade de uma junta indica a quantidade de graus de liberdade que ela permite, enquanto a restrição representa o número de limitações que o par cinemático impõe ao movimento do mecanismo. (MARTINS; MURAI, 2019).

Por exemplo, uma junta de revolução possui conectividade 1, pois possui apenas um grau de liberdade, enquanto o valor de restrição depende do grau de liberdade do **espaço** (λ) em que o mecanismo está inserido. Um objeto livre pode realizar até 6 movimentos diferentes (translações e rotações) no espaço de tridimensional, e se mover de 3 maneiras diferentes no plano ou em uma esfera (HUNT, 1990). Portanto, se o mecanismo estiver em um plano, a restrição da junta de revolução será 2; se estiver em um espaço 3D, a restrição será 5. Adicionalmente, nota-se que ao somar a conectividade à restrição de uma junta, o resultado é o grau de liberdade do espaço, como mostra a Equação 2.1.

$$f_i + c_i = \lambda \quad (2.1)$$

A expressão geral que retorna o valor de mobilidade de um mecanismo de cadeia cinemática fechada é dada pela equação 2.2. Caso o leitor tenha interesse em sua demonstração, recomenda-se a leitura do livro de Martins e Murai (2019).

$$M = (n - \bar{j} - 1) \cdot \lambda + \sum_{i=1}^{\bar{j}} f_i \quad (2.2)$$

A partir da Equação 2.2, conhecida como critério de Chebychev–Grübler–Kutzbach (GOGU, 2005), é possível calcular a mobilidade da maioria dos mecanismos. Para aplicar o conceito, utiliza-se como exemplo o mecanismo de quatro barras da Figura 5b, considerando o elo L_1 fixado. Partindo do fato de que o mecanismo possui 4 elos, supondo que as 4 juntas são de revolução e que o mecanismo atua num espaço planar, a mobilidade dele será

$$M = (4 - 4 - 1) \cdot 3 + \sum_{i=1}^4 1 = 1$$

Então, no plano, com um parâmetro independente, ou seja, uma junta de revolução independente, é possível definir totalmente a configuração do mecanismo. A condição de existência ou não de um mecanismo depende da expressão $M > 0$, pois, caso a mobilidade de sua cadeia cinemática seja menor que 1, não haverá movimento, como acontece em treliças.

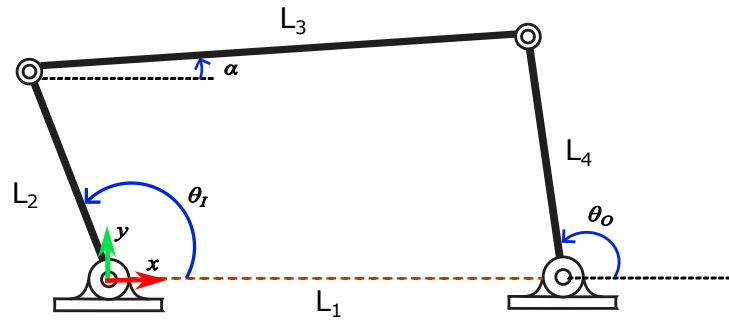
Ao acionar o mecanismo, podem ocorrer situações específicas em que seu comportamento é indesejado. Isso se deve à ocorrência de posições de **singularidade**. Nessas configurações, a cinemática do mecanismo é alterada, podendo resultar em ganho ou perda instantânea de graus de liberdade. Ao sair da posição singular, o mecanismo normalmente retoma sua mobilidade original (YAN, 1998). Um exemplo clássico de singularidade ocorre quando três juntas de revolução se alinham, o que provoca a perda de um grau de liberdade na direção desse alinhamento. Esse fenômeno pode ser facilmente observado em um braço humano: com o braço totalmente estendido, não é possível empurrá-lo na direção do alinhamento entre o ombro e o cotovelo, enquanto que, quando o braço está dobrado, esse movimento se torna viável.

2.1.2 Mecanismos Planares com 1 GDL

Com os principais conceitos de mecanismos explicitados, é possível introduzir representações estruturais de alguns mecanismos planares com 1 grau de liberdade, que serão o foco deste trabalho.

Observação: a nomenclatura **Mecanismo de 6 barras** adotada neste trabalho segue a definição apresentada por Erdman e Sandor (1984), na qual nem todos os elos precisam ser binários, e os mecanismos possuem dois circuitos. Essa definição difere de outras encontradas na literatura, onde mecanismos de 6 barras são, por vezes, estritamente considerados como estruturas hexagonais.

Figura 9 – Mecanismo de 4 barras



Fonte: Adaptado de Plecnik (2015)

2.1.2.1 Mecanismo de 4 barras

O mecanismo de quatro barras, mostrado na Figura 9, é a configuração mais simples de cadeia cinemática fechada. Ele é composto por um elo fixo L_1 (a base), um elo de entrada L_2 (onde aplicamos o movimento), um elo de saída L_4 (de onde retiramos o movimento) e uma barra acopladora L_3 , que transmite o movimento entre entrada e saída (ERDMAN; SANDOR, 1984).

Além da descrição estrutural, é importante considerar também como avaliar o desempenho desse mecanismo. Para isso, utiliza-se o **ângulo de qualidade de transmissão** (μ), um indicador da eficácia do mecanismo em função de sua geometria. Esse ângulo é definido entre a barra acopladora e a barra de saída (MARTINS; MURAI, 2019), e seu valor recomendado deve permanecer dentro do intervalo estabelecido pela inequação 2.3. Quando o valor do ângulo de qualidade de transmissão está dentro do intervalo recomendado, o esforço se distribui de maneira mais favorável entre as barras. Isso significa que o mecanismo exige menos força para produzir o mesmo movimento. Em um contexto prático, é como se o motor que aciona a entrada trabalhasse de forma mais eficiente: dentro da faixa adequada, ele consegue entregar o movimento desejado consumindo menos energia.

$$40^\circ \leq \mu \leq 140^\circ \quad (2.3)$$

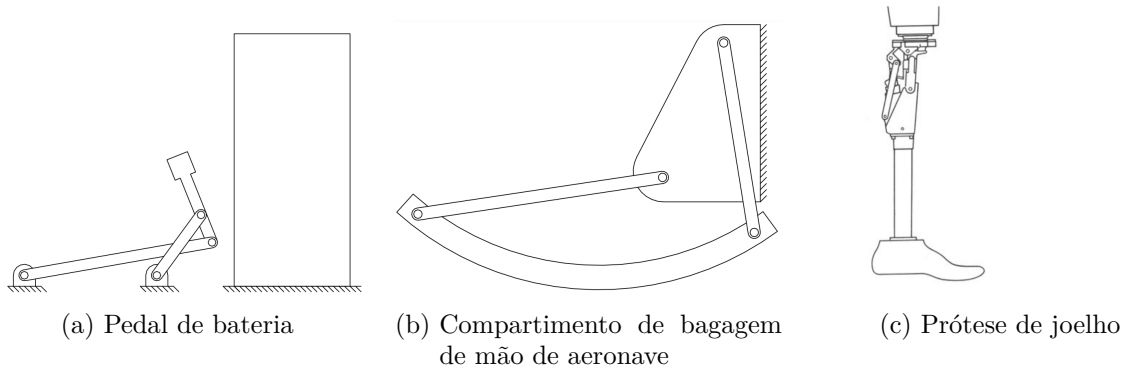
A apresentação deste mecanismo é fundamental para o desenvolvimento do trabalho, pois será utilizada para explicar de forma mais clara os passos que levarão aos mecanismos mais complexos de seis barras.

Alguns exemplos de mecanismos de 4 barras utilizados no mundo real podem ser vistos na Figura 10.

2.1.2.2 Mecanismos de 6 barras

Os mecanismos de 6 barras são as próximas cadeias cinemáticas fechadas mais simples, depois do 4 barras. Eles são organizados em duas topologias, as cadeias de **Watt** e as

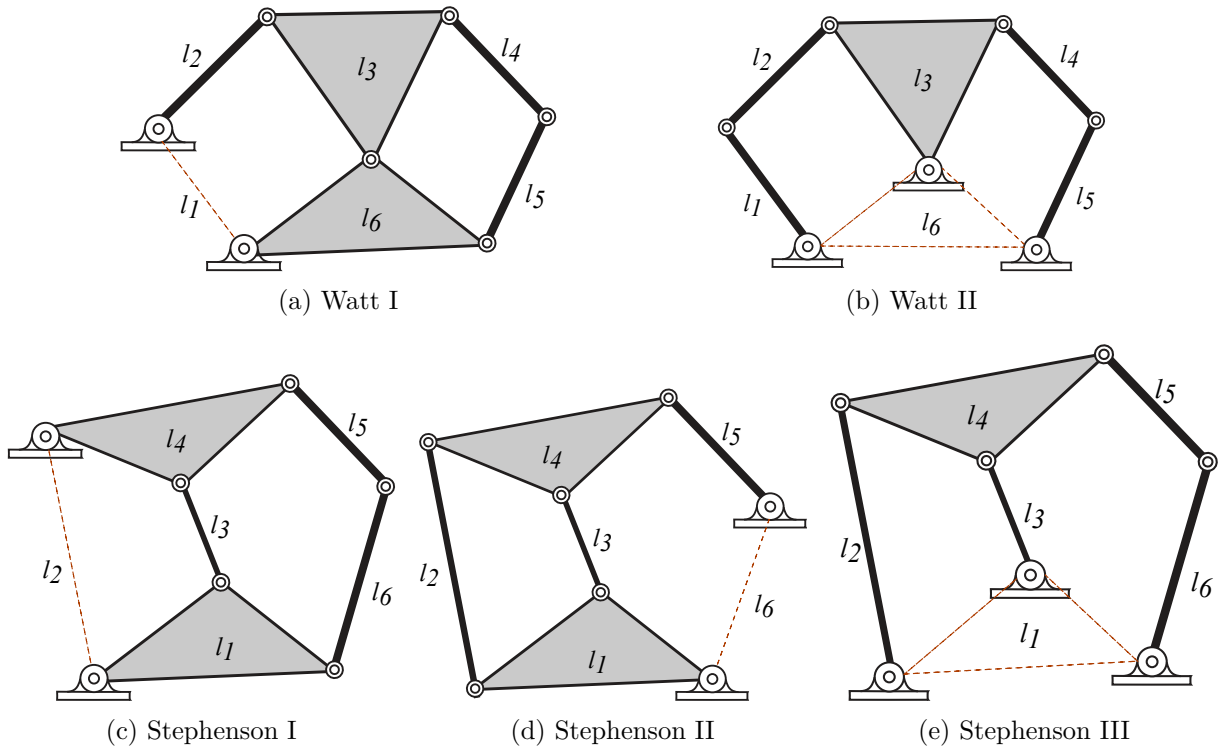
Figura 10 – Exemplos reais de mecanismos de 4 barras



Fonte: An Atlas of Structures, Mechanisms, and Robots de Dr. Bob Williams

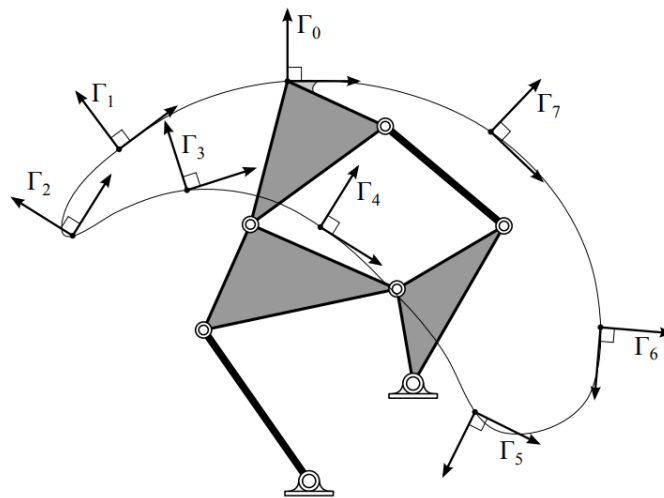
cadeias de **Stephenson**, onde cada topologia é classificada adicionalmente por qual elo é considerado o elo fixo, formando um total de cinco tipos diferentes de mecanismos de 6 barras, chamados **Watt I** (l_1 fixo), **Watt II** (l_6 fixo), **Stephenson I** (l_2 fixo), **Stephenson II** (l_6 fixo) e **Stephenson III** (l_1 fixo) (PLECNİK, 2015), podendo ser vistos na Figura 11. Nos mecanismos de Watt, os elos ternários (elos que possuem espaço para três juntas) são conectados entre si por uma junta, enquanto nos mecanismos de Stephenson, os elos ternários são separados por um elo binário (ERDMAN; SANDOR, 1984).

Figura 11 – Mecanismos de 6 barras



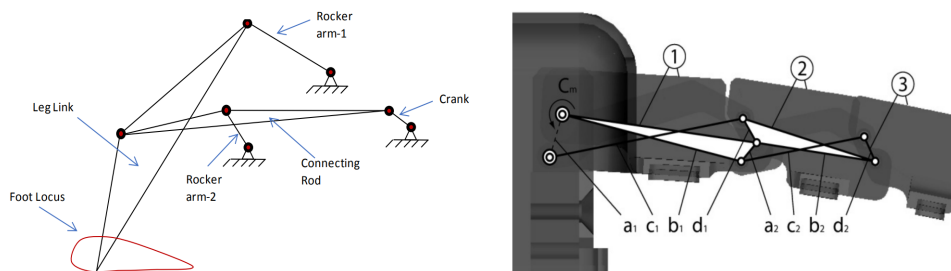
Fonte: Adaptado de Plecnik (2015)

Figura 12 – Geração de movimento de um Watt I



Fonte: (PLECNIK, 2015)

Figura 13 – Exemplos reais de mecanismos 6 barras



(a) Mecanismo de Klann (simula a perna de um animal quadrúpede) (b) Mecanismo do dedo da mão LARM

Fonte: (PRASHANTH; MANOJ; NIKHIL, 2019) e (CARBONE; ROSSI; SAVINO, 2015)

Embora os mecanismos de 6 barras representem apenas uma camada de complexidade a mais que o mecanismo de 4 barras, os movimentos que eles podem produzir são consideravelmente mais complexos. A Figura 12 mostra um exemplo de geração de movimento de um Watt I.

Alguns exemplos de mecanismos de 6 barras utilizados no mundo real podem ser vistos na Figura 13.

2.1.3 Análise Cinemática de Mecanismos de 6 barras

O processo de definir uma cadeia cinemática fechada e determinar sua mobilidade constitui o ponto de partida para a análise e a síntese de mecanismos (ERDMAN; SANDOR, 1984). Na análise, busca-se compreender o comportamento de uma máquina ou mecanismo já existente. Assim, partindo de uma geometria previamente definida, determinam-se suas características estruturais, além dos movimentos, posições, velocidades, acelera-

ções, forças e momentos ao longo do tempo (MARTINS; MURAI, 2019). Esse tipo de estudo permite responder perguntas como: “Se eu girar a entrada em 30° , quanto a saída irá girar?” ou “A barra de saída conseguirá alcançar determinada posição?”.

Neste trabalho, será realizada apenas a análise de posição dos mecanismos, a qual servirá de base para o desenvolvimento do otimizador responsável pela síntese dos mecanismos.

A **análise de posição** tem como objetivo determinar a posição e a orientação de cada componente de um mecanismo, dados os valores atribuídos aos seus graus de liberdade, que neste trabalho correspondem aos ângulos de entrada, pois todas as juntas são do tipo revolução. Para isso, resolvem-se as equações de restrição impostas pela cadeia cinemática fechada.

Há diversas abordagens para a obtenção das expressões de posição. Entre elas, destacam-se: o método das matrizes de DH (DENAVID; HARTENBERG, 1955), amplamente empregado na cinemática direta de manipuladores robóticos; o método do laço de vetores (Vector Loop), descrito em (ERDMAN; SANDOR, 1984), que pode ser formulado com números reais ou com números complexos em coordenadas polares; e o método geométrico, que determina sequencialmente as posições das juntas com base na própria geometria do mecanismo, conforme detalhado em (TSAI, 1999). Outras abordagens também podem ser utilizadas, dependendo da aplicação.

A análise dos mecanismos de seis barras neste trabalho será conduzida por meio do método geométrico, por ser mais direto e intuitivo na abordagem de mecanismos de cadeia fechada (TSAI, 1999). Além disso, sua simplicidade contribui para uma compreensão mais clara, eliminando a necessidade de resolver equações de igualdade numericamente, com exceção do mecanismo Stephenson II, que requer tal abordagem. Além de calcular as posições, também serão calculados o ângulo de saída e os ângulos de qualidade de transmissão.

Observação: Diferentemente do que ocorre com os mecanismos de quatro barras, não há na literatura uma convenção estabelecida para avaliar a qualidade de transmissão em mecanismos de seis barras. Assim, neste trabalho adotam-se dois ângulos de qualidade de transmissão, indicados nas representações gráficas da Tabela 1 como μ_1 e μ_2 .

Todos os mecanismos analisados apresentam mobilidade unitária ($M = 1$), conforme a equação 2.2, e são compostos exclusivamente por juntas de revolução. Considera-se o ângulo de entrada θ_I como variável independente, e assume-se que os parâmetros geométricos L_1, L_2, \dots, L_9 , ϕ , α e λ são previamente conhecidos. A partir das restrições geométricas de cada configuração, obtêm-se o ângulo de saída θ_O e os ângulos de qualidade de transmissão μ_1 e μ_2 .

Na Tabela 1, apresenta-se a análise de posição dos mecanismos Watt I, Watt II, Stephenson I, Stephenson II e Stephenson III, destacando apenas as principais equações que foram obtidas. As variáveis destacadas em vermelho são utilizadas como variáveis

de projeto no processo de otimização. Para os leitores que desejarem consultar o modelo completo, com todas as equações, este está disponível nos Apêndices A, B, C, D, E.

Cabe ressaltar que, embora os modelos propostos sejam plenamente funcionais para a síntese empregada neste trabalho, eles não se esgotam nas formulações apresentadas. Os sistemas de equações aqui descritos podem ser adaptados ou expandidos conforme os objetivos de estudos futuros. Por exemplo, dependendo da variável de interesse, diferentes combinações de parâmetros podem ser selecionadas como entradas ou saídas do modelo, possibilitando novas estratégias de otimização. No caso do mecanismo Watt I, por exemplo, poderia-se investigar a otimização de θ_I em função de β_2 vistos na Tabela 1, ou quaisquer outras relações cinemáticas relevantes, de acordo com as necessidades específicas de aplicações posteriores.

2.1.4 Síntese de Mecanismos

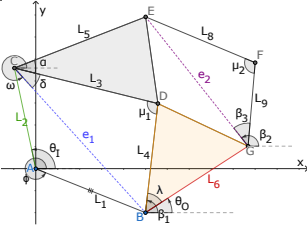
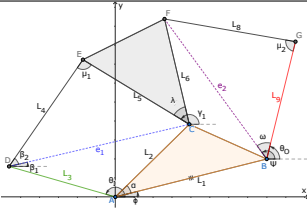
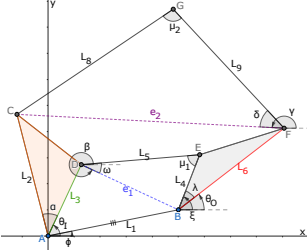
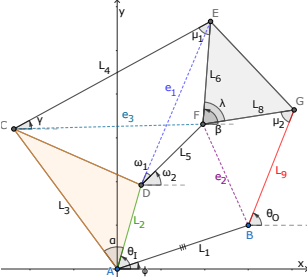
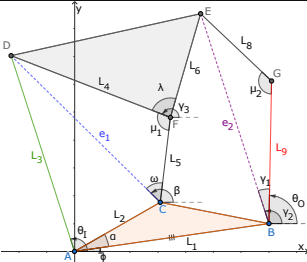
Na síntese de mecanismos, o objetivo é projetar dispositivos capazes de atender a requisitos específicos de movimento. Pode ser necessário, por exemplo, deslocar uma ferramenta da posição A para a posição B em um intervalo de tempo determinado, ou traçar um caminho específico no espaço para inserir uma peça em um conjunto. As possibilidades de aplicação são amplas (NORTON, 1998). A síntese considerada neste trabalho abrange exclusivamente os requisitos estruturais, que são mensuráveis. Dessa forma, não são considerados os requisitos de projeto, como materiais, processos de fabricação ou aspectos construtivos.

Existem três etapas fundamentais na síntese de mecanismos, que devem ser realizadas em sequência: a síntese do número, a síntese do tipo e a síntese dimensional. A síntese do número tem como objetivo determinar o número de elos e de pares cinemáticos necessários, bem como a forma como esses elementos se conectam entre si. Na síntese do tipo, define-se quais tipos de pares cinemáticos realizam essas conexões. Por fim, na síntese dimensional determinam-se as dimensões dos elos, como comprimentos e ângulos, de modo que o mecanismo apresente a cinemática desejada (MARTINS; MURAI, 2019).

Neste trabalho, aplica-se exclusivamente a síntese dimensional, uma vez que as etapas de síntese do número e de síntese do tipo são consideradas previamente definidas: utiliza-se um mecanismo de seis barras, e todas as juntas são do tipo revolução. Para leitores que desejarem aprofundar-se nas demais etapas da síntese, recomenda-se a consulta às seguintes referências: (YAN, 1998), (UICKER JR., 2017), (LIU; WANG, 2014), (CHUNG; LEE, 2001), (NORTON, 1998), (ERDMAN; SANDOR, 1984), (TSAI, 2000) e (FRISOLI; SALSEDO; BERGAMASCO, 1999).

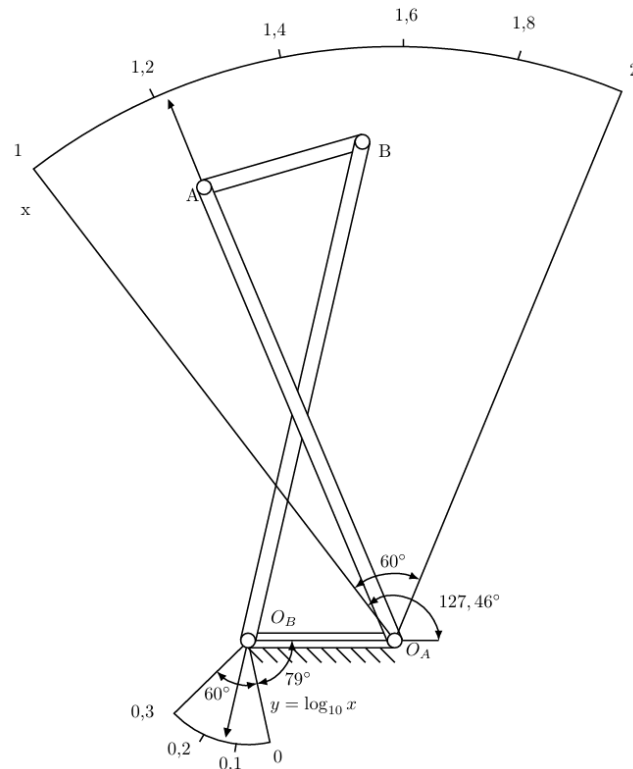
Na síntese dimensional, existem três principais vertentes para determinar as dimensões dos elos de um mecanismo. A primeira é a geração de caminho, que busca definir as dimensões necessárias para que um ponto de interesse percorra um trajeto específico

Tabela 1 – Resumo visual dos mecanismos e suas equações principais

Mecanismo	Imagem Mecanismo	Equações principais
Watt I		$\theta_O = \beta_1 - \lambda$ $\mu_1 = \arccos\left(\frac{L_4^2 + L_3^2 - e_1^2}{2L_4L_3}\right)$ $\mu_2 = \arccos\left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9}\right)$
Watt II		$\theta_O = \psi - \omega$ $\mu_1 = \arccos\left(\frac{L_4^2 + L_5^2 - e_1^2}{2L_4L_5}\right)$ $\mu_2 = \arccos\left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9}\right)$
Stephenson I		$\theta_O = \xi - \lambda$ $\mu_1 = \arccos\left(\frac{L_4^2 + L_5^2 - e_1^2}{2L_4L_5}\right)$ $\mu_2 = \arccos\left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9}\right)$
Stephenson II		$\theta_O = \arctan\left(\frac{G_y - B_y}{G_x - B_x}\right)$ $\mu_1 = \arccos\left(\frac{L_4^2 + L_6^2 - e_3^2}{2L_4L_6}\right)$ $\mu_2 = \arccos\left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9}\right)$
Stephenson III		$\theta_O = \beta_1 - \lambda$ $\mu_1 = \arccos\left(\frac{L_4^2 + L_3^2 - e_1^2}{2L_4L_3}\right)$ $\mu_2 = \arccos\left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9}\right)$

quando o mecanismo é acionado. A segunda é a geração de movimento, cujo objetivo é determinar as dimensões a partir de um conjunto finito de posições homólogas de um plano, controlando posição e orientação (MARTINS; MURAI, 2019). A terceira é a geração de função, na qual se estabelece uma relação funcional desejada entre as variáveis de entrada e de saída do mecanismo.

Os dois primeiros tipos de síntese dimensional (caminho e movimento) não serão abordados neste trabalho. Para leitores interessados nesses temas, recomenda-se a consulta a (TSAI, 2000), (UICKER JR., 2017), (CHU; SUN, 2010) e (RUSSELL; SODHI, 2005). O

Figura 14 – Mecanismo que calcula $\log_{10} x$ 

Fonte: (MARTINS; MURAI, 2019)

terceiro tipo (função) será explorado em maior profundidade na próxima seção.

2.1.4.1 Síntese de Geração de Função

A síntese de geração de função tem como objetivo definir as dimensões do mecanismo de modo que seu movimento represente uma função desejada. A ideia é simples de visualizar: para cada valor do ângulo de entrada, busca-se que o mecanismo produza um valor específico de ângulo de saída, como se estivéssemos construindo uma “calculadora mecânica”. Um exemplo clássico desse tipo de síntese está ilustrado na Figura 14, onde um mecanismo de quatro barras é utilizado para aproximar a função $\log_{10} x$ (MARTINS; MURAI, 2019).

Neste trabalho, a definição de síntese de geração de função segue Hartenberg e Denavit (1964), na qual as barras de entrada e saída são consideradas em relação aos pivôs da barra fixa.

Um dos procedimentos mais utilizados para realizar a síntese de geração de função de mecanismos de quatro barras é baseado nas equações de Freudenstein (1954). Para realizá-lo, utiliza-se como exemplo o quatro barras e seus parâmetros expostos na Figura 9, a fim de determinar as dimensões dos elos que geram a função $\theta_O = f(\theta_I)$. Para determinar a equação, parte-se da geometria do quatro barras, como mostram as Equações 2.4, que modela utilizando o método vetorial.

$$\begin{aligned} x : L_2 \cos(\theta_I) + L_3 \cos(\alpha) - L_4 \cos(\theta_O) &= L_1 \\ y : L_2 \sin(\theta_I) + L_3 \sin(\alpha) &= L_4 \sin(\theta_O) \end{aligned} \quad (2.4)$$

Isolando α , tem-se

$$\begin{aligned} x : L_3 \cos(\alpha) &= L_1 + L_4 \cos(\theta_O) - L_2 \cos(\theta_I) \\ y : L_3 \sin(\alpha) &= L_4 \sin(\theta_O) - L_2 \sin(\theta_I) \end{aligned}$$

Elevando ambas ao quadrado e somando-as, obtém-se:

$$L_3^2(\cos^2(\alpha) + \sin^2(\alpha)) = (L_1 - L_2 \cos(\theta_I) + L_4 \cos(\theta_O))^2 + (L_4 \sin(\theta_O) - L_2 \sin(\theta_I))^2$$

Expandindo os termos e fazendo substituições trigonométricas, chega-se à Equação de Freudenstein 2.5, com $\cos(\theta_O - \theta_I)$ isolado.

$$\cos(\theta_O - \theta_I) = \frac{L_1^2 + L_2^2 + L_4^2 - L_3^2}{2L_2L_4} - \frac{L_1}{L_4} \cos(\theta_I) + \frac{L_1}{L_2} \cos(\theta_O) \quad (2.5)$$

A Equação 2.5 possui três incógnitas, considerando L_1 com valor 1, que são os termos que possuem as dimensões dos elos. Portanto, ela consegue resolver sínteses de geração de função para três pares de ângulos de entrada e saída, já que um sistema com três incógnitas necessita de três equações para ser resolvido.

A equação proposta por Freudenstein (1954) foi pioneira na síntese de geração de função para mecanismos de quatro barras, permitindo a solução analítica do problema clássico de síntese para três posições, também conhecido como síntese com três pontos de precisão. Posteriormente, métodos mais avançados foram desenvolvidos para sínteses envolvendo quatro e cinco pontos de precisão, fundamentados em formulações trigonométricas, conforme apresentado em Hartenberg e Denavit (1964), e em representações com números complexos, conforme discutido em Erdman e Sandor (1984).

Para mecanismos de seis barras, diversas obras da literatura propuseram sínteses de geração de função utilizando abordagens analíticas ou numéricas e estocásticas.

Entre os métodos analíticos, destacam-se: Svoboda e James (1948), que projetaram um gerador de função para o tipo Watt II e patenteou um mecanismo capaz de calcular uma função logarítmica; McLarnan (1963), que formulou as equações de síntese para os geradores de função dos tipos Watt II, Stephenson II e Stephenson III, utilizando o método de Newton-Raphson para resolver mecanismos capazes de atender até oito pontos de precisão em um computador IBM 704; e Plecnik e McCarthy (2014), que aplicaram algoritmos de homotopia ao projeto de geradores de função dos tipos Watt II, Stephenson II e Stephenson III, obtendo conjuntos de soluções quase completos para problemas de oito posições.

Quanto aos métodos estocásticos ou baseados em otimizadores, Shiakolas, Koladiya e Kebrle (2005) aplicaram o algoritmo de evolução diferencial ao projeto de mecanismos de

seis barras com movimento de pausa (dwell). Hwang e Chen (2010) formularam a síntese ótima de geradores de função do tipo Stephenson II, com foco em evitar defeitos de ordem, circuito e ramificação. Por fim, Bulatović, Đorđević e Đorđević (2013) introduziram o algoritmo Cuckoo Search no projeto de mecanismos do tipo Stephenson III.

Apesar do avanço proporcionado por essas contribuições, observa-se que a maior parte dos trabalhos disponíveis dedica-se ao estudo de apenas um tipo de mecanismo, geralmente sob condições específicas de aplicação, e concentra-se principalmente no ajuste cinemático entre variáveis de entrada e saída. Em geral, esses métodos não incorporam de forma explícita aspectos essenciais ao comportamento interno do mecanismo, tais como qualidade de transmissão, restrições geométricas de proporcionalidade, limites articulares ou requisitos de viabilidade construtiva.

Nesse contexto, a abordagem proposta neste trabalho distingue-se por oferecer um método abrangente e flexível: o programa computacional desenvolvido integra, em uma mesma ferramenta, todos os tipos de mecanismos de seis barras (Watt I, Watt II, Stephenson I, II e III) e permite a personalização tanto dos parâmetros geométricos quanto dos critérios de otimização. Essa capacidade de ajustar o processo de síntese a diferentes objetivos projetivos, contemplando simultaneamente múltiplos tipos de mecanismos e incorporando variáveis internas relevantes, configura uma contribuição significativa quando comparada às literaturas tradicionais.

2.1.4.2 Metodologias para Síntese de Mecanismos

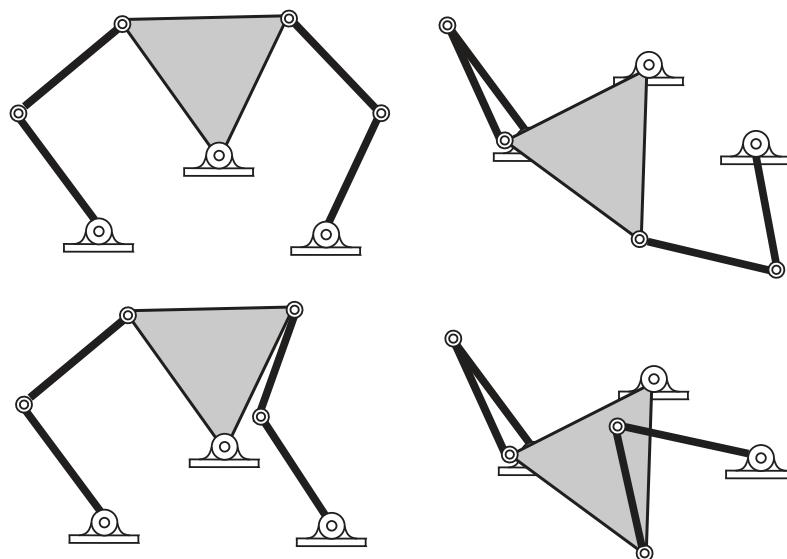
A criação de novos mecanismos é uma tarefa complexa, que exige do projetista a adoção de uma metodologia adequada (YAN, 1998). Existem diversas metodologias para síntese de mecanismos disponíveis na literatura, como a de Hartenberg e Denavit (HARTENBERG; DENAVIT, 1964), a de Hunt (HUNT, 1990) e a de Yan (YAN, 1998).

Porém, para a realização dos estudos de caso desse trabalho, a metodologia adotada é a de Martins e Murai (MARTINS; MURAI, 2019), que segue a seguinte ordem: revisão do estado da arte, análise de mecanismos existentes, definição dos requisitos, síntese do número, síntese do tipo, síntese dimensional, prototipagem, documentação e fabricação. Entre as fases de síntese do número e de prototipagem, há uma retroalimentação contínua até que uma solução seja encontrada.

2.1.5 Configurações de Montagem de Mecanismos

Ao realizar a síntese de um mecanismo, determinam-se as dimensões dos elos capazes de compor sua estrutura. Entretanto, é importante destacar que a maioria dos mecanismos admite mais de uma configuração de montagem, o que significa que um mesmo conjunto de dimensões pode originar diferentes geometrias de montagem. Conforme discutido por Parrish e McCarthy (2013), um mecanismo de seis barras pode apresentar até

Figura 15 – Montagens do mecanismo Watt II



Fonte: Adaptado de Plecnik (2015)

seis montagens distintas, cada uma resultando em um comportamento cinemático diferenciado. A Figura 15 ilustra duas possíveis configurações de montagem para o mecanismo de seis barras do tipo Watt II.

2.1.6 Defeitos de Síntese de Mecanismos

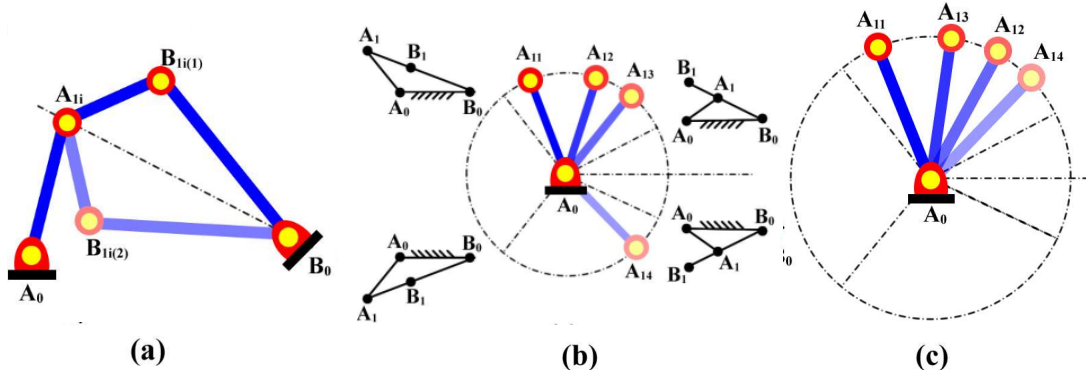
No processo de síntese, podem surgir defeitos que comprometem a utilização do mecanismo, sendo fundamental conhecê-los para preveni-los. Os três principais são os defeitos de **circuito**, de **ramificação** e de **ordem**.

O Defeito de circuito ocorre quando o mecanismo necessita ser desmontado e remontado em um modo de trabalho diferente para que o mesmo atinja todas as posições desejadas. O de ramificação ocorre quando o mecanismo se desloca até uma posição desejada e a barra de saída se alinha com a barra acopladora. E no defeito de ordem, o mecanismo é capaz de passar pelas posições desejadas, porém, não o faz na ordem desejada (MARTINS; MURAI, 2019). Na Figura 16 os principais defeitos podem ser visualizados.

2.2 Algoritmos de Otimização

Diversas disciplinas têm a otimização como parte essencial de seus fundamentos. Na física, por exemplo, os sistemas tendem naturalmente ao estado de menor energia, respeitando as leis que governam seu comportamento. No setor empresarial, o objetivo costuma ser a maximização do valor para os acionistas. O sistema em análise pode ser físico, como uma aeronave ou até mesmo uma estrutura simples, como o quadro de uma bicicleta; mas também pode ser abstrato, como um sistema de controle para veículos autônomos ou um

Figura 16 – Defeitos de Circuito (a), Ramificação (b) e Ordem (c) em mecanismos



Fonte: (ZHAO et al., 2022)

modelo de visão computacional projetado para identificar câncer em imagens de biópsia. Em todos os casos, busca-se alcançar o melhor desempenho possível (KOCHENDERFER; WHEELER, 2019).

Em um problema de otimização, procura-se maximizar ou minimizar uma função denominada **função objetivo**, que quantifica critérios como custo, erro ou eficiência. Em aplicações específicas, diferentes métricas podem ser adotadas. No contexto de mecanismos e robótica, a otimização pode envolver, por exemplo, a minimização do erro de trajetória ou a maximização da eficiência energética, entre outros objetivos relevantes.

As variáveis presentes na função objetivo são chamadas de **variáveis de projeto** e podem representar dimensões físicas ou propriedades de materiais. Tais variáveis estão sujeitas a limites inferior e superior por razões práticas e podem ser contínuas ou discretas (BASKAN, 2016). O problema de otimização pode ser expresso matematicamente da seguinte forma:

$$\text{Minimizar } f(x) \tag{2.6}$$

$$\text{sujeito a } g_i(x) \leq 0, \quad i = 1, 2, \dots, m < n \tag{2.7}$$

$$h_j(x) = 0, \quad j = 1, 2, \dots, r < n \tag{2.8}$$

$$x_l \leq x \leq x_u \tag{2.9}$$

O vetor x representa n variáveis de projeto:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

As restrições g_i são denominadas **restrições de desigualdade**, enquanto h_j correspondem às **restrições de igualdade**. Tais restrições são funções das variáveis de projeto

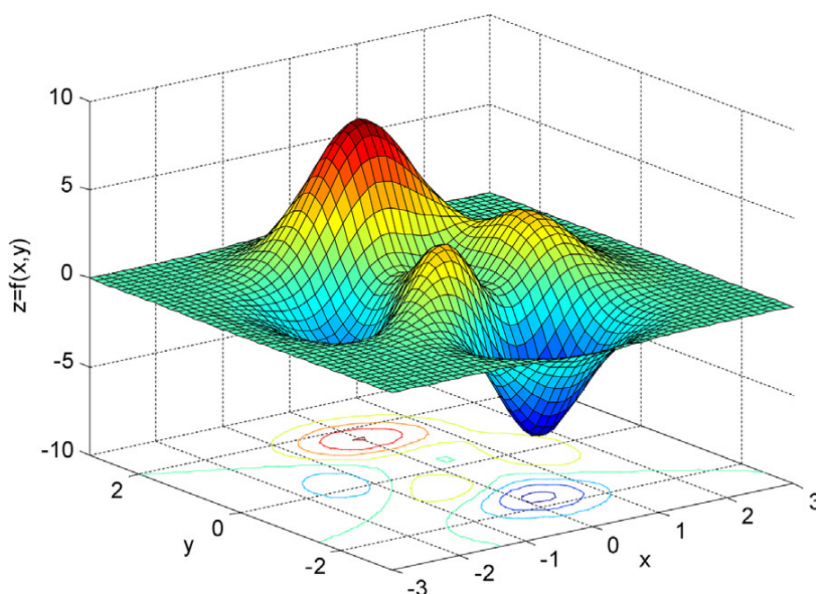
e definem o conjunto de soluções viáveis. Um ponto é considerado viável quando satisfaz todas as restrições.

Na otimização, distinguem-se dois tipos de soluções: o **mínimo local** e o **mínimo global**. Em funções não convexas ou multimodais, podem existir vários mínimos locais, isto é, pontos que representam o menor valor apenas em relação à sua vizinhança imediata. Já o mínimo global corresponde ao menor valor absoluto que a função pode assumir em todo o seu domínio (PRICE; STORN; LAMPINEN, 2006). A Figura 17 exemplifica uma função multimodal com múltiplos mínimos locais.

A distinção entre mínimos locais e globais é fundamental, pois influencia diretamente a escolha do método de otimização. Dependendo da complexidade da função e da presença de múltiplos ótimos locais, diferentes abordagens podem ser necessárias para aumentar as chances de encontrar a melhor solução. Nesse contexto, os algoritmos de otimização são geralmente classificados em duas categorias principais: **determinísticos** e **estocásticos**. Métodos determinísticos fornecem garantias teóricas para localizar mínimos locais ou globais, enquanto métodos estocásticos exploram o espaço de busca de forma probabilística, sem assegurar que o mínimo global será encontrado, sendo mais adequados para funções altamente não convexas ou multimodais (HARTKE, 2011).

Neste trabalho, o foco recai sobre a aplicação de algoritmos estocásticos, uma vez que as funções objetivo provenientes dos mecanismos de seis barras são não lineares e multimodais. Essa característica torna mais adequada a utilização de métodos probabilísticos em comparação aos determinísticos, devido ao elevado custo computacional associado à busca do mínimo global nestes últimos. Ainda assim, algoritmos determinísticos podem ser empregados em uma etapa posterior, após a otimização estocástica, com o objetivo de

Figura 17 – Exemplo de função multimodal



Fonte: (PRICE; STORN; LAMPINEN, 2006)

refinar a solução e identificar um mínimo local mais preciso.

No campo da otimização estocástica, a literatura apresenta diversos algoritmos desenvolvidos para a busca do mínimo global de funções. Entre os mais relevantes, destacam-se o *Simulated Annealing*, proposto por (KIRKPATRICK; GELATT; VECCHI, 1983) e inspirado no processo de recozimento da metalurgia, no qual soluções temporariamente piores podem ser aceitas a fim de escapar de mínimos locais; o *Tabu Search* (GLOVER; TAILLARD; TAILLARD, 1993), que busca superar ótimos locais ao permitir movimentos não melhorantes e evitar ciclos por meio de uma lista tabu; e os **Algoritmos Evolucionários** (AEs), que constituem uma classe de métodos de otimização inspirados na evolução natural e desenvolvidos de forma independente a partir da década de 1960, incluindo os Algoritmos Genéticos formalizados por Holland (HOLLAND, 1992). Desde então, os AEs consolidaram-se como uma abordagem robusta e amplamente utilizada para problemas de otimização complexos, caracterizados por múltiplos ótimos locais e funções objetivo sem propriedades analíticas específicas, como continuidade ou derivabilidade.

A lógica fundamental desses algoritmos baseia-se no uso de uma **população** de soluções candidatas que evolui ao longo das gerações por meio de operadores de variação (como recombinação, mutação ou perturbações estocásticas) e mecanismos de seleção que favorecem indivíduos mais aptos. Cada variante de AE implementa esses princípios de maneira distinta: por exemplo, os Algoritmos Genéticos representam soluções como cromossomos e utilizam operadores de cruzamento e mutação.

Entre as vantagens gerais dos Algoritmos Evolucionários destacam-se a capacidade de escapar de mínimos locais, a flexibilidade em lidar com diferentes tipos de funções objetivo e a robustez diante de espaços de busca altamente não lineares. Por outro lado, apresentam limitações como o elevado custo computacional, devido à necessidade de avaliar múltiplos indivíduos em cada geração, e a sensibilidade à configuração de parâmetros, que influencia significativamente a eficiência da convergência (FOUSKAKIS; DRAPER, 2002).

Dentre os Algoritmos Evolucionários, destaca-se a **Evolução Diferencial**, proposta por Storn e Price (1997), amplamente reconhecida na literatura como uma das abordagens mais eficazes para problemas de otimização contínua. Este método será apresentado em maior detalhe na subseção seguinte.

No contexto deste trabalho, vários algoritmos poderiam ser empregados para resolver o problema formulado. Entretanto, o método *Differential Evolution* foi escolhido em função de seu desempenho superior nos testes preliminares conduzidos pelo autor. Embora os demais algoritmos avaliados também tenham apresentado resultados válidos, a Evolução Diferencial mostrou-se consistentemente mais eficiente, em conformidade com o desempenho amplamente reportado na literatura especializada.

2.2.1 Algoritmo Evolução Diferencial

As informações apresentadas nesta subseção baseiam-se em Storn e Price (1997), artigo no qual o método foi originalmente proposto, e em Price, Storn e Lampinen (2006), que oferece uma descrição abrangente de suas variações e aplicações. A Evolução Diferencial (DE) é uma heurística de busca, isto é, uma estratégia prática que procura soluções suficientemente boas para problemas complexos, sem garantir a obtenção da solução ótima global. Seu funcionamento pode ser entendido como um processo de exploração guiada, no qual múltiplas possibilidades são avaliadas até que se alcance uma resposta adequada.

A atratividade da DE reside no fato de ser simples de implementar, exigir poucos parâmetros de ajuste e apresentar bom desempenho em problemas desafiadores, como aqueles em que a função objetivo é não linear, não derivável ou apresenta diversos mínimos e máximos espalhados, caracterizando um comportamento multimodal. Por essas características, a DE tornou-se uma ferramenta amplamente utilizada em diversas áreas da engenharia e das ciências aplicadas.

O algoritmo opera sobre uma população de NP vetores de dimensão D ($x_{i,G}$, $i = 1, 2, \dots, NP$), para cada geração G . O vetor inicial da população é escolhido aleatoriamente dentro do espaço de parâmetros. A cada iteração, os vetores são atualizados por meio de três operadores fundamentais: **mutação**, **recombinação** (*crossover*) e **seleção**. O processo de otimização por este método pode ser comparado a um grupo de exploradores em uma montanha:

- **Mutação**: alguns exploradores tentam novos caminhos, usando como referência a diferença entre outros pontos.
- **Recombinação**: misturam informações de caminhos diferentes para gerar rotas alternativas.
- **Seleção**: mantêm apenas as rotas que chegam a lugares melhores.

Formalmente, na etapa de **mutação**, para cada vetor-alvo $x_{i,G}$, gera-se um vetor mutante $v_{i,G+1}$ a partir da soma de um vetor base com a diferença ponderada entre dois outros vetores distintos da população:

$$v_{i,G+1} = x_{r_1,G} + F \cdot (x_{r_2,G} - x_{r_3,G}), \quad (2.10)$$

em que $r_1, r_2, r_3 \in \{1, 2, \dots, NP\}$ são índices aleatórios e diferentes entre si e de i , e $F \in [0, 2]$ é o fator de escala que controla a intensidade da perturbação. Essa estratégia, denominada *DE/rand/1*, é a forma mais clássica do método e garante diversidade no processo de busca.

Em seguida, aplica-se a **recombinação** (ou *crossover*), combinando o vetor mutante com o vetor-alvo. O vetor de teste $u_{i,G+1} = (u_{1i,G+1}, u_{2i,G+1}, \dots, u_{Di,G+1})$ é construído da

seguinte forma:

$$u_{ji,G+1} = \begin{cases} v_{ji,G+1}, & \text{se } \text{rand}_j \leq CR \text{ ou } j = j_{\text{rand}}, \\ x_{ji,G}, & \text{caso contrário,} \end{cases} \quad (2.11)$$

onde $\text{rand}_j \in [0, 1]$ é um número aleatório uniforme, $CR \in [0, 1]$ é a taxa de recombinação, e j_{rand} é um índice escolhido aleatoriamente para garantir que pelo menos um parâmetro seja herdado do vetor mutante $v_{i,G+1}$.

Por fim, a **seleção** decide se o vetor de teste $u_{i,G+1}$ deve ou não virar um membro da próxima geração, e é realizada comparando o vetor de teste com o vetor-alvo $x_{i,G}$ de acordo com o valor da função objetivo. O critério adotado é o guloso (*greedy criterion*):

$$x_{i,G+1} = \begin{cases} u_{i,G+1}, & \text{se } f(u_{i,G+1}) \leq f(x_{i,G}), \\ x_{i,G}, & \text{caso contrário.} \end{cases} \quad (2.12)$$

Assim, apenas soluções que apresentam melhor desempenho são mantidas na próxima geração, promovendo a convergência da população em direção a ótimos globais.

Os experimentos apresentados em Storn e Price (1997) demonstraram que a Evolução Diferencial foi capaz de superar métodos consagrados de otimização global, como o *Simulated Annealing* e Algoritmos genéticos, em diversos conjuntos de funções-teste de referência. O método não apenas apresentou maior consistência na obtenção do mínimo global, como também exigiu um número menor de avaliações da função objetivo, o que reforça sua eficiência. Além disso, os parâmetros do algoritmo, tamanho da população (NP), fator de escala (F) e taxa de recombinação (CR), podem ser ajustados dentro de intervalos relativamente bem definidos, simplificando a aplicação prática.

Em síntese, a Evolução Diferencial consolidou-se como uma das abordagens mais eficazes entre os algoritmos evolucionários, aliando simplicidade, eficiência e versatilidade. Embora a fundamentação teórica sobre sua convergência ainda não esteja totalmente estabelecida, sua eficácia prática e ampla utilização em problemas reais confirmam sua relevância como técnica de otimização contínua.

2.2.2 Aplicações da Evolução Diferencial em Mecanismos

Além dos estudos apresentados na Seção 2.1.4.1, relacionados à geração de função, a otimização também tem sido aplicada em outros problemas complexos de engenharia desenvolvidos na UFSC. Na sequência, são apresentados trabalhos que se relacionam direta ou indiretamente com a temática desta pesquisa e que contribuíram de maneira relevante para o seu desenvolvimento.

Em Weihmann, Martins e dos Santos Coelho (2012) é proposta uma versão modificada da Evolução Diferencial (MDE), na qual os parâmetros de mutação e cruzamento variam ao longo das gerações, buscando evitar a convergência prematura e melhorar a exploração do espaço de busca. A abordagem foi aplicada à otimização da capacidade de força de um

manipulador paralelo planar 3-RRR, comparando 10 estratégias clássicas de DE (best/1, best/2, rand/1, rand/2 e rand-to-best, em versões bin e exp) com suas variantes MDE.

O artigo de Mejia, Simas e Martins (2015) apresenta uma solução fechada para calcular a força máxima com momento prescrito em manipuladores planares de três graus de liberdade, derivada a partir da teoria dos parafusos, do método de Davies e de técnicas de otimização com restrições. A formulação inicial como problema de maximização sujeita a limites de torque permitiu estruturar as equações que se transformaram em uma solução analítica geral, aplicável a manipuladores seriais, paralelos e híbridos. A validação em um manipulador serial 3 DoF e em um paralelo 3RRR mostrou que os resultados coincidem com métodos de otimização, incluindo evolução diferencial, mas com ganho expressivo de desempenho, fornecendo respostas em frações de segundo em vez de horas. O trabalho mostra a importância dos métodos numéricos, que forneceram bases de comparação para a formulação subsequente de uma solução matemática fechada.

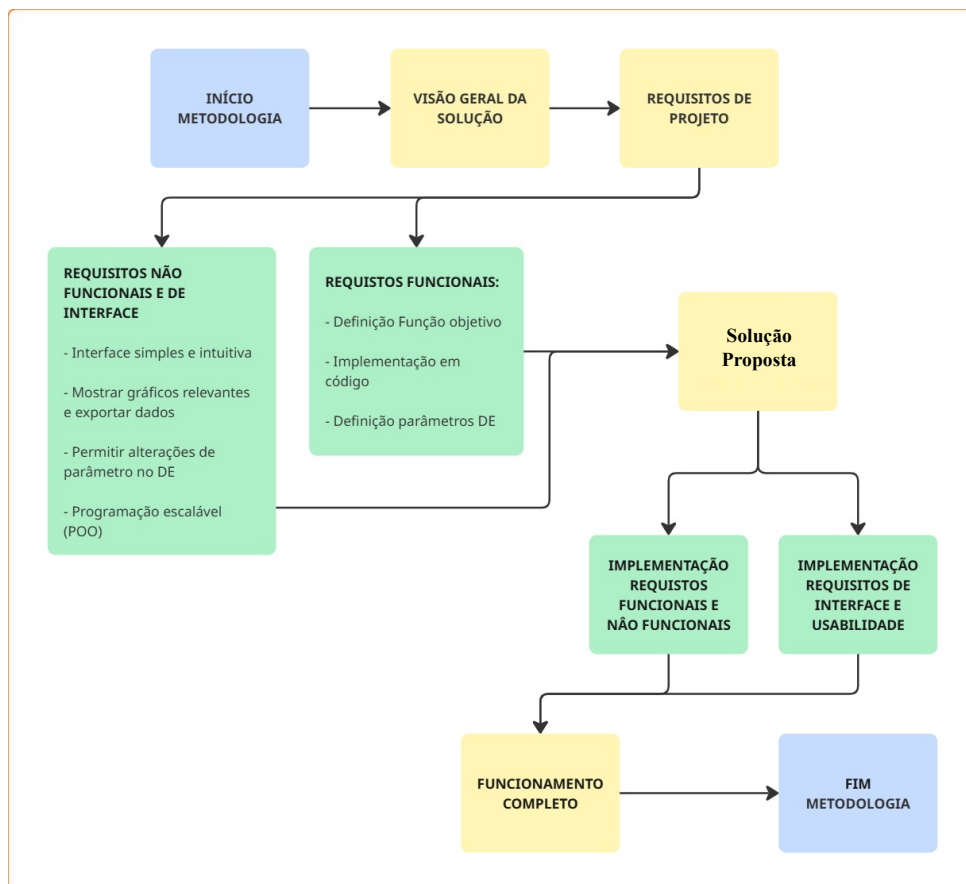
Já Krepsky et al. (2025), trabalho anterior do autor, apresenta uma aplicação parcial do programa descrito neste trabalho, na qual a otimização por síntese de geração de função do mecanismo de Watt 1 foi usada como etapa inicial na síntese dos dedos de uma mão robótica protética. Dessa forma, foi possível projetar um mecanismo de dedo com dimensões compatíveis às de um dos autores, preservando a eficiência de transmissão. A mão foi impressa em 3D e testada fisicamente em diferentes cenários de pegada de objetos, demonstrando a eficácia do projeto dos mecanismos. O projeto desenvolvido neste artigo de conferência será utilizado como um dos estudos de caso para comprovar a usabilidade do programa.

3 Metodologia

Nesta seção, apresenta-se a metodologia adotada para a síntese de mecanismos de seis barras por meio de técnicas de otimização computacional. A abordagem possui caráter predominantemente experimental e computacional, pois envolve o desenvolvimento de um sistema capaz de gerar mecanismos com base em critérios previamente definidos.

O capítulo está organizado da seguinte forma: inicialmente, apresenta-se uma visão geral da solução proposta, destacando as principais etapas da metodologia. Em seguida, são descritos os requisitos do projeto, divididos em requisitos funcionais, relacionados às áreas de mecanismos e otimização, e requisitos de interface e usabilidade, ligados à organização, apresentação das informações e interação com o usuário. Posteriormente, detalha-se a implementação da solução, descrevendo os procedimentos adotados para atender aos requisitos estabelecidos. Por fim, é realizada uma demonstração completa da interface, ilustrando passo a passo como o sistema pode ser utilizado na síntese do mecanismo. Um fluxograma que representa a estrutura desta seção é apresentado na Figura 18.

Figura 18 – Fluxograma de estruturação do capítulo



Fonte: Elaborado pelo autor

3.1 Visão Geral da Solução

A solução proposta foi desenvolvida em etapas sequenciais. Inicialmente, realizaram-se as modelagens matemáticas dos cinco mecanismos de seis barras, considerando apenas um arranjo específico de montagem para cada caso. Em seguida, tais modelos foram implementados na linguagem de programação Python.

Na segunda etapa, conduziram-se testes preliminares de otimização utilizando o algoritmo *Differential Evolution* (DE), com o objetivo de validar a implementação computacional de um dos mecanismos. Após verificar o correto funcionamento do algoritmo, foram elaboradas as funções objetivo de todos os mecanismos, estruturadas a partir de um mesmo template. Para garantir escalabilidade e evitar redundâncias, adotou-se a abordagem de programação orientada a objetos, concentrando em classes os parâmetros e variáveis comuns aos diferentes mecanismos.

Com as funções objetivo concluídas, incluindo a modelagem dos mecanismos, a definição das restrições para evitar defeitos e a configuração dos parâmetros padrão do DE, seguiu-se o desenvolvimento de funções complementares, tais como geração de gráficos, exportação de resultados para planilhas e construção da interface do sistema.

Por fim, foram realizados testes finais de validação, confirmando a consistência da solução. Concluída essa etapa, o programa foi considerado apto para aplicação em estudos de caso, os quais demonstraram sua utilização prática na síntese de mecanismos.

3.2 Requisitos do Projeto

Nesta seção, apresentam-se os requisitos definidos ao longo do planejamento e da execução do projeto. Eles estão organizados em duas categorias principais: **requisitos funcionais** e **requisitos não funcionais e de interface**.

Os **requisitos funcionais** referem-se diretamente ao núcleo da solução, abrangendo o desenvolvimento dos modelos cinemáticos, das funções objetivo e da estrutura de otimização necessária para a síntese dos mecanismos de seis barras.

Os **requisitos não funcionais e de interface** abrangem tanto os aspectos de interação com o usuário, incluindo a apresentação clara dos resultados e a facilidade de operação mesmo por usuários sem experiência em programação, quanto os elementos estruturais do sistema, relacionados à arquitetura, manutenibilidade, reprodutibilidade e robustez numérica da aplicação.

Requisitos Funcionais

1. Implementar, em linguagem Python, os modelos cinemáticos dos mecanismos de seis barras considerados;

2. Desenvolver as funções objetivo de otimização para cada mecanismo, integrando o respectivo modelo cinemático, as restrições associadas à qualidade de transmissão e a função de erro que orienta a convergência para os ângulos de entrada e saída especificados;
3. Definir os parâmetros padrão do algoritmo *Differential Evolution*, garantindo reprodutibilidade dos experimentos;
4. Disponibilizar rotinas para exportação dos resultados relevantes (parâmetros ótimos, valores de função objetivo e métricas associadas) em formato adequado para análise posterior, como arquivos de planilha.

Requisitos Não Funcionais e de Interface

1. Desenvolver uma interface simples, clara e intuitiva, que permita ao usuário operar o sistema sem a necessidade de interpretar o código-fonte;
2. Disponibilizar três visualizações gráficas principais:
 - a) gráfico das configurações de ângulos para até cinco pontos de precisão;
 - b) painel de simulação com controle deslizante, permitindo visualizar o movimento do mecanismo ao longo do intervalo de operação;
 - c) gráfico da função gerada pelos ângulos de entrada e saída, incluindo os pontos desejados e obtidos na otimização, bem como as curvas de qualidade de transmissão;
3. Implementar uma barra de configurações que permita adicionar mais de cinco pontos de precisão e ajustar os parâmetros de otimização, tais como tamanho da população, limites das variáveis e tolerâncias numéricas;
4. Adotar programação orientada a objetos para garantir escalabilidade, modularidade e facilidade de manutenção da aplicação;
5. Garantir que o tempo de execução permaneça compatível com o uso interativo da ferramenta em estudos de caso típicos;
6. Documentar os principais componentes do sistema e seus parâmetros, de modo a facilitar sua utilização e futura extensão por outros pesquisadores.

3.3 Solução Proposta

Nesta seção, será detalhada a solução desenvolvida no projeto. Assim como na etapa de definição dos requisitos, ela está dividida em duas partes: **Requisitos Funcionais** e **Requisitos de Interface e Usabilidade**.

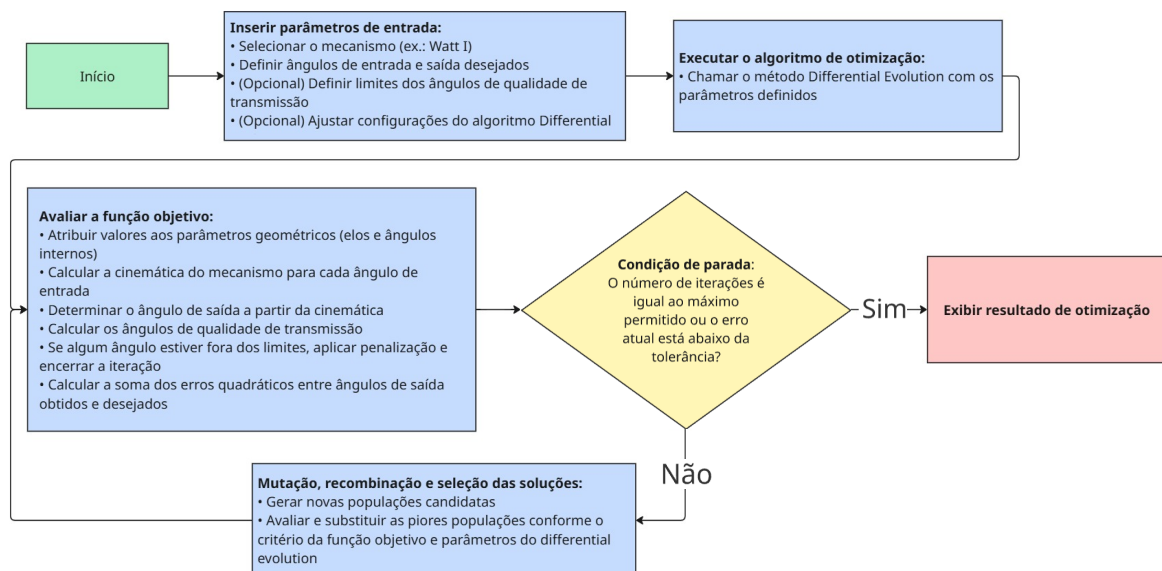
3.3.1 Requisitos Funcionais

A Figura 19 é um fluxograma que descreve as etapas funcionais do método de otimização desenvolvido. As implementações correspondentes abrangem o modelo cinemático dos mecanismos, a formulação da função objetivo, a aplicação das restrições de qualidade de transmissão e a configuração do algoritmo *Differential Evolution*.

3.3.1.1 Inserção dos Parâmetros de Entrada

Inicialmente, o usuário deve inserir os parâmetros de entrada necessários para a execução da otimização. Entre eles estão: a seleção do tipo de mecanismo (*Watt I*, *Watt II*, *Stephenson I*, *Stephenson II* ou *Stephenson III*), a definição dos ângulos de entrada e saída desejados e, opcionalmente, os limites dos ângulos de qualidade de transmissão. Também é possível ajustar as configurações do algoritmo *Differential Evolution*, como o número máximo de iterações, o tamanho da população e as tolerâncias de convergência.

Figura 19 – Fluxograma de etapa funcional



Fonte: Elaborado pelo autor

3.3.1.2 Modelagem Cinemática e Formulação da Função Objetivo

Para cada topologia de mecanismo, foi implementada uma função objetivo específica, responsável por avaliar a adequação geométrica dos parâmetros otimizados em relação ao comportamento cinemático desejado. Essa função recebe como entrada o vetor de parâmetros do mecanismo (comprimentos de elos e ângulos fixos), juntamente com os vetores de ângulos de entrada e saída desejados, retornando um valor escalar que representa o erro acumulado entre os ângulos de saída obtidos e os valores de referência.

O cálculo da função objetivo se baseia na **cinemática de posição** dos mecanismos de seis barras (*Watt* e *Stephenson*), como mostra o resumo na Tabela 1. Para cada ângulo de entrada θ_I , determina-se a configuração planar dos elos resolvendo o equacionamento geométrico do mecanismo. A partir desse modelo, obtêm-se as coordenadas cartesianas de cada junta e, conseqüentemente, os ângulos de saída θ_O , que descrevem o movimento do elo de saída.

A modelagem geométrica adotada apresenta uma formulação simplificada em relação aos métodos tradicionais de fechamento de laços vetoriais, o que permite eliminar automaticamente soluções inválidas associadas aos **defeitos de circuito**. Isso acontece pois, o método geométrico, a escolha do sinal em funções trigonométricas, como no cálculo de um arco cosseno, por exemplo, define uma única configuração geométrica válida. Assim, qualquer conjunto de parâmetros que inviabilize essa configuração resulta em um erro matemático, sendo descartado do processo de otimização.

Na implementação em Python, essa verificação é realizada por meio de um bloco `try-except`. O cálculo cinemático é tentado para cada indivíduo da população e, caso ocorra uma exceção durante a resolução geométrica (como erros de domínio em funções trigonométricas ou inconsistências numéricas), o mecanismo correspondente é penalizado na função objetivo. Esse procedimento garante que apenas configurações geometricamente viáveis contribuam para a evolução das soluções.

Com essa abordagem geral, cada topologia de mecanismo recebe um tratamento específico para a determinação das posições intermediárias: nos mecanismos *Watt I*, *Watt II*, *Stephenson I* e *Stephenson III*, as posições dos pontos intermediários são obtidas por relações trigonométricas diretas, enquanto no mecanismo *Stephenson II* é necessária a resolução de uma equação de igualdade, solucionada numericamente com o método `root` do pacote `scipy.optimize`.

A equação **erro** da função objetivo 3.1 é então definida como a soma dos desvios quadráticos entre os ângulos de saída calculados e os ângulos desejados, fornecendo ao algoritmo uma medida direta da qualidade da síntese em relação ao que é desejado pelo usuário. Essa equação formaliza o tipo de síntese adotado, caracterizado como síntese por geração de função.

A implementação em código da função objetivo genérica para os mecanismos de 6

barras pode ser vista no pseudocódigo 1. Caso o leitor tenha interesse na implementação completa de todos os mecanismos em Python, ela se encontra no repositório do GitHub: <<https://github.com/KrepsD/Six-Bar-Function-Generation-Synthesis-Tool>>

$$F_{obj} = \sum (\theta_O - \theta_{Od})^2 \quad (3.1)$$

Algorithm 1 Função Objetivo genérica para mecanismos de 6 barras

```

1: #  $p$ : vetor de parâmetros geométricos
2: #  $\theta_I$ : vetor de ângulos de entrada desejados
3: #  $\theta_{Od}$ : vetor de ângulos de saída desejados
4: #  $n$ : número de posições analisadas
5: #  $lb, rb$ : limites inferior e superior da qualidade de transmissão
6: function FUNCAO_OBJETIVO( $p, \theta_I, \theta_{Od}, n, lb, rb$ )
7:   Extrair  $L_1, L_2, L_3, L_4, L_5, L_6, L_8, L_9, \phi, \alpha, \lambda$  de  $p$ 
8:   for  $i = 1$  até  $n$  do
9:     try
10:       Calcular a cinemática do mecanismo (posições e ângulos)
11:       Obter ângulo de saída  $\theta_O$ 
12:       Calcular ângulos de qualidade de transmissão  $\mu_1$  e  $\mu_2$ 
13:       if ( $\mu_1 < lb$ ) ou ( $\mu_1 > rb$ ) ou ( $\mu_2 < lb$ ) ou ( $\mu_2 > rb$ ) then
14:         Penalizar:  $\theta_O[i] = 10^{12}$ 
15:         break ▷ Restrição não cumprida
16:       else
17:         Armazenar ângulo:  $\theta_O[i] = \theta_O$ 
18:       end if
19:     except
20:       Penalizar:  $\theta_O[i] = 10^{12}$ 
21:       break ▷ Erro geométrico ou numérico detectado
22:     end try
23:   end for
24:   Calcular soma dos erros quadráticos:
    $F_{obj} = \sum (\theta_O - \theta_{Od})^2$ 
25:   return  $F_{obj}$  ▷ Valor a ser minimizado
26: end function

```

3.3.1.3 Aplicação das Restrições de Qualidade de Transmissão

Durante a avaliação da função objetivo, são incorporadas restrições de **qualidade de transmissão**, expressas pelos ângulos μ_1 e μ_2 , que avaliam o alinhamento entre os elos e o desempenho do movimento transmitido. Valores fora do intervalo permitido resultam em penalizações na função objetivo, prevenindo configurações cinemáticas desfavoráveis e singularidades.

3.3.1.4 Execução do Algoritmo de Otimização

Com os parâmetros definidos, o método *Differential Evolution* é executado para encontrar a melhor combinação de parâmetros geométricos que satisfaça os requisitos de geração de função desejados. A implementação é realizada por meio da função *differential_evolution* da biblioteca *scipy*, que recebe a função objetivo, os limites das variáveis e os argumentos adicionais correspondentes a cada mecanismo.

Durante a execução, cada indivíduo da população representa uma possível solução. O algoritmo avalia o erro de cada conjunto de parâmetros, gera novas populações por mutação e recombinação, e substitui as piores soluções conforme o critério de desempenho estabelecido pela função objetivo. Esse processo iterativo garante a exploração eficiente do espaço de soluções e a convergência progressiva para configurações mais adequadas.

O funcionamento simplificado do algoritmo *Differential Evolution* pode ser visto no pseudocódigo 2.

Algorithm 2 Pseudocódigo simplificado do *differential_evolution*

```

1: # Busca os parâmetros geométricos  $p$  que minimizam a função objetivo
2: # Utiliza o método differential_evolution da biblioteca SciPy
3: Inicializar população aleatória de soluções  $p_j$ 
4: Definir limites de busca (bounds) e parâmetros do algoritmo (taxa de mutação, re-
   recombinação, tolerância, iterações máximas, etc)
5: while (número de iterações < máximo permitido) e (erro atual > tolerância) do
6:   for cada indivíduo  $p_j$  da população do
7:     Gerar novo candidato  $p'_j$  por mutação e recombinação
8:     Avaliar  $F_{obj}(p'_j) = \text{FUNCAO\_OBJETIVO}(p'_j, \theta_I, \theta_{Od}, n, lb, rb)$ 
9:     # Seleção: mantém o indivíduo com melhor desempenho
10:    if  $F_{obj}(p'_j) < F_{obj}(p_j)$  then
11:      Substituir  $p_j = p'_j$ 
12:    end if
13:  end for
14:  Atualizar melhor solução global  $p$  e  $F_{obj}$ 
15: end while
16: Exibir  $p$  e  $F_{obj}$ 

```

3.3.1.5 Condição de Parada e Exibição dos Resultados

O processo iterativo é mantido enquanto o número de iterações for inferior ao limite máximo e o erro atual estiver acima da tolerância especificada. Quando os critérios de parada são satisfeitos, o algoritmo encerra a otimização e exibe os parâmetros geométricos otimizados, juntamente com o valor final da função objetivo. O resultado corresponde à configuração que minimiza o erro total entre os ângulos de saída obtidos e os desejados, atendendo simultaneamente às restrições impostas.

3.3.1.6 Configurações do algoritmo

Para garantir um bom equilíbrio entre precisão, estabilidade e tempo de execução, foram adotadas as seguintes configurações para o algoritmo *DE*:

- ***num var*** = 11: Número de variáveis de decisão (ou parâmetros de projeto) utilizadas na função objetivo;
- ***bounds***: limites inferiores e superiores para os parâmetros geométricos utilizados na otimização, abrangendo tanto os comprimentos dos elos quanto os ângulos de rotação. Os elos podem variar entre 1 e 100 unidades de comprimento, enquanto os ângulos são permitidos no intervalo de 0 a 359 graus. Dentro desses limites, o algoritmo ajusta iterativamente os valores em busca da solução que minimize a função objetivo;
- ***maxiter*** = 2000: número máximo de iterações, definido para assegurar convergência consistente sem custos computacionais excessivos;
- ***popsize*** = 15: parâmetro que define o número de indivíduos por dimensão do problema. Como o modelo considerado possui 11 variáveis de decisão, o tamanho efetivo da população é dado por $15 \times 11 = 165$ indivíduos.
- ***tol*** = 1e-2 e ***atol*** = 1e-4: tolerâncias de convergência que proporcionam bom compromisso entre precisão numérica e tempo de execução;
- ***mutation*** = (0.5, 1) e ***recombination*** = 0.7: parâmetros de mutação e recombinação adotados conforme recomendações clássicas da literatura;
- ***strategy*** = 'randtobest1bin': estratégia de mutação que tende a favorecer convergência rápida, preservando diversidade suficiente para evitar mínimos locais prematuros;
- ***workers*** = -1: utilização de todos os núcleos disponíveis do processador, permitindo a execução paralela das avaliações da função objetivo.

Essas configurações oferecem robustez computacional e boa adaptabilidade para diferentes topologias de mecanismos. Quando necessário, o usuário pode ajustar manualmente cada parâmetro na aba de configurações do sistema, descrita na próxima seção.

3.3.2 Requisitos Não Funcionais e de Interface

Com base nos requisitos definidos na etapa de planejamento, a interface gráfica foi desenvolvida de modo a oferecer uma experiência intuitiva, responsiva e modular. O objetivo principal foi permitir que o usuário interaja com o programa sem a necessidade

de compreender diretamente o código-fonte, facilitando a análise dos mecanismos e a visualização dos resultados obtidos nas etapas de otimização e simulação cinemática.

A aplicação foi construída sobre a biblioteca *customtkinter*, que fornece uma camada moderna de componentes gráficos para Python, e integra elementos interativos como caixas de seleção, campos de entrada, controles deslizantes, interruptores e gráficos dinâmicos gerados com a biblioteca *matplotlib*. A arquitetura da interface segue um padrão modular, com POO, no qual cada grupo de *widgets* é criado por funções específicas, garantindo clareza e facilidade de manutenção do código.

3.3.2.1 Definição da Interface Gráfica de Usuário (GUI)

Inicialmente, a janela principal é configurada pela função construtora da classe, responsável por definir o layout e delegar a criação dos componentes visuais à função `init_ui()`. Nessa etapa, são também definidas as constantes de estilo, incluindo caminhos de imagens, cores e fontes padronizadas, o que permite a alteração global do tema da aplicação de forma centralizada.

Foram implementados **componentes interativos** que permitem a personalização e a exploração visual dos resultados:

- **Caixas seletoras:** utilizados para seleção do tipo de mecanismo (*Watt 1*, *Watt 2*, *Stephenson 1*, *2 e 3*) e para alternar entre configurações específicas de simulação, como número de casos e parâmetros de otimização. Esses elementos são armazenados em estruturas internas, permitindo fácil acesso durante as *callbacks*.
- **Campos de entrada:** recebem valores numéricos como ângulos de entrada e saída desejados, limites de otimização e número de pontos de precisão. São gerados a partir de uma lista de especificações, evitando redundância na criação dos widgets e permitindo diferentes tamanhos e posições conforme o contexto.
- **Controles deslizantes:** controlam dinamicamente o ângulo de entrada θ_I , atualizando em tempo real a posição do mecanismo visualizado. Cada *slider* está vinculado a uma função de *callback* que redesenha o mecanismo conforme o valor selecionado.
- **Interruptores:** permitem alternar exibições gráficas adicionais, como a numeração dos elos e a exibição dos ângulos ou arcos de movimento.

A interface também inclui uma seção dedicada à visualização gráfica dos resultados. A função de configuração de gráficos cria instâncias do *matplotlib* integradas à interface, possibilitando a exibição dinâmica de três tipos principais de visualização:

1. **Visualização cinemática:** exibe o mecanismo em sua configuração atual, destacando juntas, elos e ângulos calculados.

2. **Simulação interativa:** permite movimentar o mecanismo por meio de um controle deslizante, atualizando a posição dos elos em tempo real.
3. **Gráficos analíticos:** apresentam as curvas de variação dos ângulos de entrada e saída, bem como as curvas de qualidade de transmissão (μ_1 e μ_2), permitindo avaliar o comportamento global do mecanismo ao longo do ciclo de movimento.

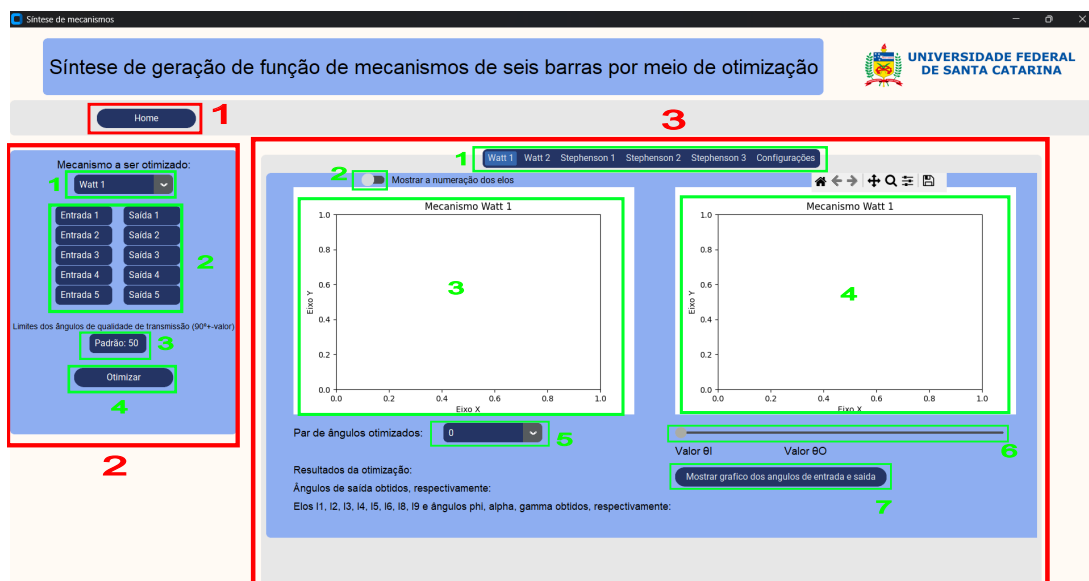
Adicionalmente, foi implementada uma **barra de configurações** que possibilita ao usuário ajustar parâmetros de otimização, como tamanho da população, limites e tolerâncias, além de permitir a inclusão de mais de cinco pontos de precisão. A interface também possui um mecanismo de reinicialização (*Home*), responsável por reconstruir todos os widgets e restaurar o estado inicial da aplicação.

Com os detalhes da implementação da interface descritos, serão apresentadas figuras das telas da interface, junto com a explicação de cada elemento nelas. Na Figura 20, é apresentada a **tela principal da aplicação**, composta por diferentes áreas funcionais que organizam as etapas de entrada de dados, visualização e execução da otimização. A interface foi estruturada em seções modulares, destacadas na imagem, que permitem ao usuário compreender rapidamente o fluxo de uso do sistema.

Módulo 1 – Cabeçalho e Navegação (Região superior em vermelho)

O primeiro módulo compreende o cabeçalho da aplicação, responsável pela identidade visual e pela navegação principal.

Figura 20 – Tela principal da Interface



Fonte: Elaborado pelo autor

- **(1) Botão Home:** posicionado no canto superior esquerdo, o botão *Home* reinicializa a aplicação, reconstruindo todos os componentes da interface e restaurando o estado padrão do programa. Essa função é essencial para o início de novas simulações ou otimizações, evitando a interferência de dados anteriores.

Módulo 2 – Painel Lateral de Controle (Região esquerda em vermelho)

O segundo módulo reúne todos os controles de entrada de dados relacionados ao mecanismo e aos parâmetros de otimização. Sua disposição vertical foi planejada para guiar o usuário de forma sequencial, da seleção do mecanismo até a execução da otimização.

- **(1) Seletor de mecanismo:** caixa de seleção do tipo *combobox* que permite escolher a topologia a ser otimizada. A escolha define automaticamente quais parâmetros geométricos e configurações internas serão utilizados pelo algoritmo.
- **(2) Campos de ângulos de entrada e saída:** organizados em pares, permitem inserir até cinco combinações de ângulos desejados. Esses valores definem a função cinemática que o mecanismo deve reproduzir.
- **(3) Campo de limite de qualidade de transmissão:** define o desvio permitido em torno de 90° para os ângulos de qualidade de transmissão, permitindo ajustes de acordo com a aplicação.
- **(4) Botão de otimização:** inicia o processo de otimização diferencial, coletando os dados inseridos e enviando-os ao núcleo computacional. Durante a execução, a interface é atualizada dinamicamente para indicar o progresso e, ao final, exibe os resultados obtidos.

Esse módulo constitui o painel de interação direta com o usuário, centralizando as ações de configuração e controle do processo de otimização.

Módulo 3 – Área de Visualização e Resultados (Região central em vermelho)

O terceiro módulo corresponde à região central e direita da interface, dedicada à análise visual e numérica dos mecanismos. Ele apresenta os resultados de forma gráfica e interativa, permitindo a inspeção detalhada do comportamento cinemático.

- **(1) Abas da interface:** permitem alternar entre as diferentes seções do programa. Ao serem clicadas, exibem o conteúdo correspondente a cada mecanismo de seis barras (*Watt 1*, *Watt 2*, *Stephenson 1*, *Stephenson 2*, *Stephenson 3*) e uma aba adicional dedicada às configurações gerais da aplicação. As abas destinadas aos diferentes mecanismos apresentam a mesma estrutura funcional e disposição de componentes, variando apenas quanto ao tipo de mecanismo selecionado.

- **(2) Interruptor de exibição:** ativa ou desativa a numeração dos elos diretamente sobre o gráfico, facilitando a identificação das juntas e elos durante a simulação.
- **(3) Gráfico do mecanismo:** exibe a configuração planar de cada um dos pares de ângulos otimizados do mecanismo otimizado, representando os elos e juntas.
- **(4) Gráfico de simulação e análise:** apresenta o gráfico com a simulação do movimento do mecanismo e as curvas de desempenho, como ângulos de entrada e saída, e qualidade de transmissão (μ_1, μ_2). Essas visualizações auxiliam na avaliação quantitativa da qualidade da solução obtida.
- **(5) Seletor de par de ângulos otimizados:** possibilita alternar entre diferentes soluções válidas encontradas pelo algoritmo de otimização, facilitando comparações entre configurações distintas do mesmo mecanismo.
- **(6) Controle deslizante do ângulo de entrada:** permite variar o valor de θ_I e observar o movimento do mecanismo em tempo real. Cada alteração aciona uma função de atualização que redesenha a estrutura no gráfico (4).
- **(7) Botão para mostrar gráficos de relações de ângulos:** botão que, quando pressionado, alterna o gráfico (4) para mostrar as relações de ângulos de entrada, saída e de qualidade de transmissão.

Na Figura 21 é possível visualizar a aba de configurações, onde é possível submeter mais de cinco pares de ângulos de precisão e mudar os parâmetros da função de otimização.

Figura 21 – Tela de configuração

Fonte: Elaborado pelo autor

3.3.3 Integração da GUI com a Aplicação Funcional

A comunicação entre a interface gráfica e o núcleo computacional é realizada por métodos de controle da classe App, como pode ser visto na UML do programa (feito com PlantUML) na Figura 22 responsáveis por interpretar as ações do usuário e coordenar a execução dos algoritmos de síntese. Esses métodos coletam os parâmetros inseridos na interface (elos, limites angulares, configuração do mecanismo, etc.), instanciam o modelo cinemático apropriado (Watt I, Watt II, Stephenson I, II ou III) e iniciam o processo de otimização. Durante a otimização, a interface permanece sincronizada com o núcleo computacional, exibindo as atualizações de ângulos, gráficos e status da execução.

Outro aspecto importante é que o *software* permite a sintetização e armazenamento simultâneo de até cinco mecanismos diferentes, um para cada topologia disponível (Watt I, Watt II, Stephenson I, II e III). Cada mecanismo é tratado como uma instância independente da classe abstrata Mecanismo, conforme a Figura 22, mantendo seus próprios resultados até que uma nova otimização seja solicitada. Essa arquitetura possibilita a comparação direta das diferentes topologias de mecanismos de seis barras dentro de uma mesma aplicação, permitindo ao usuário identificar de forma eficiente qual configuração atende melhor aos requisitos cinemáticos desejados.

Ao final do processo, além de apresentar os resultados na interface, o sistema gera automaticamente um arquivo .csv na mesma pasta do código. Esse arquivo contém todas as informações relevantes: ângulos de entrada e saída (desejados e resultantes), índices de qualidade de transmissão, parâmetros geométricos do mecanismo e o tempo de execução do algoritmo. Assim, estabelece-se uma conexão contínua entre a camada visual e o modelo matemático, permitindo uma análise integrada e interativa dos resultados obtidos.

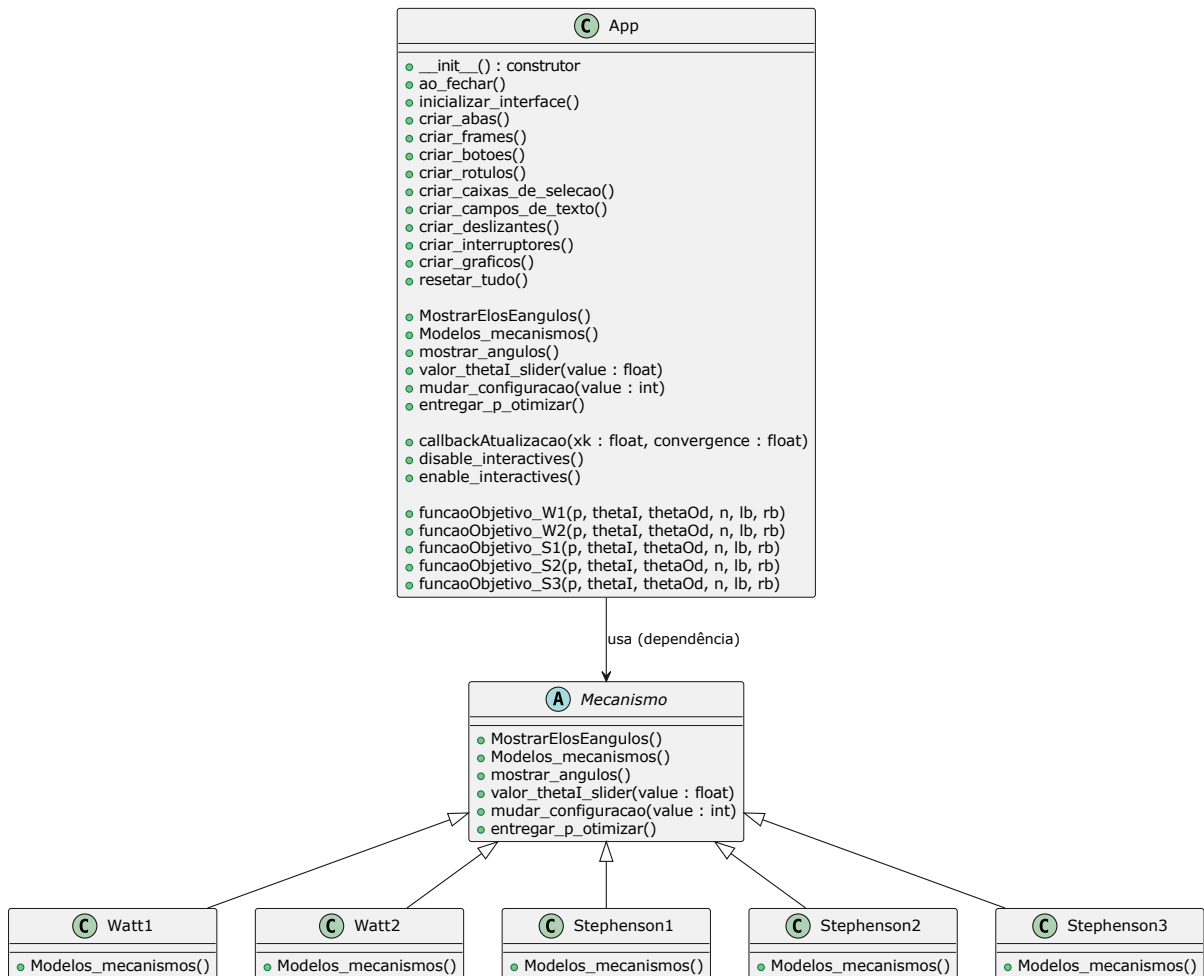
3.3.4 Funcionamento Completo

Para concluir a seção de metodologia, apresenta-se o funcionamento completo do programa, desde sua inicialização até a visualização dos gráficos e simulações gerados, bem como a exportação dos dados referentes ao mecanismo e ao processo de otimização.

Ao iniciar o programa, o usuário é direcionado para a tela principal, conforme ilustrado na Figura 20. Nessa interface, o usuário deve selecionar um dos cinco mecanismos disponíveis para otimização e inserir os pares de ângulos de precisão. Na tela inicial, é possível definir entre dois e cinco pares de ângulos, como mostrado na Figura 23. Opcionalmente, o usuário pode também especificar um valor máximo para os ângulos de qualidade de transmissão. Caso seja necessário inserir mais pares de ângulos ou alterar algum parâmetro de configuração, é possível acessar a tela de configurações, conforme apresentado na Figura 21.

Após inserir as informações necessárias, o usuário pode acionar o botão “Otimizar”, conforme ilustrado na Figura 23. Ao fazer isso, o programa inicia o processo de otimiza-

Figura 22 – UML do programa



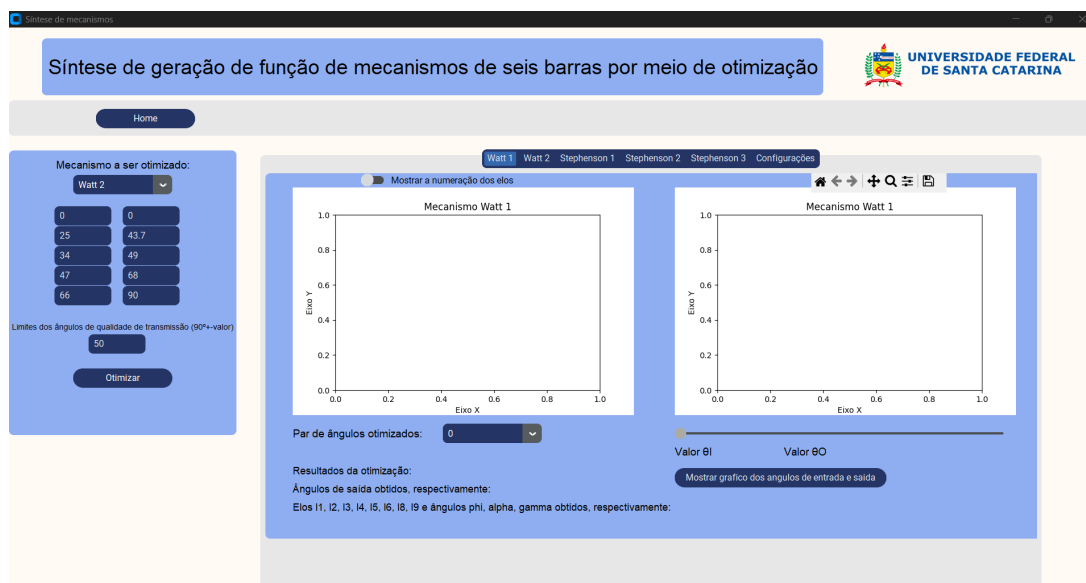
Fonte: Elaborado pelo autor

ção. Caso algum dado esteja incorreto ou inviabilize o cálculo, uma mensagem de erro é exibida abaixo do botão, orientando o usuário a revisar as entradas. Durante a execução da otimização, todos os campos de entrada e controles deslizantes são temporariamente desativados, evitando alterações que possam interferir no processamento.

Quando a otimização é concluída com sucesso, uma mensagem de confirmação é exibida abaixo do botão “Otimizar”. Em seguida, são gerados os gráficos de análise do mecanismo, e um arquivo no formato *.csv* é exportado contendo os principais resultados referentes ao mecanismo e ao processo de otimização. As Figuras 24 e 25 apresentam os gráficos de ordenação dos pares de ângulos, a simulação do mecanismo e o relacionamento entre ângulos de entrada e saída, desejados, obtidos e os ângulos de qualidade de transmissão. A Figura 26 ilustra o arquivo de dados exportado no formato *.csv*.

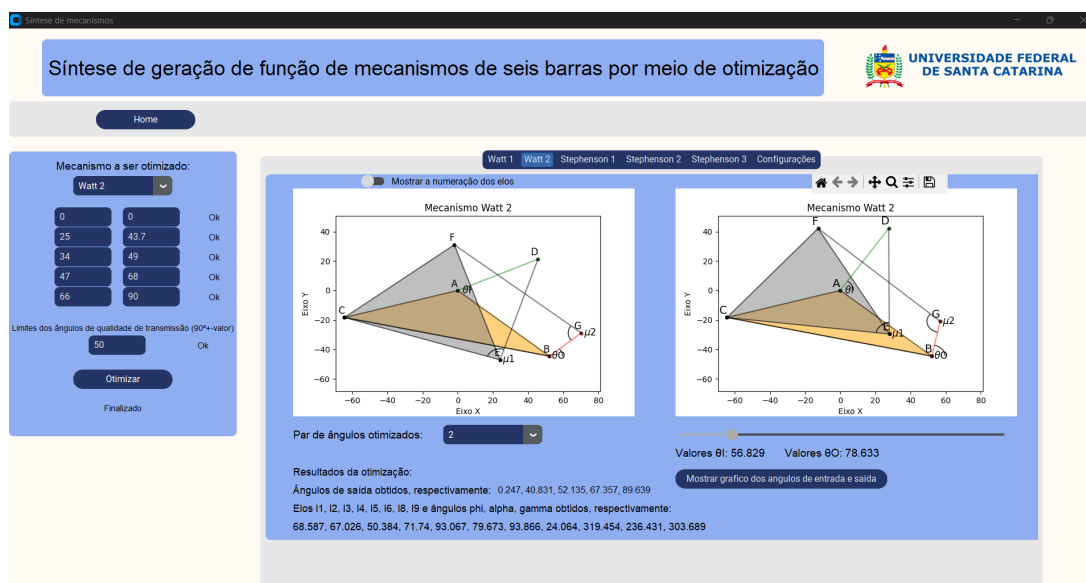
Os dados são exportados em formato tabular porque, como o programa permite a definição de um número personalizado de pontos de precisão, valores muito elevados podem inviabilizar a visualização completa das informações diretamente na interface. A exportação para um arquivo estruturado, portanto, torna-se uma forma organizada e eficiente

Figura 23 – Tela inicial com dados inseridos



Fonte: Elaborado pelo autor

Figura 24 – Primeira tela de gráficos



Fonte: Elaborado pelo autor

de acessar todos os resultados. Além disso, esse recurso é fundamental para registrar mecanismos que o usuário considere relevantes, permitindo sua reconstrução posterior em um programa CAD ou a realização de novas buscas por mecanismos da mesma topologia que possam atender de maneira ainda mais adequada ao problema em estudo.

Com o processo finalizado, o usuário pode aplicar os mesmos parâmetros aos demais mecanismos disponíveis, comparando-os para identificar aquele que melhor se adequa à aplicação desejada, considerando fatores como ângulos de qualidade de transmissão, configuração do mecanismo e espaço de trabalho. Caso não seja possível obter uma

Figura 25 – Segunda tela de gráficos

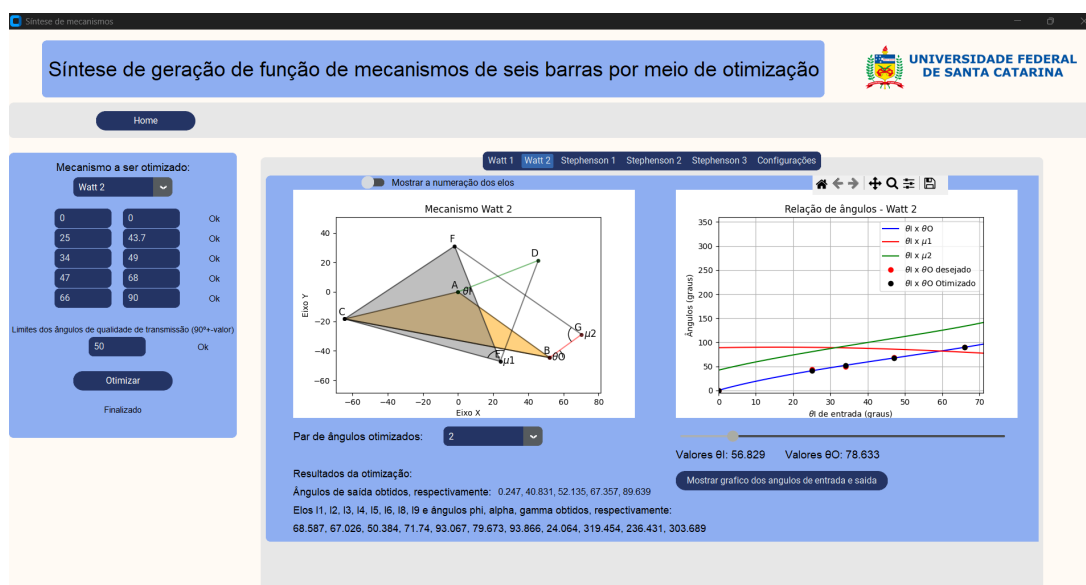


Figura 26 – Tabela com os dados exportados

L1	L2	L3	L4	L5	L6	ϕ	α	λ	μ_1	μ_2	θ_i	θ_{Od}	θ_O	tempo otimizando		
72,31	29,5	68,06	78,54	73,22	78,75	91,45	43,77	336,06	232,64	281,11	[77.42, 79.88,	[40.15, 73.31,	[0.0, 25.0,	[0.0, 43.7,	[0.28, 40.87,	13,92

Fonte: Elaborado pelo autor

solução satisfatória, o usuário pode selecionar o botão “Home” para limpar todos os dados inseridos e reiniciar o processo, testando novos parâmetros de entrada para os mecanismos ou para o otimizador. O vídeo publicado no YouTube a seguir mostra o funcionamento das principais funcionalidades do *software*: <<https://www.youtube.com/watch?v=6VuMKqHRDXE>>.

4 Resultados

Nesta seção são apresentados os resultados obtidos com o programa de otimização, destacando tanto os pontos positivos quanto as limitações observadas. Para melhor organização, os resultados foram agrupados em duas categorias: testes e validação e estudos de caso.

Na etapa de testes e validação, são mostrados os resultados das otimizações realizadas, de cada um dos mecanismos, incluindo: parâmetros geométricos obtidos, imagem do mecanismo, comparação entre os ângulos de entrada e saída desejados e os valores obtidos, além dos ângulos de qualidade de transmissão e do erro absoluto entre o ângulo de saída desejado e o calculado, considerando uma função previamente definida.

Na etapa de estudos de caso, são analisadas duas aplicações reais de geração de função, nas quais o programa é utilizado para solucionar o problema proposto.

4.1 Testes e Validação

Para a realização dos testes, foi necessário selecionar duas funções a serem utilizadas como referência no processo de geração de função dos mecanismos. A primeira função escolhida foi a função logarítmica, amplamente adotada em estudos de validação nessa área e já utilizada em trabalhos anteriores, como em (PLECNIK; MCCARTHY, 2014). A equação logarítmica apresentada na Equação 4.1 foi escalada em 100 vezes, de modo a representar adequadamente valores na ordem de grandeza de graus. A Figura 27 ilustra o comportamento da função após o escalonamento.

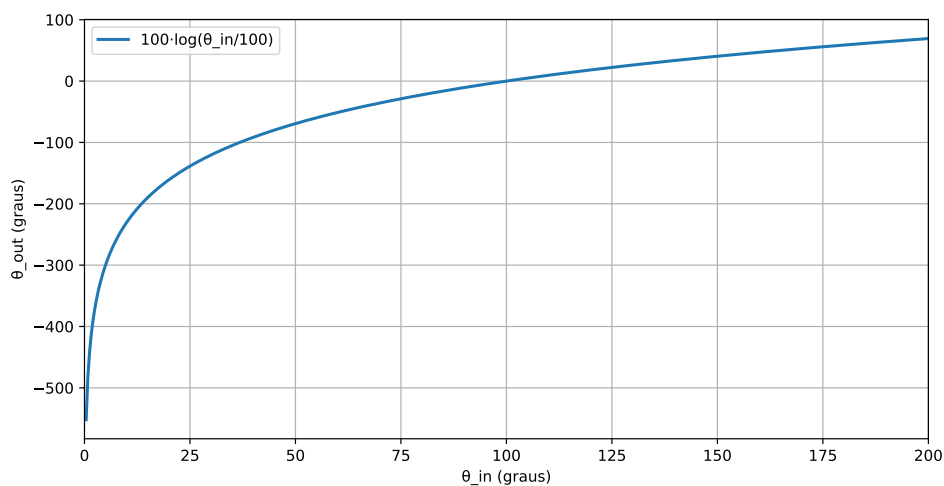
$$f(\theta_{out}) = 100 \cdot \ln \left(\frac{\theta_{in}}{100} \right) \quad (4.1)$$

A segunda função selecionada foi desenvolvida especificamente para este trabalho, utilizando uma série de Fourier trigonométrica, conforme a Equação 4.2, com os coeficientes apresentados na Tabela 2. A Figura 28 ilustra o comportamento da função resultante (expressa em graus), cuja variação foi limitada ao intervalo de 0° a 240° para utilização nos testes.

$$f(\theta_{in}) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos(n\omega\theta_{in}) + b_n \sin(n\omega\theta_{in})) \quad \text{com} \quad \omega = \frac{2\pi}{5} \quad (4.2)$$

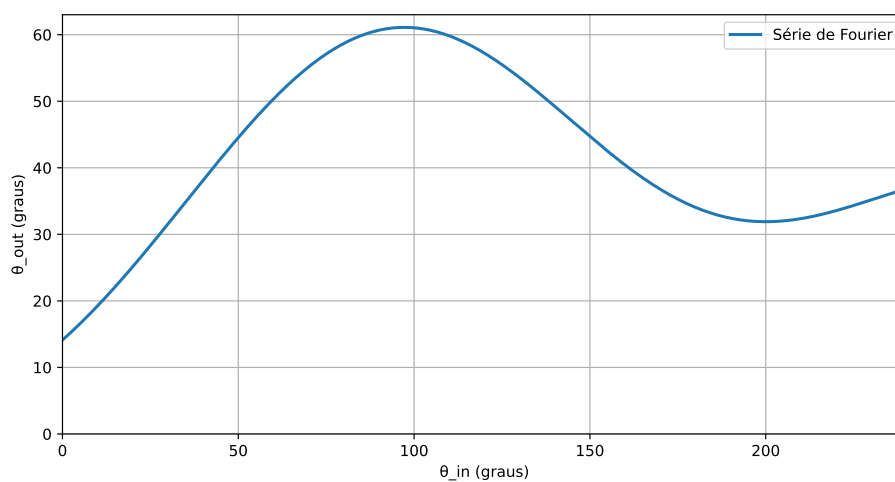
Para cada função estudada, serão realizados testes com diferentes quantidades de pontos de precisão. Para a função logarítmica, a otimização será conduzida utilizando cinco pontos de precisão, enquanto para a função definida por série de Fourier serão considerados vinte pontos. Em cada teste, os cinco mecanismos analisados serão avaliados quanto

Figura 27 – Gráfico da função logarítmica 4.1



Fonte: Elaborado pelo autor

Figura 28 – Gráfico da função dada pela série de fourier 4.2



Fonte: Elaborado pelo autor

Tabela 2 – Coeficientes da série de Fourier de $f(\theta_{in})$

Harmônico n	a_n	b_n
0	1.0941492537	—
1	-0.15746786420	0.31831985387
2	-0.11809710263	0.20410666371
3	-0.02511624061	-0.04259180001

aos parâmetros geométricos obtidos, à configuração final do mecanismo sintetizado, à comparação entre os ângulos de entrada e saída desejados e os valores calculados, aos ângulos de qualidade de transmissão e ao erro absoluto entre o ângulo de saída desejado e o obtido. Ao final testes, são apresentadas observações e análises adicionais, discutindo os resultados obtidos e destacando os principais comportamentos identificados durante o processo de otimização.

Todos os testes utilizam os parâmetros de otimização definidos como padrão no capítulo de Metodologia. Da mesma forma, os limites do ângulo de qualidade de transmissão adotados seguem o intervalo padrão de $90^\circ \pm 50^\circ$.

Observação: Todos os pontos de precisão serão determinados utilizando a função de geração de pontos de Chebyshev (Equação 4.3), descrita em Stewart (1996). Esse método distribui os pontos de forma não uniforme, concentrando-os nas extremidades do intervalo, o que melhora a aproximação de funções não lineares e reduz oscilações indesejadas na interpolação.

Além disso, nas representações gráficas dos mecanismos obtidos, o elo fixo é sempre definido como aquele compreendido entre as juntas A e B, no caso de elos binários, e entre as juntas A, B e C, no caso de elos ternários. O elo de entrada é indicado em verde, enquanto o elo de saída é indicado em vermelho.

$$t_i = \frac{a+b}{2} + \frac{b-a}{2} \cos\left(\frac{(2i-1)\pi}{2N}\right), \quad i = 1, 2, \dots, N \quad (4.3)$$

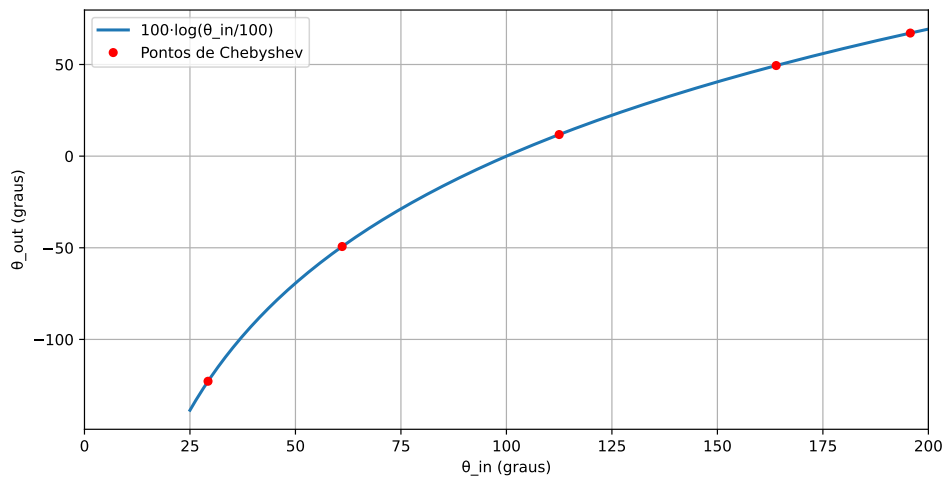
4.1.1 Teste da Função Logarítmica

Para o teste utilizando a função logarítmica, foram definidos cinco pontos de precisão. Os pares de ângulos de entrada e saída empregados na otimização estão apresentados na Tabela 3 e podem ser visualizados no gráfico da função na Figura 29. Os resultados referentes ao ângulo de saída desejado, ao ângulo obtido e ao erro absoluto para cada mecanismo estão reunidos na Tabela 4. Já os parâmetros geométricos otimizados, juntamente com o tempo total de execução da otimização, são apresentados na Tabela 5. Por fim, as Figuras 30, 31, 32, 33 e 34 ilustram o mecanismo resultante para cada caso, além dos gráficos comparativos das relações entre os ângulos relevantes.

Tabela 3 – Ângulos de entrada e saída da otimização

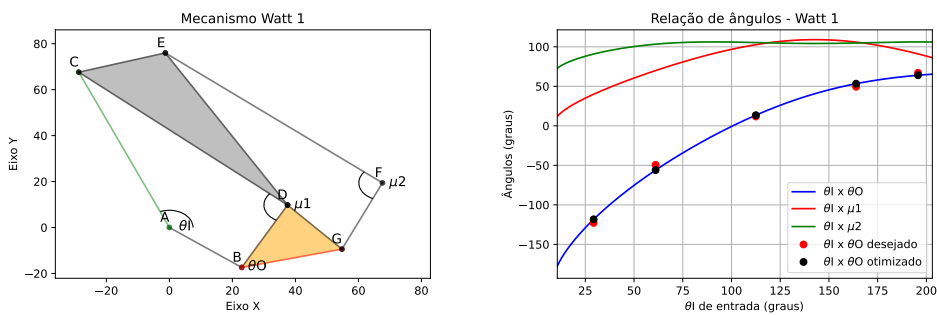
Índice	Ângulos de entrada θ_{I_n} (°)	Ângulos de saída θ_{O_n} (°)
1	29.282	-122.817
2	61.068	-49.316
3	112.500	11.778
4	163.931	49.427
5	195.717	67.150

Figura 29 – Gráfico da função log com os pontos de precisão



Fonte: Elaborado pelo autor

Figura 30 – Watt I para 5 pontos de precisão da função ln 4.1



Fonte: Elaborado pelo autor

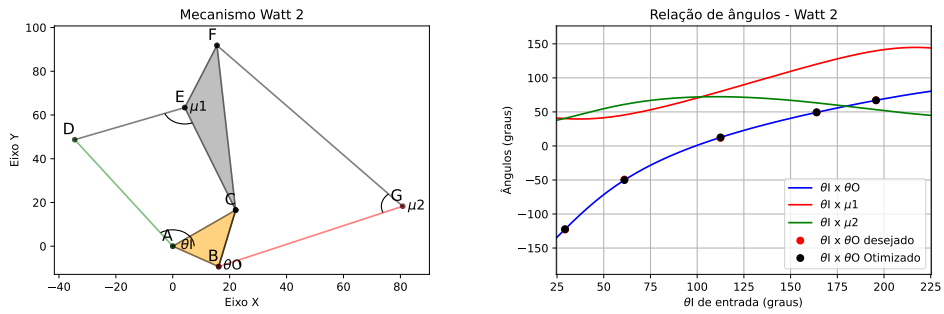
Tabela 4 – Ângulos de entrada, ângulos de saída desejados, obtidos e erro absoluto (em graus)

Mecanismo	Ângulo de entrada (°)	Ângulo de saída desejado (°)	Ângulo de saída obtido (°)	Erro absoluto (°)
Watt I	29.28	-122.82	-118.29	4.53
	61.07	-49.32	-56.03	6.71
	112.50	11.78	13.48	1.70
	163.93	49.43	53.33	3.90
	195.72	67.15	64.01	3.14
Watt II	29.28	-122.82	-122.32	0.50
	61.07	-49.32	-50.13	0.81
	112.50	11.78	12.58	0.80
	163.93	49.43	49.33	0.10
	195.72	67.15	67.06	0.09
Stephenson I	29.28	-122.82	-118.59	4.23
	61.07	-49.32	-56.25	6.93
	112.50	11.78	13.35	1.57
	163.93	49.43	53.19	3.76
	195.72	67.15	63.84	3.31
Stephenson II	29.28	-122.82	-122.88	0.06
	61.07	-49.32	-49.30	0.02
	112.50	11.78	11.73	0.05
	163.93	49.43	49.35	0.08
	195.72	67.15	67.15	0.00
Stephenson III	29.28	-122.82	-122.78	0.04
	61.07	-49.32	-49.53	0.21
	112.50	11.78	12.27	0.49
	163.93	49.43	49.06	0.37
	195.72	67.15	67.24	0.09

Tabela 5 – Parâmetros geométricos otimizados e tempo de otimização para a função ln

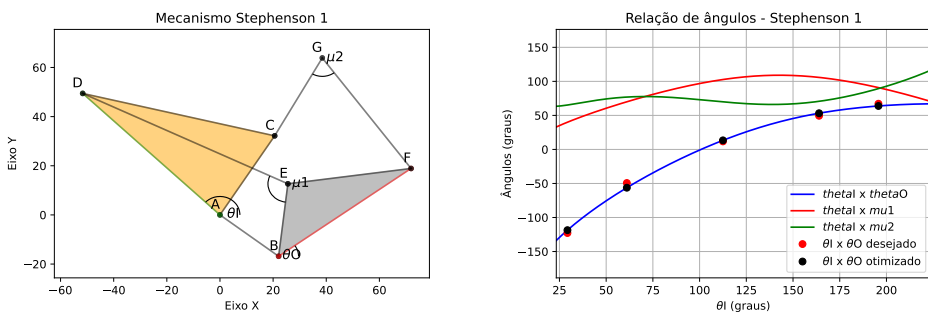
Mecanismo	L ₁	L ₂	L ₃	L ₄	L ₅	L ₆	L ₈	L ₉	ϕ	α	λ
Watt I	28.79	73.48	87.96	30.81	28.73	32.8	89.16	31.58	322.88	58.07	47.76
	Tempo de otimização: 4.26 s										
Watt II	18.62	27.60	59.68	41.37	50.16	75.58	98.34	70.32	329.97	66.82	15.9
	Tempo de otimização: 10.3 s										
Stephenson I	27.76	38.20	71.52	29.70	85.55	61.34	36.42	56.03	322.67	281.12	47.63
	Tempo de otimização: 4.86 s										
Stephenson II	19.76	14.37	63.27	47.33	81.03	73.80	72.09	46.53	86.81	278.67	82.67
	Tempo de otimização: 13.2 s										
Stephenson III	59.01	24.28	65.25	77.29	49.36	39.47	67.48	64.76	302.91	40.36	167.63
	Tempo de otimização: 8.89 s										

Figura 31 – Watt II para 5 pontos de precisão da função logarítmica mostrada na Eq. 4.1



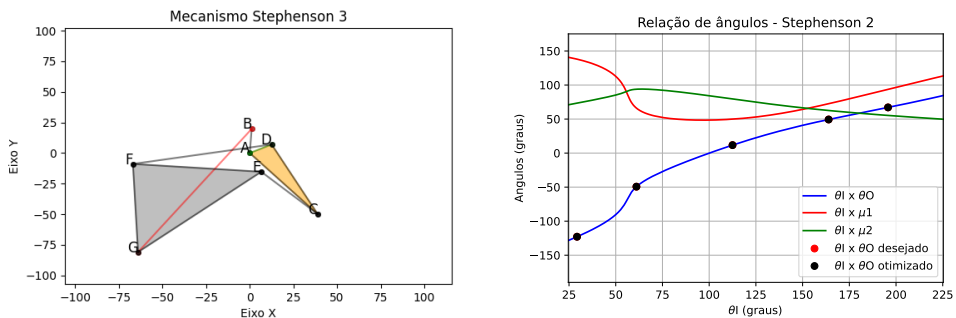
Fonte: Elaborado pelo autor

Figura 32 – Stephenson I para 5 pontos de precisão da função ln 4.1



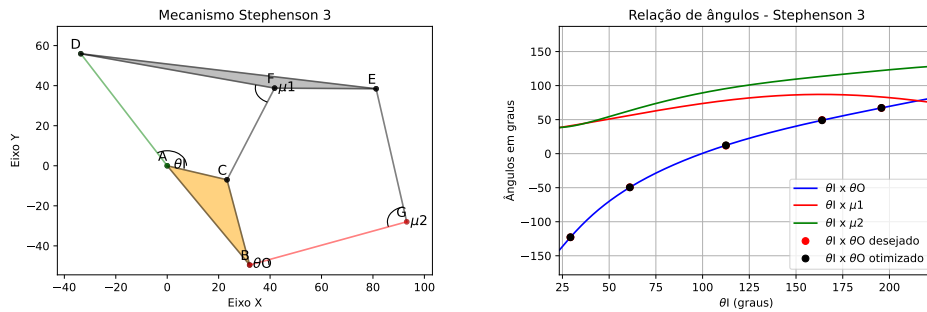
Fonte: Elaborado pelo autor

Figura 33 – Stephenson II para 5 pontos de precisão da função ln 4.1



Fonte: Elaborado pelo autor

Figura 34 – Stephenson III para 5 pontos de precisão da função ln 4.1



Fonte: Elaborado pelo autor

4.1.2 Teste da Função de Série de Fourier

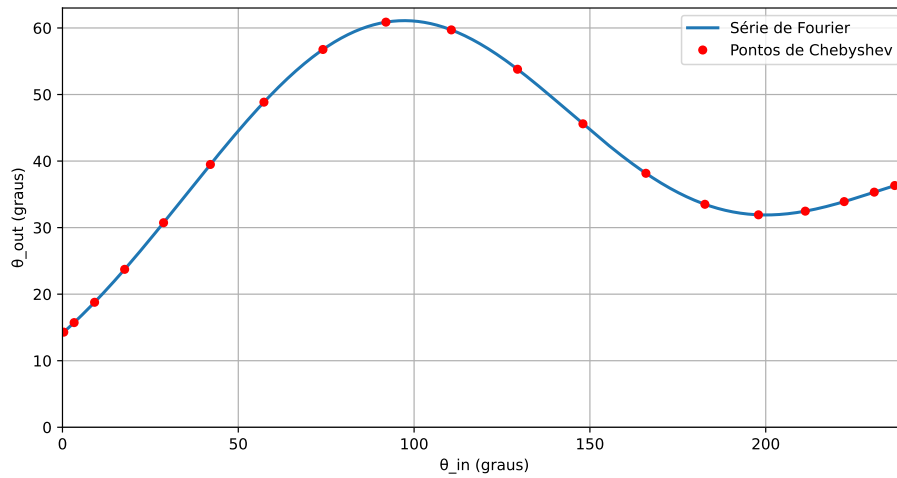
Para o teste com 20 pontos de precisão, foram utilizados os pares de ângulos de entrada e saída apresentados na Tabela 6. A distribuição desses pontos e o comportamento da função podem ser visualizados no gráfico da Figura 35. Após a execução do processo de otimização, foram obtidos, para cada mecanismo, os valores do ângulo de saída calculado, do ângulo desejado e o erro absoluto entre ambos. Como o conjunto de pontos é extenso, apresenta-se apenas o erro absoluto médio para cada mecanismo, conforme a Tabela 7.

Os parâmetros geométricos resultantes da otimização, juntamente com os respectivos tempos de processamento, são apresentados na Tabela 8. Por fim, as Figuras 36, 37, 38, 39 e 40 ilustram as configurações finais dos mecanismos otimizados, bem como os gráficos comparativos entre os ângulos de entrada e saída.

Tabela 6 – Ângulos de entrada e saída da otimização com 20 pontos

Índice	Ângulo de entrada θ_{I_n} (°)	Ângulo de saída θ_{O_n} (°)
1	0.370	14.292
2	3.316	15.726
3	9.134	18.780
4	17.683	23.730
5	28.751	30.736
6	42.066	39.490
7	57.300	48.851
8	74.078	56.769
9	91.987	60.877
10	110.585	59.712
11	129.415	53.798
12	148.013	45.599
13	165.922	38.163
14	182.700	33.506
15	197.934	31.923
16	211.249	32.473
17	222.317	33.908
18	230.866	35.329
19	236.684	36.324
20	239.630	36.810

Figura 35 – Gráfico da função de Série de Fourier com os pontos de precisão



Fonte: Elaborado pelo autor

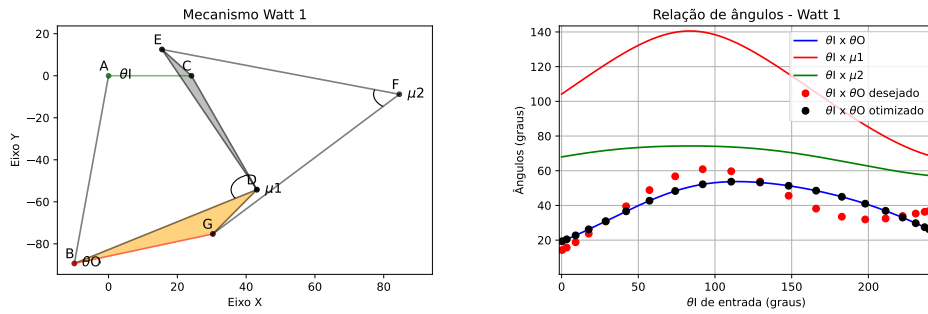
Tabela 7 – Erro absoluto médio do ângulo de saída (em graus)

Mecanismo	Erro absoluto médio (°)
Watt I	5.807
Watt II	1.814
Stephenson I	6.383
Stephenson II	1.175
Stephenson III	0.479

Tabela 8 – Parâmetros geométricos otimizados e tempo para a função da série de Fourier

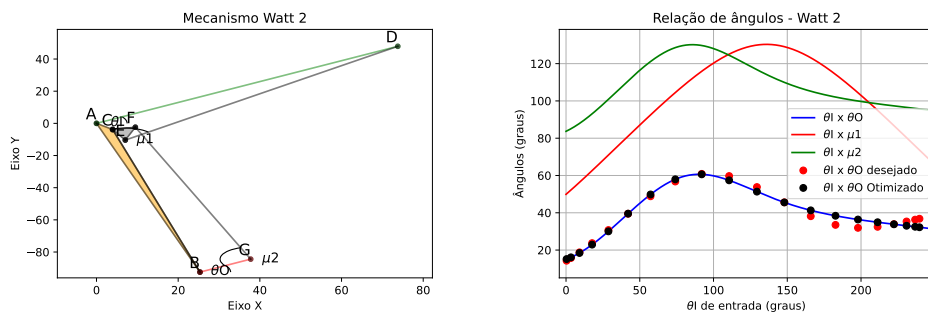
Mecanismo	L_1	L_2	L_3	L_4	L_5	L_6	L_8	L_9	ϕ	α	λ
Watt I	89.88	24.11	57.44	63.64	15.16	42.63	72.23	85.79	263.65	195.08	14.29
	Tempo de otimização: 4.34 s										
Watt II	95.9	5.49	87.97	88.6	7.26	5.74	86.74	14.81	285.32	30.63	281.06
	Tempo de otimização: 12.89 s										
Stephenson I	91.87	8.71	24.42	65.58	57.94	23.72	36.98	53.56	263.02	293.40	14.09
	Tempo de otimização: 4.48 s										
Stephenson II	38.08	7.98	20.01	48.83	78.56	86.74	46.64	21.36	46.02	286.86	18.50
	Tempo de otimização: 109.41 s										
Stephenson III	30.03	99.87	34.09	52.58	89.19	90.13	99.47	59.44	217.74	138.38	82.41
	Tempo de otimização: 15.46 s										

Figura 36 – Watt I para 20 pontos de precisão da função da série de Fourier 4.2



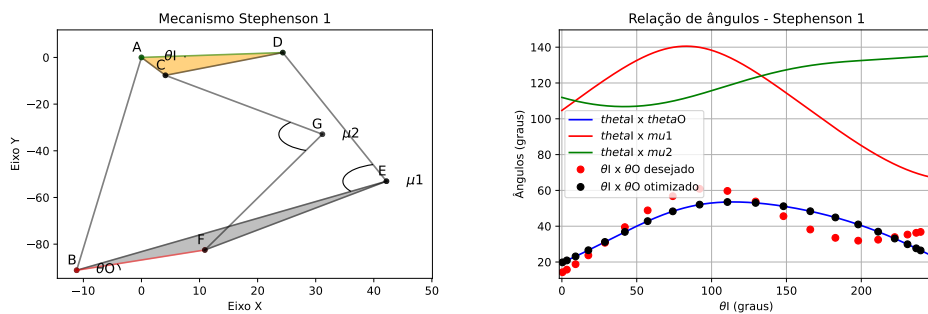
Fonte: Elaborado pelo autor

Figura 37 – Watt II para 20 pontos de precisão da função da série de Fourier 4.2



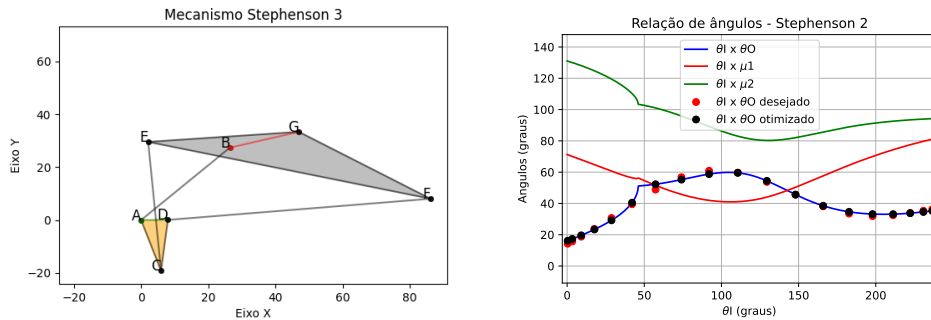
Fonte: Elaborado pelo autor

Figura 38 – Stephenson I para 20 pontos de precisão da função da série de Fourier 4.2



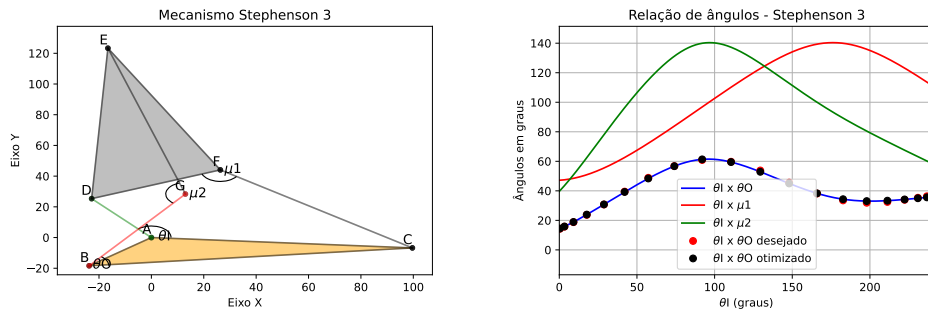
Fonte: Elaborado pelo autor

Figura 39 – Stephenson II para 20 pontos de precisão da função da série de Fourier 4.2



Fonte: Elaborado pelo autor

Figura 40 – Stephenson III para 20 pontos de precisão da função da série de Fourier 4.2



Fonte: Elaborado pelo autor

4.1.3 Discussão dos Testes

Com base nos resultados apresentados na seção anterior, esta seção tem como objetivo discutir o desempenho dos mecanismos analisados nos testes de validação. A partir dos dados obtidos, busca-se identificar tendências, pontos fortes e limitações de cada configuração, considerando aspectos como a precisão na geração do movimento, o tempo de otimização e a robustez geométrica das soluções encontradas.

De modo geral, os mecanismos apresentaram desempenho satisfatório quanto à precisão no atendimento dos pontos de precisão, mantendo-se dentro da faixa ideal dos ângulos de qualidade de transmissão. No entanto, ao observar o comportamento visual dos gráficos, especialmente no segundo teste, referente à função da série de Fourier (4.2), nota-se que os mecanismos Watt I e Stephenson I apresentaram menor fidelidade à curva desejada.

Em ambos os testes, observa-se tanto nas imagens dos resultados quanto na Tabela 3 que os mecanismos Watt II, Stephenson II e Stephenson III apresentaram desempenhos significativamente superiores aos mecanismos Watt I e Stephenson I, tanto em termos de semelhança das curvas quanto em precisão dos pontos. Essa diferença é coerente com a natureza geométrica de cada mecanismo. Conforme ilustrado na Figura 11, os mecanismos Watt I e Stephenson I possuem uma topologia essencialmente equivalente à de um mecanismo de quatro barras quando se considera a síntese por geração de função,

isto é, a relação direta entre as barras de entrada e de saída.

Assim, a síntese de geração de função para esses mecanismos equivale, na prática, à de um mecanismo de quatro barras acrescido de elos adicionais, mas com uma segunda restrição de qualidade de transmissão. Por esse motivo, Watt I e Stephenson I tendem a apresentar menor desempenho na reprodução de curvas complexas, pois possuem uma geometria menos flexível. Esse comportamento é coerente com o relatado por Plecnik e McCarthy (2014), que destaca os mecanismos Watt II, Stephenson II e Stephenson III como os mais adequados para síntese de geração de função. Ainda assim, há utilidade prática na aplicação dos mecanismos Watt I e Stephenson I. No capítulo seguinte, um desses mecanismos será sintetizado em um estudo de caso.

Um aspecto negativo observado nos resultados refere-se ao mecanismo Stephenson II. No segundo teste, nota-se uma descontinuidade no gráfico de ângulos da Figura 39, próxima de 50° do ângulo de entrada. Conforme discutido nas seções de Revisão de Literatura e Metodologia, esse mecanismo exige a resolução de uma equação de igualdade para garantir que todos os elos mantenham comprimento constante. Em alguns casos, essa restrição pode gerar soluções numéricas distintas para ângulos consecutivos, resultando em uma aparente descontinuidade no gráfico. Ressalta-se, contudo, que tal comportamento não ocorreria em uma montagem física do mecanismo, onde o movimento seria contínuo.

Outro ponto relevante é o desempenho do mecanismo Watt II no segundo teste. Embora tenha apresentado erro absoluto médio reduzido e boa correspondência com a curva desejada, sua geometria não é totalmente viável, devido às proporções excessivamente desiguais entre os elos, como evidenciado na Figura 37 e na Tabela 8. Para obter uma configuração mais realista, seria necessário impor restrições de proporcionalidade aos comprimentos dos elos, limitando a variação relativa entre eles.

Quanto aos tempos de processamento apresentados nas Tabelas 5 e 8, o algoritmo demonstrou bom desempenho computacional, com tempos de execução da ordem de dezenas de segundos para a maioria dos mecanismos. A única exceção foi novamente o Stephenson II no caso de 20 pontos de precisão, cujo tempo de otimização foi superior devido à necessidade de resolver a equação de igualdade adicional dentro da função objetivo.

4.2 Estudos de Caso

Com base nos resultados obtidos na etapa de validação e nas análises comparativas entre os diferentes mecanismos, esta seção apresenta dois estudos de caso que visam demonstrar a aplicabilidade prática das metodologias de síntese e otimização desenvolvidas ao longo deste trabalho. O objetivo é avaliar como os mecanismos otimizados podem ser adaptados para reproduzir movimentos biomecânicos e funcionais, bem como para atender a requisitos de engenharia voltados a dispositivos de assistência e reabilitação.

O primeiro estudo de caso aborda a síntese de mecanismos para representar o movimento das juntas do quadril, joelho e tornozelo humano, explorando a capacidade dos modelos em gerar trajetórias compatíveis com o movimento natural das articulações. Já o segundo estudo de caso trata da síntese de mecanismos aplicados ao desenvolvimento de uma prótese de mão, destacando a importância da precisão na geração de movimento e da adequação geométrica do sistema.

Em ambos os casos, são aplicadas as mesmas estratégias de otimização descritas anteriormente, com ajustes nos parâmetros geométricos e nas restrições cinemáticas conforme as exigências de cada aplicação. Dessa forma, busca-se demonstrar a versatilidade da metodologia proposta e sua relevância tanto para o campo da engenharia mecânica quanto para aplicações biomédicas e robóticas.

4.2.1 Síntese de Mecanismos para Juntas de Quadril, Joelho e Tornozelo Humano

As informações utilizadas para a realização deste estudo de caso foram obtidas a partir do artigo de Plecnik e McCarthy (2016), no qual os autores apresentam um método computacional de solução exata para a síntese de geração de função de mecanismos do tipo Stephenson II de seis barras capazes de coordenar onze pares de ângulos de entrada e saída. Esse método foi aplicado ao projeto de mecanismos que reproduzem o movimento das articulações do quadril, joelho e tornozelo de um andar humano, modelado a partir de dados experimentais obtidos por meio da análise de vídeo da marcha.

Para representar o comportamento periódico de cada articulação, os autores ajustaram as curvas de movimento a séries de Fourier, de modo a obter uma formulação analítica contínua das funções de deslocamento angular. A expressão geral utilizada foi

$$f(t) = \frac{1}{2}a_0 + \sum_{m=1}^o [a_m \cos(mt) + b_m \sin(mt)], \quad (4.4)$$

em que o representa a ordem da série, e os coeficientes a_m e b_m são determinados a partir da decomposição harmônica dos dados. As funções correspondentes ao quadril, joelho e tornozelo foram determinadas com base nos coeficientes apresentados na Tabela 9, que definem o comportamento angular de cada articulação em função do tempo.

A partir das expressões apresentadas no artigo, utilizou-se o programa desenvolvido nesta monografia para identificar os mecanismos que melhor reproduzissem as curvas de movimento das articulações e, em seguida, comparar os resultados obtidos com aqueles apresentados pelo autor.

Para este caso, foram selecionados 30 pontos de precisão, buscando maximizar a similaridade entre as curvas geradas e as de referência. As curvas correspondentes, com pontos distribuídos segundo o critério de Chebyshev (Eq. 4.3) e expressas em graus, são apresentadas na Figura 41.

Tabela 9 – Coeficientes de Fourier utilizados por Plecnik e McCarthy (2016) para a síntese das funções articulares

	Quadril	Joelho	Tornozelo
Ordem (o)	3	5	5
a_0	-0.238996	-0.423173	-0.399166
a_1	0.265074	0.165304	0.248170
b_1	0.046321	0.382295	0.349262
a_2	0.002658	0.209291	0.106419
b_2	-0.056328	0.126733	0.103722
a_3	-0.020584	0.049303	0.027326
b_3	0.014721	0.040758	0.096972
a_4	0	0.001075	0.017224
b_4	0	0.017789	0.004267
a_5	0	0.004296	-0.012690
b_5	0	0.005227	-0.016123

Tabela 10 – Parâmetros geométricos otimizados e tempo para as curvas das articulações

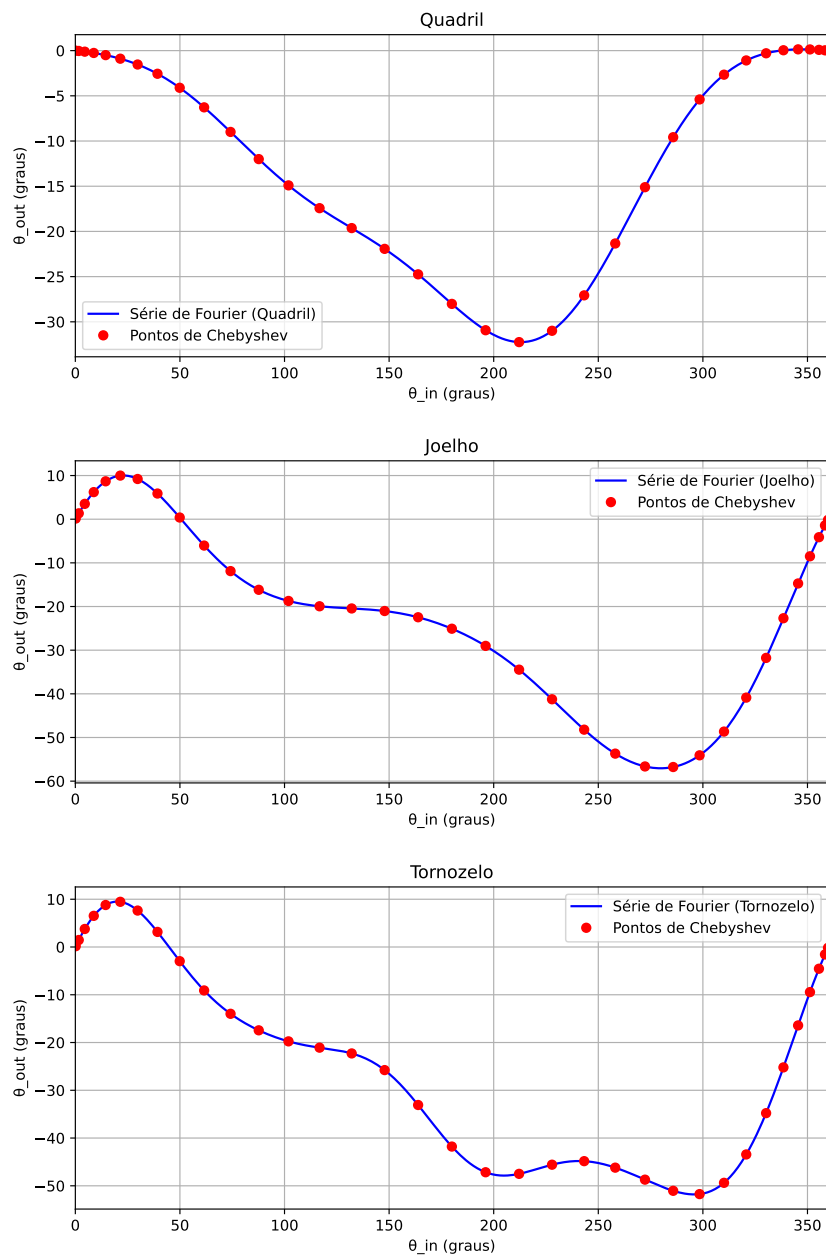
Mecanismo	L₁	L₂	L₃	L₄	L₅	L₆	L₈	L₉	ϕ	α	λ
Watt II	87.17	25.59	69.06	53.39	47.48	19.22	32.30	70.78	204.97	41.97	232.96
	Tempo de otimização: 13.21 s										
Stephenson III	46.18	49.07	29.31	74.89	69.56	37.16	94.31	97.69	123.27	295.69	286.19
	Tempo de otimização: 19.08 s										
Stephenson II	95.16	19.33	10.77	55.00	53.74	24.59	43.44	30.68	84.29	87.94	78.24
	Tempo de otimização: 86.30 s										

As configurações adotadas mantiveram-se idênticas às utilizadas nos testes, com exceção dos limites dos ângulos de qualidade de transmissão, os quais foram modificados conforme a necessidade de cada caso.

Os resultados podem ser observados nas Figuras 42, 43 e 44, que apresentam as geometrias dos mecanismos que obtiveram o melhor desempenho para cada curva articular. O mecanismo do tipo Watt II foi selecionado para o quadril, o Stephenson III para o joelho e o Stephenson II para o tornozelo. Também são apresentados os gráficos que comparam as curvas geradas pelos mecanismos com as curvas reais de cada articulação. Os parâmetros geométricos otimizados e os respectivos tempos de processamento estão reunidos na Tabela 10.

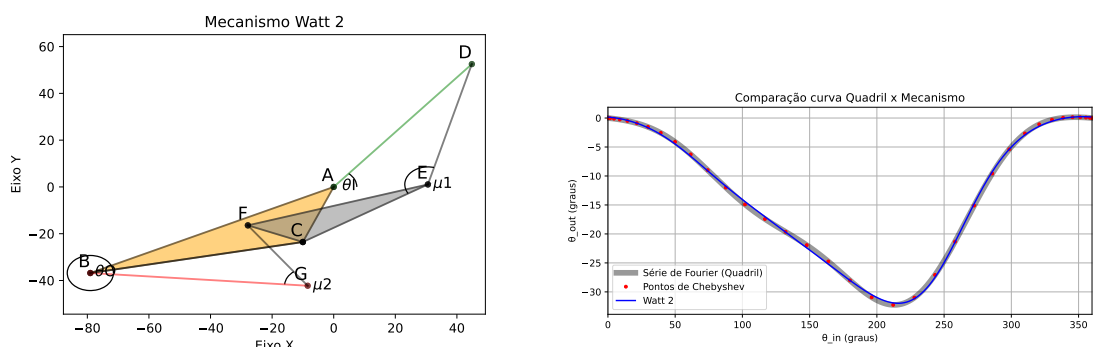
Para verificar o correto funcionamento dos mecanismos desenvolvidos, elaborou-se um modelo simplificado no software CAD OnShape, permitindo avaliar a movimentação relativa dos conjuntos mecânicos. As dimensões da perna foram obtidas a partir do artigo de Plecnik e McCarthy (2016). A visualização completa do projeto, contendo os mecanismos correspondentes ao quadril, joelho e tornozelo, é apresentada na Figura 45. Além disso, foi registrada uma simulação do movimento sincronizado dos mecanismos, disponibilizada em vídeo no YouTube no seguinte link: <<https://www.youtube.com/watch?v=D1nKKlMZ-qU>>.

Figura 41 – Curvas das articulações



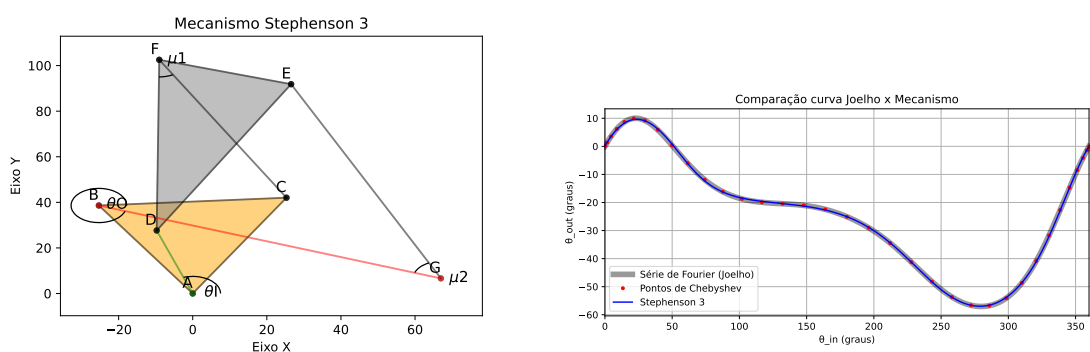
Fonte: Adaptado de Plecnik e McCarthy (2016)

Figura 42 – Mecanismo Watt II para articulação do quadril



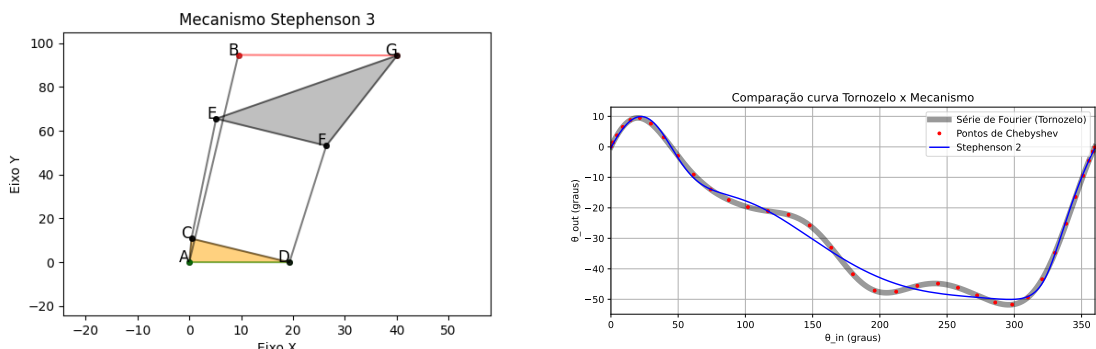
Fonte: Elaborado pelo autor

Figura 43 – Mecanismo Stephenson III para articulação do joelho



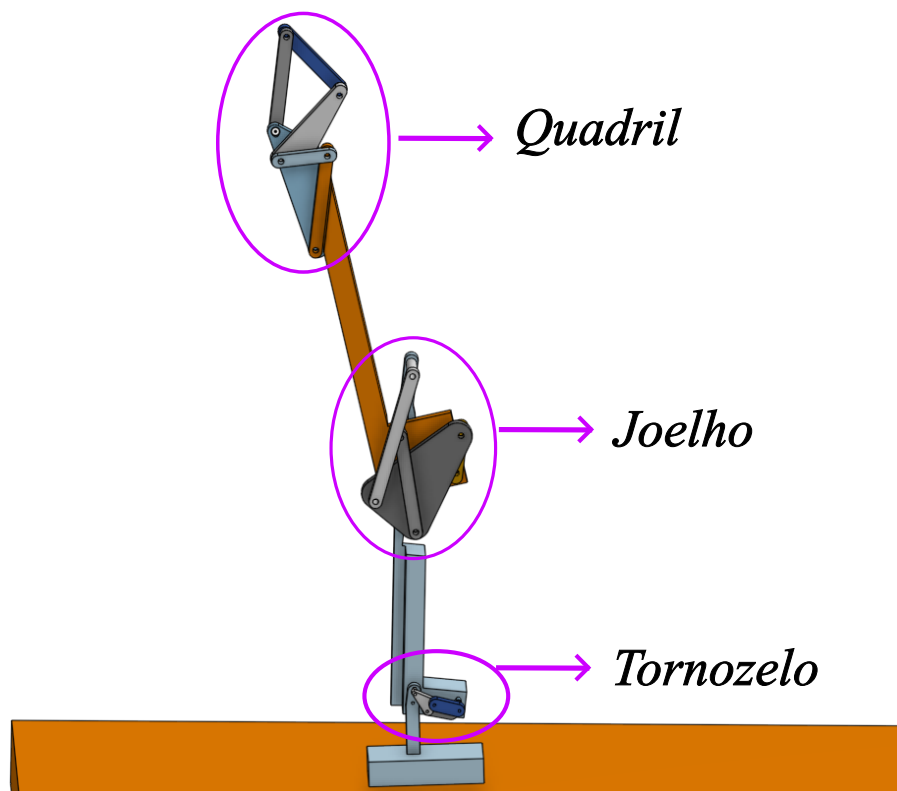
Fonte: Elaborado pelo autor

Figura 44 – Mecanismo Stephenson II para articulação do tornozelo



Fonte: Elaborado pelo autor

Figura 45 – Projeto simplificado de uma perna humanóide com mecanismos



Fonte: Elaborado pelo autor

Com todos os dados necessários apresentados, torna-se possível discutir e comparar os resultados obtidos. Para as curvas dos mecanismos correspondentes às articulações do quadril e do joelho, obteve-se um desempenho bastante satisfatório, com os mecanismos apresentando curvas muito semelhantes às obtidas pela série de Fourier das articulações. Por outro lado, a curva da articulação do tornozelo, que representa o maior desafio devido à sua complexidade, não apresentou um resultado tão bom quanto as demais, embora o desempenho ainda possa ser considerado adequado.

Ao comparar com os resultados do artigo de referência Plecnik e McCarthy (2016), que originou a metodologia aplicada neste trabalho, nota-se que os autores não apresentaram dados numéricos de erro, o que torna a comparação predominantemente visual, baseada nos gráficos das curvas e no desempenho computacional. As curvas obtidas para os mecanismos do quadril e do joelho mostraram resultados semelhantes aos do artigo original, enquanto a curva correspondente ao tornozelo apresentou menor aderência à trajetória de referência. Esse comportamento pode ser atribuído ao espaço de soluções mais restrito decorrente da simplificação da modelagem cinemática adotada neste trabalho.

No que diz respeito ao desempenho computacional, observa-se que, no artigo original, a obtenção das soluções das equações de síntese exigiu o uso do software BERTINI, o qual, a partir de uma homotopia de parâmetros construída sobre um conjunto geral de raízes, resolveu um sistema composto por 10 polinômios de oitavo grau em aproximadamente duas

horas, utilizando um processador de 64 *cores* a 2,2 GHz. Já os mecanismos propostos neste trabalho demandaram no máximo 86,30 segundos de processamento, mesmo utilizando um processador mais modesto, de 12 *cores* a 2,5 GHz.

Além disso, os resultados obtidos pelo método do artigo original ainda necessitam de filtragem posterior para identificar as soluções viáveis, o que aumenta o tempo total de processamento.

Portanto, embora o desempenho funcional da curva do tornozelo não tenha sido tão satisfatório quanto o apresentado no artigo original, o método desenvolvido nesta monografia demonstra uma vantagem significativa em termos de simplicidade e tempo de execução. Ele permite que o usuário sintetize mecanismos de forma muito mais rápida e acessível. Caso seja necessário alcançar resultados mais precisos, é possível ajustar as configurações do otimizador, aumentando o número de iterações e reduzindo a tolerância, de modo a aprimorar a convergência, ainda que com o custo de um tempo de processamento maior.

4.2.2 Síntese de Mecanismos para uma Prótese de Mão

O segundo estudo de caso tem origem na aplicação do programa de síntese e otimização ainda em suas fases iniciais de desenvolvimento, quando era possível realizar a otimização apenas para o mecanismo do tipo Watt I. Esse estudo se baseia no artigo de Krepsky et al. (2025), de autoria de um dos mesmos pesquisadores desta monografia, publicado na biblioteca digital IEEE Xplore e apresentado na CROS 2025. Ele apresenta o desenvolvimento de uma prótese de mão biomecânica de baixo custo, inspirada na *LARM Hand* (CECCARELLI; ZOTTOLA, 2017) e controlada por sinais EMG. A aplicação do programa desenvolvido permitiu realizar a síntese dimensional dos mecanismos que compõem os dedos da prótese de forma eficiente.

Este estudo de caso será descrito de maneira mais concisa, priorizando a apresentação dos resultados, dado que as capacidades do programa já foram discutidas anteriormente.

Síntese de Geração de Função do Mecanismo

A concepção da prótese partiu da definição de uma estrutura composta por cinco dedos, cada um baseado em um mecanismo do tipo Watt I com duas cadeias quádruplas cruzadas. Essa configuração foi escolhida por permitir um movimento coordenado das três falanges, de forma análoga às articulações interfalângicas humanas. Cada elo ternário do mecanismo corresponde a uma falange, enquanto as juntas revolutas representam as articulações do dedo.

O foco principal deste estudo foi a etapa de síntese de geração de função. Para isso, o programa desenvolvido foi configurado para realizar a otimização automática dos parâmetros geométricos do mecanismo Watt I, aplicando o algoritmo DE.

Para a síntese de geração de função, foram definidos seis pares de ângulos de entrada e saída baseados na cinemática de referência da *LARM Hand* (CECCARELLI; ZOTTOLA, 2017), conforme Tabela 11. Esses valores descrevem a trajetória desejada de flexão e extensão do dedo indicador. O algoritmo de otimização buscou minimizar o erro quadrático médio entre o ângulo de saída obtido (θ_O) e o ângulo desejado (θ_{Od}), penalizando soluções com ângulos de transmissão (μ_1 e μ_2) fora do intervalo de 40° a 140° , conforme os critérios de qualidade de transmissão.

Tabela 11 – Pares de ângulos de entrada e saída utilizados na síntese de função.

Índice	$\theta_I(^{\circ})$	$\theta_O(^{\circ})$
1	40	20
2	63	56
3	69	59.57
4	79	70.17
5	90	80
6	100	93.80

Os limites geométricos adotados para cada elo e ângulo estão apresentados na Tabela 12. Esses valores foram definidos a partir das dimensões antropométricas médias da mão humana e de restrições cinemáticas do modelo de referência. O otimizador foi executado até que a convergência fosse alcançada, resultando nos parâmetros apresentados na Tabela 13.

Tabela 12 – Limites geométricos adotados na otimização.

Variável	Limite Inferior	Limite Superior
L_1	10	20
L_2	35	45
L_3	1	50
L_4	40	50
L_5	20	30
L_6	1	50
L_8	5	15
L_9	1	30
ϕ	170°	220°
α	3°	100°
λ	3°	100°

O mecanismo obtido ao final da otimização foi denominado mecanismo primitivo, por representar a configuração inicial resultante do processo computacional, responsável por definir as proporções básicas dos elos e o padrão de movimento inicial. Entretanto, foram

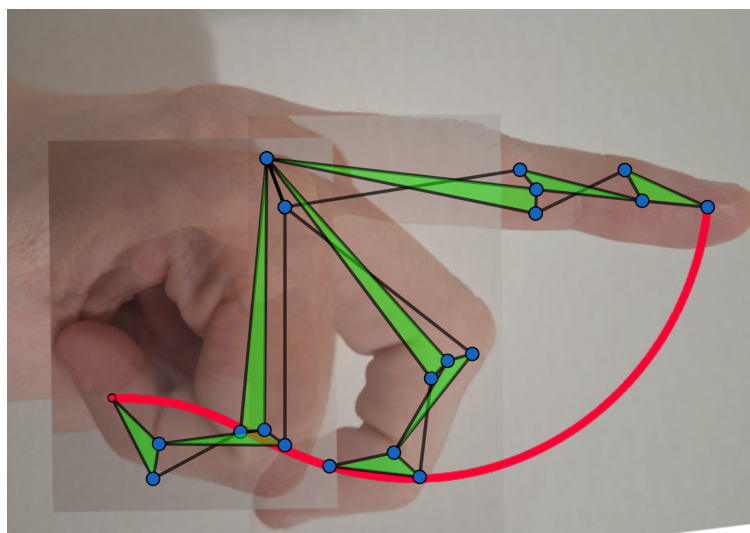
Tabela 13 – Parâmetros geométricos otimizados para o dedo indicador.

L_1	L_2	L_3	L_4	L_5	L_6	L_8	L_9	ϕ	α	λ
8.54	39.10	4.20	44.60	20.70	45.10	5.80	16.30	200.6°	36°	5°

necessários ajustes manuais no *software* GeoGebra para que a cinemática do sistema reproduzisse de forma mais fiel possível o movimento de fechamento de um dedo humanoíde.

A trajetória final obtida após a adequação do mecanismo primitivo é apresentada na Figura 46. Observa-se que o movimento da ponta do dedo segue de forma contínua e suave a curva desejada de flexão. O mesmo conjunto de dimensões foi utilizado para os dedos médio e anelar, enquanto o polegar e o mínimo foram reproduzidos em escala reduzida de 17,5% para adequação à proporção anatômica.

Figura 46 – Trajetória simulada da ponta do dedo obtida pela síntese de função.



Fonte: Adaptado de Krepsky et al. (2025)

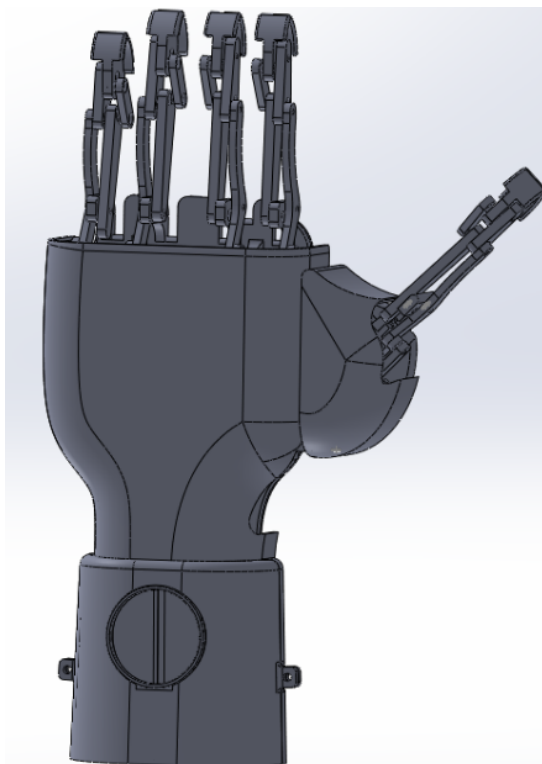
Discussão e Resultados Resumidos

A partir dos parâmetros obtidos pela síntese, foi modelado o protótipo completo da mão em *SolidWorks*, como ilustra a Figura 47, integrando os mecanismos dos cinco dedos e os elos de transmissão dos motores. O controle do sistema foi implementado com sinais EMG processados por um microcontrolador ESP32 e acionamento dos servomotores via Arduino Mini.

Os testes experimentais, apresentados em Krepsky et al. (2025), indicaram taxa de sucesso de 90% na apreensão de objetos em posição vertical e 80% em diferentes orientações, com tempo médio de resposta de 313 ms. Esses resultados demonstram que o modelo sintetizado reproduz de forma eficiente o comportamento funcional de uma mão humana, confirmando a aplicabilidade do método de síntese de geração de função

para o projeto de dispositivos biomecânicos e de reabilitação. A Figura 48 mostra o protótipo físico da prótese segurando um dos objetos dos testes. Na URL do YouTube: <<https://www.youtube.com/shorts/xJApSnnJVpI>> é possível visualizar um vídeo que mostra o movimento da prótese com acionamento via sinais eletromiográficos

Figura 47 – Protótipo final da prótese de mão modelado em *SolidWorks*.



Fonte: Krepsky et al. (2025)

Figura 48 – Protótipo final da prótese de mão segurando uma lixa



Fonte: Krepsky et al. (2025)

5 Conclusões

O presente trabalho apresentou o desenvolvimento de um programa computacional para a síntese de geração de função de mecanismos de seis barras, contemplando todas as topologias clássicas, Watt I, Watt II, Stephenson I, Stephenson II e Stephenson III, utilizando o método de otimização por Evolução Diferencial (DE). A aplicação dessa técnica permitiu formular um processo de síntese mais geral, automatizado e adaptável, capaz de atender a diferentes números de pares de precisão e restrições geométricas.

O sistema foi desenvolvido em Python, com programação orientada a objetos e interface gráfica (GUI), o que assegurou escalabilidade, portabilidade e facilidade de uso. Além disso, a estrutura modular facilita a incorporação de novos mecanismos e algoritmos em trabalhos futuros.

A metodologia implementada demonstrou eficácia na obtenção de mecanismos que reproduzem funções matemáticas de referência, evidenciando a robustez e versatilidade da abordagem baseada em otimização estocástica. Os estudos de caso, realizados para funções logarítmicas, funções compostas por séries de Fourier e trajetórias articulares humanas, validaram a aplicabilidade do programa tanto em contextos acadêmicos quanto em projetos de engenharia voltados à biomecânica e à robótica.

Os resultados obtidos mostraram que o algoritmo de Evolução Diferencial foi capaz de minimizar o erro entre ângulos de entrada e saída desejados e obtidos, produzindo soluções consistentes para diferentes configurações. Em especial, observou-se que os mecanismos Watt II, Stephenson II e Stephenson III apresentaram melhor desempenho em termos de precisão e estabilidade cinemática.

Como limitação, destaca-se que o estudo abrangeu apenas a síntese dimensional e a análise de posição, não incluindo avaliações de forças, velocidades ou acelerações. Como trabalhos futuros, propõe-se a expansão do escopo do programa com a inclusão de análises adicionais, tais como critérios dinâmicos, esforços internos e sensibilidade geométrica. Adicionalmente, a integração com ambientes CAD tridimensionais permitiria a geração automática de modelos paramétricos, facilitando a prototipagem rápida, a análise estrutural e a validação experimental dos mecanismos obtidos.

Em síntese, o trabalho contribui para a área de Engenharia de Controle e Automação ao oferecer uma ferramenta computacional que unifica a teoria de mecanismos à prática de otimização, favorecendo o projeto automatizado de mecanismos complexos e incentivando o uso de técnicas modernas de síntese no ensino e na pesquisa aplicada.

Referências Bibliográficas

- BASKAN, O. *Optimization Algorithms*. Rijeka: IntechOpen, 2016. ISBN 978-953-51-2593-8. Disponível em: <<https://doi.org/10.5772/61426>>.
- BULATOVIĆ, R. R.; ĐORĐEVIĆ, S. R.; ĐORĐEVIĆ, V. S. Cuckoo search algorithm: A metaheuristic approach to solving the problem of optimum synthesis of a six-bar double dwell linkage. *Mechanism and Machine Theory*, v. 61, p. 1–13, 2013. ISSN 0094-114X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0094114X12002108>>.
- CARBONE, G.; ROSSI, C.; SAVINO, S. Performance comparison between federica hand and larm hand. *International Journal of Advanced Robotic Systems*, SAGE Publications Sage UK: London, England, v. 12, n. 7, p. 90, 2015.
- CECCARELLI, M.; ZOTTOLA, M. Design and simulation of an underactuated finger mechanism for larm hand. *Robotica*, v. 35, n. 3, p. 483–497, 2017.
- CHU, J.; SUN, J. Numerical atlas method for path generation of spherical four-bar mechanism. *Mechanism and Machine Theory*, v. 45, n. 6, p. 867–879, 2010. ISSN 0094-114X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0094114X09002286>>.
- CHUNG, Y.-H.; LEE, J.-W. Design of a new 2 dof parallel mechanism. In: IEEE. *2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics. Proceedings (Cat. No. 01TH8556)*. [S.l.], 2001. v. 1, p. 129–134.
- DENAVID, J.; HARTENBERG, R. S. A kinematic notation for lower-pair mechanisms based on matrices. *Trans. ASME E, Journal of Applied Mechanics*, v. 22, p. 215–221, June 1955.
- ERDMAN, A.; SANDOR, G. *Advanced Mechanism Design: Analysis and Synthesis*. Prentice-Hall, 1984. Disponível em: <<https://books.google.com.br/books?id=GLopzQEACAAJ>>.
- FOUSKAKIS, D.; DRAPER, D. Stochastic optimization: a review. *International Statistical Review*, Wiley Online Library, v. 70, n. 3, p. 315–349, 2002.
- FREUDENSTEIN, F. An analytical approach to the design of four-link mechanisms. *Transactions of the American Society of Mechanical Engineers*, American Society of Mechanical Engineers, v. 76, n. 3, p. 483–489, 1954.
- FRISOLI, A.; SALSEDO, F.; BERGAMASCO, M. Design of a new tendon driven haptic interface with six degrees of freedom. In: IEEE. *8th IEEE International Workshop on Robot and Human Interaction. RO-MAN'99 (Cat. No. 99TH8483)*. [S.l.], 1999. p. 303–308.
- GLOVER, F.; TAILLARD, E.; TAILLARD, E. A user's guide to tabu search. *Annals of operations research*, Springer, v. 41, n. 1, p. 1–28, 1993.

- GOGU, G. Chebychev–grübler–kutzbach’s criterion for mobility calculation of multi-loop mechanisms revisited via theory of linear transformations. *European Journal of Mechanics - A/Solids*, v. 24, n. 3, p. 427–441, 2005. ISSN 0997-7538. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0997753805000033>>.
- HARTENBERG, R.; DENAVIT, J. *Kinematic Synthesis of Linkages*. McGraw-Hill, 1964. (McGraw-Hill series in mechanical engineering). ISBN 9780070269101. Disponível em: <<https://books.google.com.br/books?id=kWRaDQEACAAJ>>.
- HARTKE, B. Global optimization. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, Wiley Online Library, v. 1, n. 6, p. 879–887, 2011.
- HOLLAND, J. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press, 1992. (Complex Adaptive Systems). ISBN 9780262581110. Disponível em: <<https://books.google.com.br/books?id=5EgGaBkwvWcC>>.
- HUNT, K. *Kinematic Geometry of Mechanisms*. Clarendon Press, 1990. (Oxford engineering science series). ISBN 9780198562337. Disponível em: <<https://books.google.com.br/books?id=0rlqQgAACAAJ>>.
- HWANG, W.; CHEN, Y. Defect-free synthesis of stephenson-ii function generators. *Journal of Mechanisms and Robotics*, American Society of Mechanical Engineers(ASME), v. 2, n. 4, out. 2010. ISSN 1942-4302.
- KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by simulated annealing. *Science*, v. 220, n. 4598, p. 671–680, 1983. Disponível em: <<https://www.science.org/doi/abs/10.1126/science.220.4598.671>>.
- KOCHENDERFER, M. J.; WHEELER, T. A. *Algorithms for Optimization*. [S.l.]: The MIT Press, 2019. ISBN 0262039427.
- KREPSKY, D. V. et al. Development of a low-cost biomechanical hand prosthesis prototype. In: *2025 Brazilian Conference on Robotics (CROS)*. [S.l.: s.n.], 2025. v. 1, p. 1–6.
- LIU, X.-J.; WANG, J. Type synthesis of parallel mechanisms. In: _____. *Parallel Kinematics: Type, Kinematics, and Optimal Design*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. p. 31–77. ISBN 978-3-642-36929-2. Disponível em: <https://doi.org/10.1007/978-3-642-36929-2_2>.
- MARTINS, D.; MURAI, E. Mecanismos: síntese e análise com aplicações em robótica. *Edufsc, Florianópolis*, 2019.
- MCLARNAN, C. W. Synthesis of six-link plane mechanisms by numerical analysis. *Journal of Engineering for Industry*, v. 85, p. 5–10, 1963. Disponível em: <<https://api.semanticscholar.org/CorpusID:109651381>>.
- MEJIA, L.; SIMAS, H.; MARTINS, D. Force capability in general 3dof planar mechanisms. *Mechanism and Machine Theory*, v. 91, p. 120–134, 2015. ISSN 0094-114X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0094114X15000865>>.

NORTON, R. L. *Design of Machinery: An Introduction to the Synthesis and Analysis of Mechanisms and Machines*. 2nd. ed. [S.l.]: Mcgraw-Hill Series in Mechanical Engineering, 1998. ISBN 0070483957; 9780070483958.

PARRISH, B.; MCCARTHY, J. M. Identification of a usable six-bar linkage for dimensional synthesis. In: VIADERO, F.; CECCARELLI, M. (Ed.). *New Trends in Mechanism and Machine Science*. Dordrecht: Springer Netherlands, 2013. p. 255–262. ISBN 978-94-007-4902-3.

PLECNIK, M. *The Kinematic Design of Six-bar Linkages Using Polynomial Homotopy Continuation*. University of California, Irvine, 2015. ISBN 9781321854688. Disponível em: <<https://books.google.com.br/books?id=t6cHjwEACAAJ>>.

PLECNIK, M. M.; MCCARTHY, J. M. Numerical synthesis of six-bar linkages for mechanical computation. *Journal of Mechanisms and Robotics*, v. 6, p. 031012, 2014. Disponível em: <<https://api.semanticscholar.org/CorpusID:17138974>>.

PLECNIK, M. M.; MCCARTHY, J. M. Computational design of stephenson ii six-bar function generators for 11 accuracy points. *Journal of Mechanisms and Robotics*, American Society of Mechanical Engineers, v. 8, n. 1, p. 011017, 2016.

PRASHANTH, N.; MANOJ, R. M.; NIKHIL, B. Influence of link lengths amp; input angles on the foot locus trajectory of klann mechanism. *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, v. 624, n. 1, p. 012014, oct 2019. Disponível em: <<https://doi.org/10.1088/1757-899X/624/1/012014>>.

PRICE, K.; STORN, R.; LAMPINEN, J. *Differential Evolution: A Practical Approach to Global Optimization*. Springer Berlin Heidelberg, 2006. (Natural Computing Series). ISBN 9783540313069. Disponível em: <<https://books.google.com.br/books?id=hakXI-dEhTkC>>.

RAO, J.; DUKKIPATI, R. *Mechanism and Machine Theory*. New Age International (P) Limited, 2007. ISBN 9788122404265. Disponível em: <<https://books.google.com.br/books?id=AWMJtpIuZBgC>>.

RUSSELL, K.; SODHI, R. S. On the design of slider-crank mechanisms. part i: multi-phase motion generation. *Mechanism and Machine Theory*, v. 40, n. 3, p. 285–299, 2005. ISSN 0094-114X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0094114X04001302>>.

SHIAKOLAS, P.; KOLADIYA, D.; KEBRLE, J. On the optimum synthesis of six-bar linkages using differential evolution and the geometric centroid of precision positions technique. *Mechanism and Machine Theory*, v. 40, n. 3, p. 319–335, 2005. ISSN 0094-114X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0094114X04001260>>.

STEWART, G. *Afternotes on Numerical Analysis*. Society for Industrial and Applied Mathematics, 1996. (Other Titles in Applied Mathematics). ISBN 9780898713626. Disponível em: <https://books.google.com.br/books?id=LzCTp-r_QxEC>.

STORN, R.; PRICE, K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *J. of Global Optimization*, Kluwer Academic

Publishers, USA, v. 11, n. 4, p. 341–359, dez. 1997. ISSN 0925-5001. Disponível em: <<https://doi.org/10.1023/A:1008202821328>>.

SVOBODA, A.; JAMES, H. *Computing Mechanisms and Linkages*. McGraw-Hill, 1948. (Massachusetts Institute of Technology. Radiation Laboratory. Radiation Laboratory series). Disponível em: <<https://books.google.com.br/books?id=subxzQEACAAJ>>.

TSAI, L. *Mechanism Design: Enumeration of Kinematic Structures According to Function*. CRC Press, 2000. (Mechanical and Aerospace Engineering Series). ISBN 9781420058420. Disponível em: <<https://books.google.com.br/books?id=X0AHKxwWTsYC>>.

TSAI, L.-W. *Robot Analysis and Design: The Mechanics of Serial and Parallel Manipulators*. 1st. ed. USA: John Wiley & Sons, Inc., 1999. ISBN 0471325937.

UICKER JR., G. R. P. J. E. S. J. J. *Theory of Machines and Mechanisms*. 5th. ed. [S.l.]: Oxford University Press, 2017. ISBN 9780190264482; 0190264489.

WEIHMANN, L.; MARTINS, D.; dos Santos Coelho, L. Modified differential evolution approach for optimization of planar parallel manipulators force capabilities. *Expert Systems with Applications*, v. 39, n. 6, p. 6150–6156, 2012. ISSN 0957-4174. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0957417411016459>>.

YAN, H. *Creative Design of Mechanical Devices*. Springer Singapore, 1998. ISBN 9789813083578. Disponível em: <<https://books.google.com.br/books?id=i2er4dY1TJUC>>.

ZHAO, Y. et al. Tga-based solutions map method for four-position synthesis of planar 4r linkage. *Mechanical Sciences*, Copernicus Publications Göttingen, Germany, v. 13, n. 2, p. 771–790, 2022.

Apêndices

APÊNDICE A – Modelo Watt I

$$\mathbf{A} = (0, 0)$$

$$\mathbf{B} = (L_1 \cos(\phi), L_1 \sin(\phi))$$

$$\mathbf{C} = (L_2 \cos(\theta_I), L_2 \sin(\theta_I))$$

$$e_1 = \sqrt{(\mathbf{B}_x - \mathbf{C}_x)^2 + (\mathbf{B}_y - \mathbf{C}_y)^2}$$

$$\omega = \arctan(\mathbf{B}_y - \mathbf{C}_y, \mathbf{B}_x - \mathbf{C}_x)$$

$$\delta = \arccos\left(\frac{L_3^2 + e_1^2 - L_4^2}{2e_1 L_3}\right)$$

$$\mathbf{D} = (\mathbf{C}_x + L_3 \cos(\delta + \omega), \mathbf{C}_y + L_3 \sin(\delta + \omega))$$

$$\mathbf{E} = (\mathbf{C}_x + L_5 \cos(\delta + \omega + \alpha), \mathbf{C}_y + L_5 \sin(\delta + \omega + \alpha))$$

$$\beta_1 = \arctan(\mathbf{D}_y - \mathbf{B}_y, \mathbf{D}_x - \mathbf{B}_x)$$

$$\theta_O = \beta_1 - \lambda$$

$$\mathbf{G} = (\mathbf{B}_x + L_6 \cos(\theta_O), \mathbf{B}_y + L_6 \sin(\theta_O))$$

$$e_2 = \sqrt{(\mathbf{E}_x - \mathbf{G}_x)^2 + (\mathbf{E}_y - \mathbf{G}_y)^2}$$

$$\beta_2 = \arctan(\mathbf{E}_y - \mathbf{G}_y, \mathbf{E}_x - \mathbf{G}_x)$$

$$\beta_3 = \arccos\left(\frac{L_9^2 + e_2^2 - L_8^2}{2L_9 e_2}\right)$$

$$\mathbf{F} = (\mathbf{G}_x + L_9 \cos(\beta_2 - \beta_3), \mathbf{G}_y + L_9 \sin(\beta_2 - \beta_3))$$

$$\mu_1 = \arccos \left(\frac{L_4^2 + L_3^2 - e_1^2}{2L_4L_3} \right)$$

$$\mu_2 = \arccos \left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9} \right)$$

APÊNDICE B – Modelo Watt II

$$\mathbf{A} = (0, 0)$$

$$\mathbf{B} = (L_1 \cos \phi, L_1 \sin \phi)$$

$$\mathbf{C} = (L_2 \cos(\phi + \alpha_1), L_2 \sin(\phi + \alpha_1))$$

$$\mathbf{D} = (L_3 \cos \theta_{\text{in}}, L_3 \sin \theta_{\text{in}})$$

$$x_1 = \sqrt{(D_x - C_x)^2 + (D_y - C_y)^2}$$

$$\beta_1 = \arctan(D_y - C_y, D_x - C_x)$$

$$\beta_2 = \arccos\left(\frac{L_4^2 + x_1^2 - L_5^2}{2L_4x_1}\right)$$

$$\mathbf{E} = (D_x + L_4 \cos(\beta_1 + \beta_2), D_y + L_4 \sin(\beta_1 + \beta_2))$$

$$\lambda_0 = \arctan(E_y - C_y, E_x - C_x)$$

$$\mathbf{F} = (C_x + L_6 \cos(\lambda_0 - \lambda_1), C_y + L_6 \sin(\lambda_0 - \lambda_1))$$

$$x_2 = \sqrt{(F_x - B_x)^2 + (F_y - B_y)^2}$$

$$\psi = \arctan(F_y - B_y, F_x - B_x)$$

$$\omega_2 = \arccos\left(\frac{x_2^2 + L_9^2 - L_8^2}{2L_9x_2}\right)$$

$$\omega_1 = \psi - \omega_2 - \phi$$

$$\theta_O = \phi + \omega_1$$

$$\mathbf{G} = (B_x + L_9 \cos \theta_O, B_y + L_9 \sin \theta_O)$$

$$\mu_1 = \arccos \left(\frac{L_4^2 + L_5^2 - x_1^2}{2L_4L_5} \right)$$

$$\mu_2 = \arccos \left(\frac{L_8^2 + L_9^2 - x_2^2}{2L_8L_9} \right)$$

APÊNDICE C – Modelo Stephenson I

$$\mathbf{A} = (0, 0)$$

$$\mathbf{B} = (L_1 \cos \phi, L_1 \sin \phi)$$

$$\mathbf{C} = (L_2 \cos(\theta_{\text{in}} + \alpha_1), L_2 \sin(\theta_{\text{in}} + \alpha_1))$$

$$\mathbf{D} = (L_3 \cos \theta_{\text{in}}, L_3 \sin \theta_{\text{in}})$$

$$e_1 = \sqrt{(D_x - B_x)^2 + (D_y - B_y)^2}$$

$$\beta = \arctan(B_y - D_y, B_x - D_x)$$

$$\omega = \arccos\left(\frac{e_1^2 + L_5^2 - L_4^2}{2e_1L_5}\right)$$

$$\mathbf{E} = (D_x + L_5 \cos(\beta + \omega), D_y + L_5 \sin(\beta + \omega))$$

$$\xi = \arctan(E_y - B_y, E_x - B_x)$$

$$\theta_O = \xi - \lambda_1$$

$$\mathbf{F} = (B_x + L_6 \cos(\xi - \lambda_1), B_y + L_6 \sin(\xi - \lambda_1))$$

$$e_2 = \sqrt{(C_x - F_x)^2 + (C_y - F_y)^2}$$

$$\gamma = \arctan(C_y - F_y, C_x - F_x)$$

$$\delta = \arccos\left(\frac{e_2^2 + L_9^2 - L_8^2}{2e_2L_9}\right)$$

$$\mathbf{G} = (F_x + L_9 \cos(\gamma - \delta), F_y + L_9 \sin(\gamma - \delta))$$

$$\mu_1 = \arccos\left(\frac{L_5^2 + L_4^2 - e_1^2}{2L_5L_4}\right)$$

$$\mu_2 = \arccos\left(\frac{L_9^2 + L_8^2 - e_2^2}{2L_9L_8}\right)$$

APÊNDICE D – Modelo Stephenson II

$$\mathbf{A} = (0, 0)$$

$$\mathbf{B} = (L_1 \cos \phi, L_1 \sin \phi)$$

$$\mathbf{C} = (L_3 \cos(\theta_{\text{in}} + \alpha_1), L_3 \sin(\theta_{\text{in}} + \alpha_1))$$

$$\mathbf{D} = (L_2 \cos \theta_{\text{in}}, L_2 \sin \theta_{\text{in}})$$

$$\mathbf{E}(\gamma) = (C_x + L_4 \cos \gamma, C_y + L_4 \sin \gamma)$$

$$e_1(\gamma) = \sqrt{(E_x(\gamma) - D_x)^2 + (E_y(\gamma) - D_y)^2}$$

$$\omega(\gamma) = \arctan(E_y(\gamma) - D_y, E_x(\gamma) - D_x)$$

$$\omega_2(\gamma) = \arccos\left(\frac{e_1(\gamma)^2 + L_5^2 - L_6^2}{2e_1(\gamma)L_5}\right)$$

$$\mathbf{F}(\gamma) = (D_x + L_5 \cos(\omega(\gamma) - \omega_2(\gamma)), D_y + L_5 \sin(\omega(\gamma) - \omega_2(\gamma)))$$

$$\omega_3(\gamma) = \arctan(E_y(\gamma) - F_y(\gamma), E_x(\gamma) - F_x(\gamma))$$

$$\mathbf{G}(\gamma) = (F_x(\gamma) + L_8 \cos(\omega_3(\gamma) - \lambda_1), F_y(\gamma) + L_8 \sin(\omega_3(\gamma) - \lambda_1))$$

$$\gamma = \text{solve} \left[\boxed{L_9 - \sqrt{(G_x(\gamma) - B_x)^2 + (G_y(\gamma) - B_y)^2} = 0} \right]$$

$$\mathbf{E} = (C_x + L_4 \cos \gamma, C_y + L_4 \sin \gamma)$$

$$e_1 = \sqrt{(E_x - D_x)^2 + (E_y - D_y)^2}$$

$$\omega = \arctan(E_y - D_y, E_x - D_x)$$

$$\omega_2 = \arccos\left(\frac{e_1^2 + L_5^2 - L_6^2}{2e_1L_5}\right)$$

$$\mathbf{F} = (D_x + L_5 \cos(\omega - \omega_2), D_y + L_5 \sin(\omega - \omega_2))$$

$$\omega_3 = \arctan(E_y - F_y, E_x - F_x)$$

$$\mathbf{G} = (F_x + L_8 \cos(\omega_3 - \lambda_1), F_y + L_8 \sin(\omega_3 - \lambda_1))$$

$$\theta_O = \arctan(G_y - B_y, G_x - B_x)$$

$$e_2 = \sqrt{(F_x - B_x)^2 + (F_y - B_y)^2}$$

$$e_3 = \sqrt{(F_x - C_x)^2 + (F_y - C_y)^2}$$

$$\mu_1 = \arccos\left(\frac{L_4^2 + L_6^2 - e_3^2}{2L_4L_6}\right)$$

$$\mu_2 = \arccos\left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9}\right)$$

APÊNDICE E – Modelo Stephenson III

$$\mathbf{A} = (0, 0)$$

$$\mathbf{B} = (L_1 \cos \phi, L_1 \sin \phi)$$

$$\mathbf{C} = (L_2 \cos(\phi + \alpha_1), L_2 \sin(\phi + \alpha_1))$$

$$\mathbf{D} = (L_3 \cos \theta_{\text{in}}, L_3 \sin \theta_{\text{in}})$$

$$e_1 = \sqrt{(D_x - C_x)^2 + (D_y - C_y)^2}$$

$$\omega = \arccos\left(\frac{e_1^2 + L_5^2 - L_4^2}{2e_1 L_5}\right)$$

$$\beta = \arctan(D_y - C_y, D_x - C_x)$$

$$\mathbf{F} = (C_x + L_5 \cos(\beta - \omega), C_y + L_5 \sin(\beta - \omega))$$

$$\lambda_0 = \arctan(D_y - F_y, D_x - F_x)$$

$$\mathbf{E} = (F_x + L_6 \cos(\lambda_0 - \lambda_1), F_y + L_6 \sin(\lambda_0 - \lambda_1))$$

$$e_2 = \sqrt{(E_x - B_x)^2 + (E_y - B_y)^2}$$

$$\gamma = \arccos\left(\frac{e_2^2 + L_9^2 - L_8^2}{2e_2 L_9}\right)$$

$$\gamma_1 = \arctan(E_y - B_y, E_x - B_x)$$

$$\theta_O = \gamma_1 - \gamma$$

$$\mathbf{G} = (B_x + L_9 \cos \theta_O, B_y + L_9 \sin \theta_O)$$

$$\mu_1 = \arccos \left(\frac{L_5^2 + L_4^2 - e_1^2}{2L_5L_4} \right)$$

$$\mu_2 = \arccos \left(\frac{L_8^2 + L_9^2 - e_2^2}{2L_8L_9} \right)$$