



Universidade Federal do Amazonas
Faculdade de Tecnologia



Programa de Pós-Graduação em Engenharia de
Produção - PPGEP

DAVE MONTEIRO BONATES

ALGORITMO GENÉTICO APLICADO AO PROBLEMA DO SEQUENCIAMENTO
DE PRODUÇÃO: O PROBLEMA DO CAIXEIRO VIAJANTE NO SETOR DE DUAS
RODAS DO POLO INDUSTRIAL DE MANAUS

MANAUS

2025

DAVE MONTEIRO BONATES

**ALGORITMO GENÉTICO APLICADO AO PROBLEMA DO SEQUENCIAMENTO
DE PRODUÇÃO: O PROBLEMA DO CAIXEIRO VIAJANTE NO SETOR DE DUAS
RODAS DO POLO INDUSTRIAL DE MANAUS**

Dissertação de Mestrado em Engenharia de Produção apresentada ao Programa de Pós-Graduação Stricto Sensu, em Engenharia de Produção da Universidade Federal do Amazonas como pré-requisito para obtenção do título de Mestre em Engenharia de Produção pela Universidade Federal do Amazonas – UFAM.

Área de Concentração: Gestão da Produção e Operações.

Linha de Pesquisa: Modelagem, Simulação e Otimização de Processos.

Orientadora: Professora Dra. Gabriela de Mattos Veroneze

MANAUS

2025

Ficha Catalográfica

Elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

B699a Bonates, Dave Monteiro

Algoritmo genético aplicado ao problema do sequenciamento de produção: o problema do caixeiro viajante no setor de duas rodas do polo industrial de Manaus / Dave Monteiro Bonates. - 2025.
83 f. : il., color. ; 31 cm.

Orientador(a): Gabriela de Mattos Veroneze.

Dissertação (mestrado) - Universidade Federal do Amazonas, Programa de Pós-Graduação em Engenharia de Produção, Manaus, 2025.

1. Algoritmos genéticos. 2. Sequenciamento de produção. 3. Problema do caixeiro viajante. 4. Polo industrial de Manaus. I. Veroneze, Gabriela de Mattos. II. Universidade Federal do Amazonas. Programa de Pós-Graduação em Engenharia de Produção. III. Título

DAVE MONTEIRO BONATES

**ALGORITMO GENÉTICO APLICADO AO PROBLEMA DO SEQUENCIAMENTO
DE PRODUÇÃO: O PROBLEMA DO CAIXEIRO VIAJANTE NO SETOR DE DUAS
RODAS DO POLO INDUSTRIAL DE MANAUS**

Dissertação de Mestrado em Engenharia de Produção apresentada ao Programa de Pós-Graduação Stricto Sensu, em Engenharia de Produção da Universidade Federal do Amazonas como pré-requisito para obtenção do título de Mestre em Engenharia de Produção pela Universidade Federal do Amazonas – UFAM.

Área de Concentração: Gestão da Produção e Operações.

Linha de Pesquisa: Modelagem, Simulação e Otimização de Processos.

Orientador: Gabriela de Mattos Veroneze, Ph.D.

BANCA EXAMINADORA

Profa. Dr. Gabriela de Mattos Veroneze
Programa de Pós-Graduação em Engenharia de Produção – UFAM
Orientadora e Presidente

Profa. Dr. Jaime Casanova Soeiro Júnior
Programa de Pós-Graduação em Engenharia de Produção – UFAM
Membro PPGEP

Profa. Dr. Anelize Seniski Silva
Programa de Pós-Graduação em Engenharia de Produção – UFAM
Membro Externo

Manaus, 28 de abril de 2025

À minha mãe, Mirlane, e ao meu pai,
Derblay, por tudo que fizeram e me
ensinaram. Cada conquista minha é fruto
do sacrifício e do esforço de vocês.
Sou eternamente grato.

AGRADECIMENTOS

À minha companheira de vida, Amanda Rangel, por estar ao meu lado em cada etapa, com amor, paciência e apoio. Sua presença foi essencial para que eu seguisse em frente.

À minha família, pelo suporte constante, pelo grande incentivo e por estar sempre ao meu lado.

Ao corpo docente do Programa de Pós-Graduação em Engenharia de Produção da UFAM, pelos ensinamentos, orientações e experiências compartilhadas, que enriqueceram minha formação acadêmica e pessoal. Em especial, à minha orientadora, professora Dra. Gabriela de Mattos Veroneze, pelas valiosas contribuições a este trabalho, pela paciência, dedicação e parceria durante todo o processo.

Por fim, estendo meus agradecimentos a todos que contribuíram para a concretização deste trabalho.

RESUMO

O aumento da complexidade nas linhas de montagem, impulsionado pela crescente diversidade de modelos, tem imposto desafios significativos ao sequenciamento de produção na indústria de motocicletas. Este trabalho propõe uma abordagem inovadora para otimização do sequenciamento produtivo por meio da aplicação de Algoritmos Genéticos, modelando o problema a partir do clássico Problema do Caixeiro Viajante. A pesquisa foi desenvolvida em uma empresa do setor de duas rodas localizada no Polo Industrial de Manaus, contexto no qual a troca entre modelos impacta diretamente o tempo total de produção. Foram desenvolvidos experimentos com três linhas de montagem reais, cujos resultados demonstraram a capacidade dos Algoritmos Genéticos em reduzir significativamente os tempos de troca, com ganhos médios de aproximadamente 80% em relação às sequências geradas aleatoriamente. A implementação utilizou a biblioteca DEAP em Python, com calibração criteriosa de parâmetros como tamanho populacional, taxa de mutação e número de gerações. Os resultados indicam que a abordagem é eficaz, robusta e escalável, apresentando ganhos expressivos na eficiência operacional. Este estudo contribui ao evidenciar o potencial dos Algoritmos Genéticos como ferramenta estratégica de apoio à decisão no contexto industrial, e reforça a viabilidade de sua aplicação prática em problemas combinatórios complexos.

Palavras-chave: Algoritmos Genéticos, Sequenciamento de Produção, Problema do Caixeiro Viajante, Polo Industrial de Manaus.

ABSTRACT

The increasing complexity of assembly lines, driven by the growing variety of motorcycle models, has posed significant challenges to production sequencing in the motorcycle industry. This study proposes an innovative approach to optimize production sequencing through the application of Genetic Algorithms, modeling the problem based on the classical Traveling Salesman Problem (TSP). The research was conducted in a two-wheeler manufacturing company located in the Manaus Industrial Hub, where model changeovers directly impact total production time. Experiments were carried out using three real-world assembly lines, and the results demonstrated the capability of Genetic Algorithms to significantly reduce changeover times, achieving average improvements of approximately 80% compared to randomly generated sequences. The implementation employed the DEAP library in Python, with careful calibration of parameters such as population size, mutation rate, and number of generations. The findings indicate that the proposed approach is effective, robust, and scalable, delivering substantial gains in operational efficiency. This study contributes by highlighting the potential of Genetic Algorithms as a strategic decision-support tool in industrial contexts and reinforces the feasibility of their practical application to complex combinatorial problems.

Keywords: Genetic Algorithms. Production Sequencing. Traveling Salesman Problem, Manaus Industrial Hub.

LISTA DE FIGURAS

Figura 1 - Participação dos subsetores de atividades no faturamento do Polo Industrial de Manaus – jan a out/2024	13
Figura 2 - Representação do Problema do Caixeiro Viajante (PCV)	17
Figura 3 - Ilustração de ótimo local e ótimo global.....	26
Figura 4 - Procedimento de otimização de um Algoritmo Genético	26
Figura 5 – Terminologia dos Algoritmos Genéticos	27
Figura 6 - Exemplo da operação de cruzamento	32
Figura 7 - Exemplo da operação de mutação	34
Figura 8 - Etapas metodológicas da pesquisa	44
Figura 9 - Ilustração do impacto entre as trocas de modelos de motocicletas.....	46
Figura 10 - Analogia ilustrativa entre o PCV e o problema do sequenciamento de produção de motocicletas.....	47
Figura 11 - Relação entre número de instâncias e tempo de processamento.....	48
Figura 12 - Exemplo de cromossomo permutacional e sua sequência representativa	53
Figura 13 - Evolução das soluções do AG para a Linha de Montagem A.....	60
Figura 14 - Evolução das soluções do AG para a Linha de Montagem B.....	61
Figura 15 - Evolução das soluções do AG para a Linha de Montagem C	62
Figura 16 - Comparação entre tempo total de trocas de modelos: AG vs. Sequência Aleatória	64
Figura 17 - Comparativo do AG para diferentes parâmetros de tamanho populacional (POP) e taxa de mutação (MUT)	66

LISTA DE TABELAS

Tabela 1 - Modelos de motocicletas e linhas de montagem	45
Tabela 2 - Comparativo entre métodos de resolução	49
Tabela 3 - Comparativo entre métodos meta-heurísticos	50
Tabela 4 - Comparativo entre as ferramentas para implementação do AG	52
Tabela 5 - Quantidade de cavaletes por linha de montagem	55
Tabela 6 - Tempo de produção dos modelos de motocicletas	55
Tabela 7 - Comparação das sequências de produção: AG vs. Sequência Aleatória	63

LISTA DE SIGLAS

ABC – Artificial Bee Colony

ACO – Ant Colony Optimization

AG – Algoritmo Genético

DEAP – Distributed Evolutionary Algorithms in Python

GRASP – Greedy Randomized Adaptive Search Procedure

PCV – Problema do Caixeiro Viajante

PIM – Polo Industrial de Manaus

PSO – Particle Swarm Optimization

SA – Simulated Annealing

TS – Tabu Search

VRP – Vehicle Routing Problem

PL – Programação Linear

GR – Greedy Algorithm

NN – Nearest Neighbor

CI – Cheapest Insertion

FI – Farthest Insertion

NI – Nearest Insertion

RI – Random Insertion

LK – Lin-Kernighan

BB – Branch and Bound

FF – Ford-Fulkerson

HK – Held-Karp

PMX – Partially Matched Crossover

OBM – Order Based Mutation

SUMÁRIO

1.	INTRODUÇÃO	11
1.1	CONTEXTUALIZAÇÃO	11
1.2	FORMULAÇÃO DO PROBLEMA	12
1.3	OBJETIVOS	12
1.3.1	OBJETIVO GERAL	12
1.3.2	OBJETIVOS ESPECÍFICOS	12
1.4	JUSTIFICATIVA	13
1.5	ESTRUTURA DA DISSERTAÇÃO	14
2.	ESTADO DA ARTE	15
2.1	PESQUISA OPERACIONAL E PROGRAMAÇÃO MATEMÁTICA.....	15
2.2	O PROBLEMA DO CAIXEIRO VIAJANTE (PCV).....	16
2.2.1.	O PROBLEMA DO SEQUENCIAMENTO DA PRODUÇÃO E SUA RELAÇÃO COM O PCV.....	18
2.3	MÉTODOS DE RESOLUÇÃO DO PCV	20
2.3.1.	MÉTODOS HEURÍSTICOS	21
2.3.2.	MÉTODOS META-HEURÍSTICOS	22
2.3.3.	MÉTODOS EXATOS.....	23
2.4	ALGORÍTIMOS GENÉTICOS	24
2.4.1.	ENQUADRAMENTO GERAL	24
2.4.2.	CODIFICAÇÃO	27
2.4.3.	GERAÇÃO DA POPULAÇÃO INICIAL	28
2.4.4.	FUNÇÃO DE APTIDÃO.....	29
2.4.5.	OPERADORES GENÉTICOS	30
2.4.5.1.	SELEÇÃO	30
2.4.5.2.	CRUZAMENTO	31
2.4.5.3.	MUTAÇÃO	33
2.4.6.	CRITÉRIO DE PARADA.....	34
2.4.7.	AMBIENTES PARA IMPLEMENTAÇÃO DO AG	35

2.4.7.1.	DISTRIBUTED EVOLUTIONARY IN PYTHON.....	36
2.4.7.2.	MATLAB.....	36
2.4.7.3.	MICROSOFT EXCEL SOLVER.....	37
2.4.7.4.	HEURISTICLAB.....	38
2.4.7.5.	EVOLVER.....	39
3.	TRABALHOS RELACIONADOS.....	40
4.	METODOLOGIA.....	44
4.1	ETAPAS DE DESENVOLVIMENTO.....	44
4.2	DESCRIÇÃO DO PROBLEMA.....	45
4.3	RELAÇÃO DO PCV COM O PROBLEMA DO SEQUENCIAMENTO DE PRODUÇÃO DE MOTOCICLETAS.....	47
4.3.1.	COMPLEXIDADE DO PROBLEMA.....	48
4.4	ESCOLHA DO MÉTODO DE RESOLUÇÃO.....	49
4.5	ESCOLHA DO AMBIENTE DE DESENVOLVIMENTO.....	51
4.6	FUNCIONAMENTO DO AG NO PROBLEMA DE SEQUENCIAMENTO DE PRODUÇÃO DE MOTOCICLETAS.....	53
4.7	CÁLCULO DO TEMPO ENTRE AS TROCAS DE MODELOS.....	54
4.8	DEFINIÇÃO DOS PARÂMETROS DO ALGORITMO GENÉTICO (CONDIÇÕES DE CONTORNOS).....	56
4.4.1	FUNÇÃO DE AVALIAÇÃO.....	56
4.4.2	TAMANHO DA POPULAÇÃO.....	57
4.4.3	SELEÇÃO.....	58
4.4.4	CRUZAMENTO.....	58
4.4.5	MUTAÇÃO.....	59
4.4.6	CRITÉRIO DE PARADA.....	59
5.	RESULTADOS E DISCUSSÃO.....	60
5.1	LINHA DE MONTAGEM A.....	60
5.2	LINHA DE MONTAGEM B.....	61
5.3	LINHA DE MONTAGEM C.....	62
5.4	COMPARATIVO ENTRE SEQUÊNCIA OTIMIZADA E ALEATÓRIA.....	63
5.5	ANÁLISE DE SENSIBILIDADE DOS PARÂMETROS DO AG.....	64
5.6	DISCUSSÃO DOS RESULTADOS OBTIDOS.....	67
6.	CONCLUSÃO.....	69
7.	CONTRIBUIÇÕES DA PESQUISA.....	70
7.1	IMPACTOS ACADÊMICOS.....	70
7.2	IMPACTOS ECONÔMICOS.....	71
7.3	IMPACTOS SOCIAIS.....	71

1. INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Nos últimos anos, tem-se observado um aumento expressivo na competitividade entre empresas de diversos setores. Esse fenômeno é impulsionado por uma combinação de fatores, como a globalização, que expandiu os mercados e intensificou a concorrência, e o avanço tecnológico, que viabilizou o desenvolvimento de novos produtos e serviços, além de aprimorar os processos produtivos (Baumann, 2022).

Nesse cenário, as empresas têm investido cada vez mais em inovação e otimização de processos para se destacar no mercado e alcançar melhores resultados. Diversas estratégias e abordagens surgem com o objetivo de maximizar recursos, eliminar desperdícios e aumentar a eficiência operacional. Ao implementar tais estratégias, as empresas conseguem não apenas melhorar a produtividade, mas também obter vantagens competitivas significativas no mercado (H. L. Corrêa, C. A. Corrêa, 2022).

A otimização dos processos produtivos é uma das principais formas de atingir esses objetivos. Esse processo envolve identificar gargalos, eliminar atividades desnecessárias, melhorar o sequenciamento de produção e utilizar os recursos disponíveis de forma eficiente. Tais ações resultam em maior produtividade, redução de tempos de produção e minimização de custos (Slack, Brandon-Jones e Burgess, 2023).

No setor industrial brasileiro, o Polo Industrial de Manaus (PIM) destaca-se como um importante centro de produção, especialmente no segmento de motocicletas. Contudo, a diversidade de modelos fabricados e a necessidade de alta eficiência impõem desafios significativos às linhas de montagem. Entre esses desafios, destaca-se a dificuldade de gerenciar o sequenciamento de produção de forma eficiente, evitando gargalos, tempos de troca elevados e desperdício de recursos.

Nesse contexto, o uso de ferramentas computacionais avançadas torna-se essencial para enfrentar problemas produtivos complexos. Entre essas ferramentas, os algoritmos genéticos destacam-se como uma abordagem robusta e eficiente para otimizações. Inspirados na evolução natural, esses algoritmos oferecem soluções práticas e inovadoras para cenários que envolvem múltiplas variáveis e restrições,

como o sequenciamento de produção. Sua aplicabilidade prática tem contribuído significativamente para que indústrias aumentem a competitividade e reduzam custos (Katoch, Chauhan e Kumar, 2020).

Dessa forma, este trabalho situa-se na interseção entre a necessidade de melhorias nos processos produtivos e o potencial das aplicações avançadas da computação evolutiva, com foco no aumento da eficiência operacional e no fortalecimento da competitividade do Polo Industrial de Manaus.

1.2 FORMULAÇÃO DO PROBLEMA

O aumento da diversidade de produtos e a pressão por eficiência na produção industrial têm intensificado os desafios enfrentados pelas linhas de montagem, particularmente no setor de duas rodas. No Polo Industrial de Manaus, onde a produção de motocicletas é significativa para a economia regional, o sequenciamento de produção torna-se um elemento crítico para a operação eficiente das linhas. O problema consiste em organizar o sequenciamento da produção de maneira a reduzir os tempos de troca entre os diferentes modelos de motocicletas.

Os métodos tradicionais de planejamento produtivo têm mostrado limitações quando aplicados a cenários de alta complexidade, nos quais as restrições e variáveis são muitas e interdependentes. Assim, surge a seguinte questão da pesquisa:

"Como aplicar algoritmos genéticos para otimizar o sequenciamento de produção em linhas de montagem de motocicletas de uma indústria do setor de duas rodas no Polo Industrial de Manaus?"

Essa questão norteadora guia o presente trabalho, que busca propor uma abordagem que contemple as particularidades do ambiente industrial estudado e contribua para a solução desta problemática.

1.3 OBJETIVOS

1.3.1 OBJETIVO GERAL

Propor uma estratégia para otimizar o sequenciamento de produção de motocicletas por meio da utilização de algoritmos genéticos.

1.3.2 OBJETIVOS ESPECÍFICOS

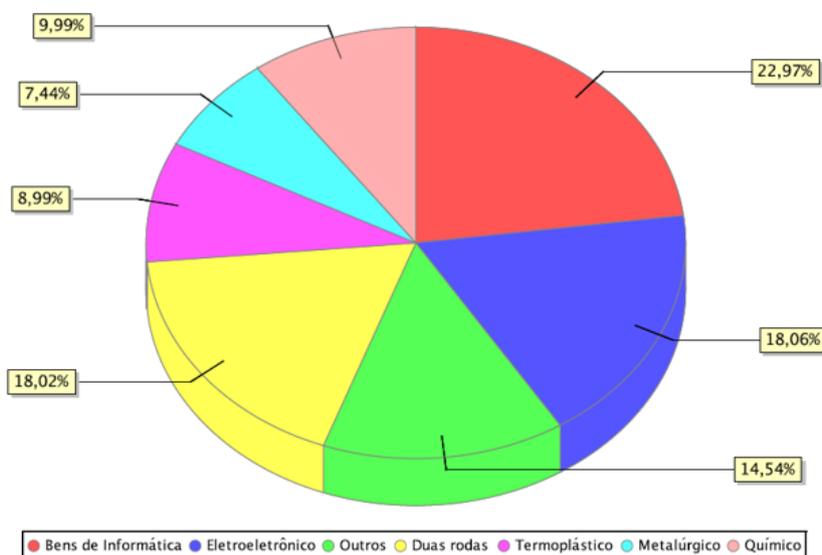
- i. Desenvolver uma modelagem computacional que simule o processo de sequenciamento de produção utilizando algoritmos genéticos, considerando as particularidades do problema estudado;

- ii. Apresentar uma proposta teórica de sequenciamento eficiente para os modelos de motocicletas nas linhas de montagem;
- iii. Demonstrar o potencial dos algoritmos genéticos como uma ferramenta alternativa para solucionar problemas em ambientes industriais, fomentando futuras pesquisas e ampliando sua aplicação em novos contextos.

1.4 JUSTIFICATIVA

A busca por soluções inovadoras para enfrentar os desafios produtivos contemporâneos tem impulsionado a adoção de ferramentas computacionais avançadas em diversos setores industriais. Inserida nesse contexto, a empresa objeto deste estudo é uma montadora de motocicletas de grande porte, localizada em Manaus, no estado do Amazonas, e integrante do setor de duas rodas do Polo Industrial de Manaus. Esse setor, de grande relevância para a economia regional, ocupa a terceira posição em participação no faturamento do PIM, conforme ilustrado na Figura 1.

Figura 1 - Participação dos subsetores de atividades no faturamento do Polo Industrial de Manaus – jan a out/2024



Fonte: Brasil (2024)

Com linhas de montagem complexas e uma grande variedade de modelos, a empresa enfrenta desafios no sequenciamento da produção, processo que exige a organização eficiente das ordens de fabricação para minimizar os tempos de troca entre modelos, reduzir desperdícios e otimizar o uso dos recursos disponíveis. No

entanto, os métodos tradicionais de planejamento produtivo apresentam limitações diante da alta complexidade e da interdependência de múltiplas variáveis, dificultando a busca por soluções eficientes.

Diante desse cenário, os algoritmos genéticos surgem como uma abordagem inovadora e eficaz. Inspirados na evolução natural, esses algoritmos possibilitam a busca de soluções ótimas ou subótimas para problemas combinatórios complexos, como o sequenciamento de produção. Sua flexibilidade e capacidade de adaptação torna-os uma ferramenta estratégica para otimizar processos industriais (Alhijawi e Awajan, 2024).

A implementação dessa abordagem não apenas fortalece a competitividade da empresa no mercado, mas também contribui para o desenvolvimento do setor de duas rodas no PIM. Dessa forma, este trabalho justifica-se pela necessidade de integrar tecnologias avançadas à tomada de decisões industriais, promovendo o uso de ferramentas inovadoras e atendendo às demandas da empresa. Além disso, a pesquisa amplia o entendimento sobre a aplicação dos algoritmos genéticos em diferentes contextos industriais, fornecendo subsídios para futuras investigações e implementações em outras áreas, industriais ou não.

1.5 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação foi organizada em oito capítulos, cada um com uma abordagem específica e conectados para atender ao objetivo geral e específicos do trabalho.

Capítulo 1 - Introdução: Este capítulo apresenta uma contextualização do tema, a formulação do problema, os objetivos geral e específicos, e a justificativa para a realização do trabalho, destacando a relevância da aplicação de algoritmos genéticos em problemas complexos de sequenciamento de produção.

Capítulo 2 - Revisão Teórica: Explora conceitos fundamentais relacionados ao Problema do Caixeiro Viajante, sua relação com o sequenciamento de produção, métodos de resolução do PCV, e uma introdução aos algoritmos genéticos, além de apresentar ferramentas e ambientes para sua implementação.

Capítulo 3 - Trabalhos Relacionados: Este capítulo apresenta pesquisas que empregaram Algoritmos Genéticos em problemas similares, destacando aplicações relevantes para o desenvolvimento desta pesquisa.

Capítulo 4 - Metodologia: Descreve detalhadamente a escolha do método de resolução e do ambiente de desenvolvimento, além de como os algoritmos genéticos foram configurados para a resolução do problema, incluindo definição de parâmetros como função de avaliação, tamanho da população, seleção, cruzamento, mutação e critério de parada.

Capítulo 5 - Resultados e Discussão: Apresenta os resultados obtidos nas três linhas de montagem (A, B e C), analisando a eficiência do AG no sequenciamento de produção e discutindo os impactos observados.

Capítulo 6 - Conclusão: Sintetiza os principais resultados do trabalho, discute as limitações encontradas e sugere direções para pesquisas futuras.

Capítulo 7 - Contribuições da Pesquisa: Detalha os impactos acadêmicos, econômicos e sociais da pesquisa, demonstrando como os resultados podem ser aplicados na indústria e na academia.

2. ESTADO DA ARTE

2.1 PESQUISA OPERACIONAL E PROGRAMAÇÃO MATEMÁTICA

A Pesquisa Operacional (PO) é uma disciplina que utiliza métodos científicos para resolver problemas complexos, proporcionando suporte analítico à tomada de decisões em cenários onde os recursos são limitados. Sua aplicação é amplamente difundida em setores como transporte, varejo e manufatura, onde desempenha um papel crucial na otimização de processos, redução de custos e melhoria da satisfação do cliente. Por meio de uma variedade de métodos analíticos, a PO busca soluções eficazes para problemas diversos, trazendo benefícios tanto para indivíduos quanto para empresas e governos. Sua abordagem sistemática permite enfrentar desafios com base em modelagem matemática e simulações (Arenales et al., 2015).

Dentro desse contexto, a programação matemática se destaca como um dos pilares da PO, abrangendo uma ampla gama de técnicas e algoritmos para resolver problemas de otimização. Segundo Hillier e Lieberman (2013), a programação matemática permite modelar e solucionar problemas por meio de representações formais baseadas em equações, inequações e funções matemáticas. A formulação correta do problema é essencial para garantir a obtenção de soluções ótimas, sendo classificada em categorias como programação linear, inteira e não linear, dependendo da natureza do problema e das restrições envolvidas.

A modelagem matemática desempenha um papel central nesse processo, ao traduzir situações do mundo real em estruturas matemáticas que representam de forma precisa as interações, restrições e objetivos do sistema analisado. Esse processo envolve o uso de equações e funções que capturam as características específicas do problema, proporcionando uma base sólida para a aplicação de algoritmos de otimização (Arenales et al., 2015).

Para resolver os modelos formulados, é necessário utilizar algoritmos de otimização apropriados, que variam de acordo com a complexidade e as especificidades de cada problema. Esses algoritmos são projetados para explorar o espaço de busca de maneira eficaz, identificando soluções que atendam aos requisitos definidos pela modelagem. A escolha do método mais adequado é um passo estratégico, sendo fundamental para garantir a eficácia do processo de otimização e a qualidade dos resultados (Didden et al., 2023).

Diante dos desafios crescentes e da crescente complexidade dos problemas enfrentados nos diversos setores, existe uma demanda contínua por algoritmos mais eficientes e técnicas de otimização avançadas. Segundo Diaby e Karwan (2016), a evolução dessas ferramentas é essencial para acompanhar a necessidade de soluções mais rápidas e precisas. A integração entre modelagem matemática, algoritmos otimizados e recursos computacionais robustos consolida a programação matemática como uma peça-chave na pesquisa operacional moderna.

Essa base teórica se conecta diretamente ao Problema do Caixeiro Viajante, um dos casos mais emblemáticos de problemas de otimização combinatória. Este problema exemplifica os desafios enfrentados pela PO, ao mesmo tempo em que destaca o potencial das ferramentas de programação matemática na busca por soluções eficientes em contextos reais e complexos.

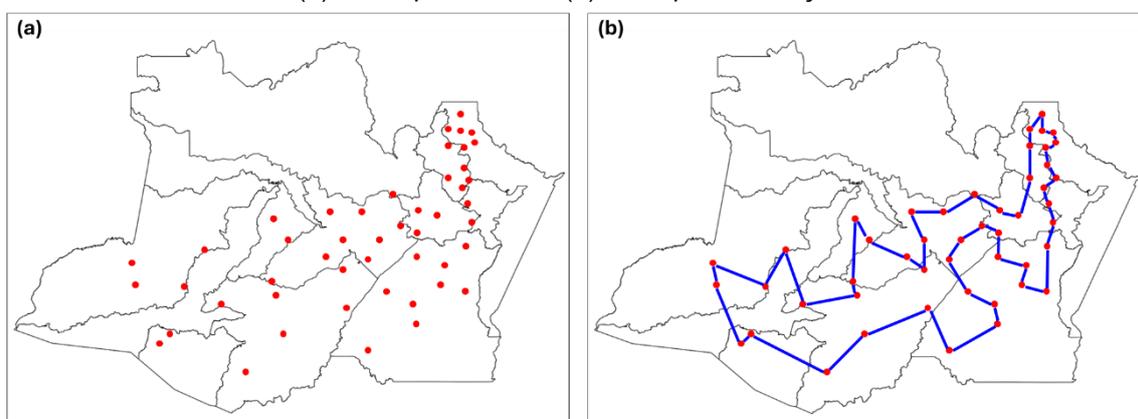
2.2 O PROBLEMA DO CAIXEIRO VIAJANTE (PCV)

O Problema do Caixeiro Viajante (PCV) é um dos tópicos mais relevantes no campo da otimização combinatória, amplamente explorado devido à sua importância teórica e prática. Segundo Silva (2019), o PCV possui aplicações significativas em áreas como logística, roteamento de veículos e otimização de processos. Essas aplicações reforçam sua relevância, especialmente em um mundo onde a eficiência em processos é crucial. No entanto, sua complexidade computacional o torna um dos problemas mais desafiadores enfrentados pela comunidade científica.

Em termos gerais, o PCV consiste em determinar o menor percurso possível para um viajante que precisa visitar um conjunto de cidades, passando por cada uma exatamente uma vez e retornando ao ponto de partida. O objetivo principal é minimizar a distância total percorrida, o que caracteriza o problema como um desafio de natureza combinatória complexa. Essa formulação, conforme descrita por Hariyadi et al. (2020), destaca a necessidade de se buscar soluções que sejam, se não ótimas, ao menos suficientemente eficazes para atender às demandas práticas.

Como enfatizado por Arenales et al. (2015), o PCV possui algumas restrições fundamentais que aumentam sua complexidade. Primeiramente, o caixeiro viajante deve visitar todas as cidades especificadas no roteiro, garantindo que cada uma seja visitada uma única vez. Além disso, o trajeto deve ser fechado, ou seja, formar um ciclo completo em que o ponto final coincide com o ponto de partida (Figura 2). Essas condições fazem do PCV um problema intrinsecamente desafiador, tanto no plano teórico, ao desenvolver modelos matemáticos para sua descrição, quanto no plano prático, ao implementar algoritmos que resolvam instâncias reais.

Figura 2 - Representação do Problema do Caixeiro Viajante (PCV)
(a) Exemplo do PCV. (b) Exemplo de solução do PCV.



Fonte: Adaptado de Shahab (2019)

A formulação matemática do PCV é fundamental para o desenvolvimento de algoritmos capazes de resolver o problema. Como demonstrado por Silva (2019), essas formulações utilizam representações matemáticas que permitem expressar o problema de maneira clara e precisa, facilitando a aplicação de métodos computacionais. A modelagem matemática desempenha, assim, um papel essencial na busca por soluções eficientes.

Ao longo das últimas décadas, a complexidade computacional do PCV tem sido um tema central de estudo na área de otimização. Sua classificação como um

problema NP-difícil, conforme descrito por Korte e Vygen (2018), significa que não há um algoritmo eficiente conhecido que seja capaz de resolvê-lo em tempo polinomial para todas as instâncias possíveis. À medida que o número de cidades aumenta, o espaço de soluções cresce exponencialmente, tornando inviável a resolução exata de grandes instâncias com os métodos tradicionais.

Essa dificuldade impulsionou o desenvolvimento de algoritmos heurísticos, que não garantem a obtenção da solução ótima, mas oferecem respostas "suficientemente boas" em um tempo computacional viável. Esses métodos, embora eficazes para instâncias menores, enfrentam limitações significativas à medida que a escala do problema aumenta (T. Mzili, I. Mzili e M. Riffi, 2023).

A busca por soluções para o PCV continua sendo um desafio em aberto, especialmente para instâncias maiores. Para enfrentar essa complexidade, a comunidade científica tem explorado técnicas híbridas, combinando heurísticas com algoritmos mais avançados e técnicas de otimização. Essa abordagem busca equilibrar a qualidade das soluções com o tempo necessário para encontrá-las, como sugerido por Abdulkarim e Alshammari (2015). Além disso, o avanço das tecnologias computacionais tem ampliado as possibilidades de enfrentar problemas desse porte.

Em síntese, o Problema do Caixeiro Viajante é um desafio emblemático da otimização combinatória, com grande impacto em diversas áreas práticas. Sua complexidade computacional e a ausência de soluções exatas em tempo polinomial para grandes instâncias ressaltam a necessidade de métodos com abordagens eficientes. Este problema não só continua a estimular avanços significativos na ciência da computação e na matemática aplicada, mas também serve como referência para abordar problemas semelhantes em contextos diversos.

2.2.1. O PROBLEMA DO SEQUENCIAMENTO DA PRODUÇÃO E SUA RELAÇÃO COM O PCV

O sequenciamento da produção é uma atividade essencial no ambiente industrial, visando determinar a ordem ideal para a execução de tarefas ou operações, de modo a otimizar critérios como tempo de processamento, utilização de recursos e cumprimento de prazos. Uma abordagem relevante para resolver problemas de sequenciamento é a modelagem baseada no PCV, que fornece uma estrutura matemática clara e lógica para formular esses problemas. Esse paralelo entre

otimização de roteamento e sequenciamento é bem documentado na literatura recente, como pode ser observado nos trabalhos a seguir.

No estudo realizado por Rao, Xanthopoulos e Zheng (2020), o PCV foi utilizado como base para modelar o problema de sequenciamento na produção de gizes de cera. A analogia entre as cidades do PCV e as cores dos gizes permitiu representar matematicamente os tempos de setup dependentes da sequência, possibilitando uma formulação clara do problema e o desenvolvimento de soluções que minimizam esses tempos para otimização do processo.

Na indústria têxtil, uma abordagem semelhante foi explorada no trabalho de Gomes, Pinheiro e Saraiva (2021), relacionando o problema do sequenciamento do tingimento de tecidos com o PCV, onde as cores dos tecidos são tratadas como cidades, e os tempos de troca entre cores correspondem às distâncias. Essa modelagem matemática baseada no PCV permitiu que o problema fosse analisado e solucionado com maior precisão, resultando na minimização de tempos de setup e aumento da produtividade.

Em problemas de setup dependentes da sequência em máquinas únicas, o PCV fornece uma base teórica para construir matrizes de tempos de transição entre tarefas, que são análogas às matrizes de distâncias no problema clássico do PCV. Essa estrutura facilita a modelagem e a análise do problema, permitindo que soluções sejam encontradas de forma eficiente, mesmo em cenários complexos (Freitas e Fuchigami, 2022).

No caso de um problema de fluxo contínuo sem espera, a modelagem baseada no PCV assimétrico, uma variação do PCV, foi essencial para lidar com a restrição de ausência de tempos de espera. A utilização dessa estrutura matemática permitiu formular o problema de forma precisa e conduzir à minimização do tempo total da operação, mostrando a aplicabilidade do PCV na representação de problemas industriais específicos (Gao et al., 2022).

Além disso, no contexto de sequenciamento de grupos de tarefas, a analogia ao PCV assimétrico também foi utilizado para modelar os tempos de setup dependentes entre grupos e dentro deles. Essa abordagem matemática forneceu uma estrutura sólida para representar as interações entre os grupos, permitindo que o problema fosse resolvido com eficiência e precisão (He et al., 2023).

Na indústria siderúrgica, o sequenciamento de tarefas em um laminador de aço quente foi formulado com base no PCV e problemas relacionados, como o Problema

de Orientação. Essa modelagem permitiu representar matematicamente os tempos improdutivos entre operações, possibilitando a análise e otimização do cronograma de produção, com resultados positivos em termos de redução de tempos ociosos (Kowalski, Kugi e Steinboeck, 2023).

Em problemas que combinam múltiplos objetivos, como eficiência energética e produtividade, o PCV foi utilizado como analogia para modelar problemas de fluxo com permutação mista sem ociosidade. A lógica de representação matemática derivada do PCV ajudou a estruturar o problema, permitindo encontrar soluções que equilibram diferentes objetivos industriais (Zhong et al., 2021).

Já em linhas de montagem flexíveis na Indústria 4.0, o Problema Generalizado do Caixeiro Viajante foi aplicado para modelar configurações de máquinas com base no agrupamento de cidades. Essa estrutura matemática derivada do PCV facilitou a formulação e solução do problema, reduzindo os custos operacionais e melhorando o desempenho em sistemas produtivos modernos (Zerovnik e Herakovic, 2021).

Esses estudos demonstram que a modelagem baseada no PCV não apenas possibilita a estruturação clara de problemas complexos de sequenciamento, mas também oferece uma base lógica e matemática sólida para o desenvolvimento de soluções eficientes. Essa abordagem destaca-se por sua adaptabilidade a diferentes contextos e por promover uma otimização mais assertiva, sistemática e fundamentada.

2.3 MÉTODOS DE RESOLUÇÃO DO PCV

O Problema do Caixeiro Viajante (PCV) pode ser abordado por meio de uma ampla gama de métodos, cada um com suas particularidades, vantagens e limitações. A escolha do método mais adequado depende de uma análise cuidadosa das características da instância do problema, das restrições de tempo e dos objetivos específicos envolvidos (Sathya e Muthukumaravel, 2015).

Os métodos utilizados para resolver o PCV abrangem desde abordagens exatas, como as baseadas em programação linear, que buscam garantir a solução ótima, até métodos mais pragmáticos que utilizam estratégias intuitivas ou regras práticas para casos específicos. Essa diversidade de abordagens reflete a complexidade do problema e a necessidade de equilibrar eficiência computacional e precisão das soluções.

De acordo com Sathya e Muthukumaravel (2015), os métodos empregados na resolução do PCV podem ser classificados em três categorias principais: heurísticos, meta-heurísticos e métodos exatos. A escolha entre essas categorias deve ser feita com base nos requisitos específicos do problema em questão, considerando a complexidade da instância e a necessidade de balancear a qualidade da solução com os recursos disponíveis. Dessa forma, a abordagem ideal será aquela que melhor atender às demandas práticas e teóricas do caso analisado.

2.3.1. MÉTODOS HEURÍSTICOS

Os algoritmos heurísticos utilizam técnicas baseadas em regras práticas, intuições e estratégias específicas para encontrar soluções aproximadas para problemas complexos, como o Problema do Caixeiro Viajante (PCV). Por não buscarem necessariamente a solução ótima, mas sim uma solução satisfatória em um tempo reduzido, esses algoritmos têm se mostrado eficazes para lidar com problemas em que o tempo de processamento ou a complexidade computacional são fatores limitantes (Shahab, 2019).

De maneira geral, algoritmos heurísticos constroem soluções passo a passo, utilizando estratégias que exploram as características específicas do problema. Um exemplo clássico é o algoritmo do vizinho mais próximo, que seleciona, a cada interação, a cidade mais próxima da última visitada, construindo assim uma rota de maneira incremental. Outro exemplo amplamente utilizado é o algoritmo 2-opt, que realiza trocas de arestas em uma rota existente para reduzir o comprimento total, melhorando a qualidade da solução gradativamente. Esses métodos são rápidos, computacionalmente eficientes e frequentemente capazes de produzir soluções subótimas próximas à ótima, dependendo da configuração do problema (Resende et al., 2018).

Além dos exemplos mencionados, uma variedade de outros algoritmos heurísticos também pode ser empregada na resolução do PCV, cada um com abordagens distintas e adequadas para diferentes cenários. Entre os mais conhecidos estão: (Boru et al., 2024; Vianna, 2022; Freitas e Fuchigami, 2022; Placido, Archetti e Cerrone, 2022)

- *Greedy Algorithm (GA)*: constrói a solução de forma gananciosa, escolhendo localmente o melhor passo possível;

- *Nearest Neighbor (NN)*: semelhante ao vizinho mais próximo, baseia-se em selecionar o ponto mais próximo em cada etapa;
- *Cheapest Insertion (CI)*: insere novas cidades na rota no ponto de menor custo incremental;
- *Farthest Insertion (FI)*: prioriza a inserção de cidades mais distantes na rota existente;
- *Nearest Insertion (NI)*: insere a cidade mais próxima de qualquer ponto da rota atual;
- *Random Insertion (RI)*: insere cidades em posições aleatórias na rota;
- *2-Opt/3-Opt*: realiza trocas de duas ou três arestas para melhorar a solução inicial;
- *Lin-Kernighan (LK)*: uma extensão poderosa do 2-opt que explora múltiplas melhorias locais de maneira adaptativa.

Esses algoritmos oferecem uma ampla gama de possibilidades para abordar o PCV, permitindo a escolha do método mais apropriado de acordo com as características e as restrições do problema analisado. Embora geralmente resultem em soluções subótimas, sua rapidez e simplicidade os tornam ferramentas com aplicações fáceis e simples de implementar.

2.3.2. MÉTODOS META-HEURÍSTICOS

As meta-heurísticas são algoritmos projetados para encontrar soluções de alta qualidade, e em alguns casos até soluções ótimas, ao aplicar heurísticas específicas em cada etapa do processo de busca. Essas heurísticas são cuidadosamente adaptadas às características do problema em análise, permitindo uma exploração eficiente e direcionada do espaço de busca em busca de soluções promissoras (Osaba, Yang e Ser, 2020).

Uma das principais vantagens das meta-heurísticas é sua capacidade de superar mínimos locais, um desafio comum em problemas de otimização. Diferentemente de muitos métodos tradicionais, as meta-heurísticas empregam mecanismos que permitem escapar dessas armadilhas, explorando o espaço de busca de maneira mais ampla e, frequentemente, mais eficaz. Isso as torna particularmente úteis em problemas complexos, onde métodos exatos podem se tornar inviáveis devido ao alto custo computacional ou à grande quantidade de soluções possíveis (Osaba, Yang e Ser, 2020).

Existem diversas meta-heurísticas amplamente aplicadas em problemas de otimização, cada uma com abordagens e características específicas. Entre os métodos mais reconhecidos, destacam-se: (Toaza e Esztergár-Kiss, 2023)

- Algoritmo Genético: baseado na evolução biológica, utiliza operações como seleção, cruzamento e mutação para melhorar soluções;
- Otimização por colônia de formigas (*Ant Colony Optimization* - ACO): inspirado no comportamento de formigas em busca de alimentos, utiliza a comunicação baseada em feromônios para explorar soluções;
- Recozimento Simulado (*Simulated Annealing* - SA): imita o processo de recozimento de metais para evitar mínimos locais e encontrar soluções globais;
- Algoritmo de colônia de abelhas (*Artificial Bee Colony algorithm* – ABC): simula o comportamento de colônias de abelhas na busca por alimento, equilibrando exploração e intensificação;
- Otimização por enxame de partículas (*Particle Swarm Optimization* – PSO): inspirado no comportamento coletivo de enxames, como cardumes de peixes, utiliza partículas para explorar o espaço de busca de forma colaborativa;
- Busca Tabu (*Tabu Search* – TS): utiliza uma estrutura de memória para armazenar soluções previamente exploradas e impedir a repetição de trajetórias ineficazes, permitindo escapar de ótimos locais e melhorar a busca global;
- Procedimento de busca adaptativa gananciosa e aleatória (*Greedy Randomized Adaptive Search Procedure* - GRASP): combina construção de soluções baseadas em escolhas gananciosas e busca local para refinamento.

Essas técnicas são amplamente utilizadas devido à sua flexibilidade, eficiência e capacidade de abordar problemas em que a dimensão ou a complexidade tornam outros métodos inviáveis. A escolha da meta-heurística mais adequada depende das características específicas do problema e dos objetivos definidos, sendo fundamental considerar a natureza do espaço de busca e as limitações computacionais do cenário analisado.

2.3.3. MÉTODOS EXATOS

Os métodos exatos de otimização são abordagens baseadas em técnicas rigorosas e precisas, projetadas para resolver problemas de otimização encontrando a solução ótima. Esses métodos garantem a identificação da melhor solução possível

para um problema, independentemente de sua complexidade ou tamanho (Manyam, Sundar e Casbeer, 2020).

O funcionamento desses algoritmos envolve uma busca exaustiva, na qual todas as combinações possíveis de soluções são analisadas. Essa abordagem assegura que o resultado obtido seja globalmente ótimo, tornando esses métodos especialmente relevantes em contextos que exigem alta precisão e confiabilidade, como na tomada de decisões críticas (Vianna, 2022).

Uma das maiores vantagens dos métodos exatos é sua confiabilidade: eles oferecem a garantia de alcançar a solução ótima para o problema em questão. Essa precisão os torna indispensáveis em cenários onde não há margem para erros. Entretanto, a principal limitação dos métodos exatos está no alto custo computacional associado à sua natureza exaustiva. Em problemas de grande escala, como o Problema do Caixeiro Viajante (PCV), o tempo de execução pode se tornar impraticável devido ao crescimento exponencial do número de combinações a serem analisadas. Isso restringe sua aplicação a problemas menores ou a situações em que os recursos computacionais sejam abundantes (Chandra, Natalia e Naro, 2021).

Apesar das limitações, os métodos exatos continuam sendo ferramentas valiosas para a resolução de problemas de otimização. Eles oferecem uma compreensão abrangente da estrutura do problema e são indispensáveis em contextos que priorizam precisão e certeza nos resultados. Além disso, eles fornecem um ponto de referência para avaliar a qualidade de soluções obtidas por métodos heurísticos e meta-heurísticos, sendo frequentemente usados em estudos comparativos e na validação de algoritmos mais eficientes (Cavani, Iori e Roberti, 2021).

Essas características tornam os métodos exatos uma escolha fundamental em cenários que exigem soluções de alta qualidade e confiabilidade, apesar de suas limitações computacionais em problemas mais complexos. Alguns exemplos de métodos exatos são: Programação Linear (PL), Método *Branch and Bound* (BB), Algoritmo *Ford-Fulkerson* (FF), Algoritmo de *Held-Karp* (HK).

2.4 ALGORÍTIMOS GENÉTICOS

2.4.1. ENQUADRAMENTO GERAL

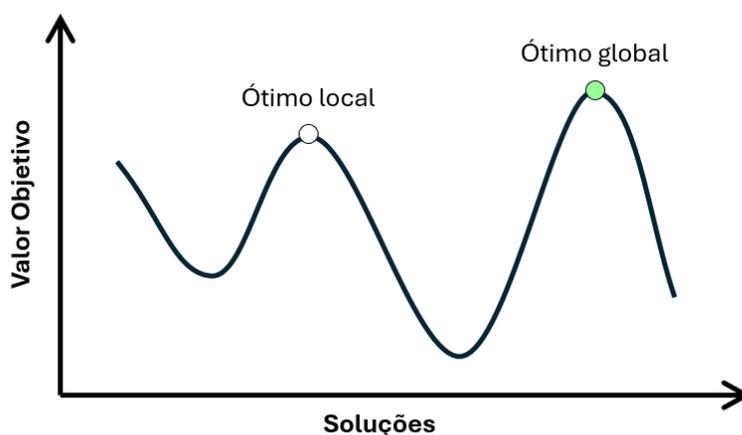
Os algoritmos genéticos (AGs) destacam-se como uma das abordagens mais relevantes no campo da inteligência computacional e da otimização combinatória.

Introduzidos na década de 1970 por John Holland (1992) e seus colaboradores, os AGs foram concebidos como um modelo computacional inspirado na biologia evolutiva, com o objetivo de estudar e compreender processos adaptativos. A base dessa metodologia reside no princípio da teoria da evolução e na seleção natural das espécies de Charles Darwin (2009), segundo o qual os organismos mais adaptados ao ambiente têm maior probabilidade de sobrevivência e reprodução. Esse fundamento biológico foi habilmente adaptado para abordar problemas complexos, particularmente aqueles cuja solução exata é inviável dentro de prazos computacionalmente razoáveis.

Os AGs funcionam por meio de um processo iterativo que simula a evolução natural para encontrar soluções satisfatórias. O ponto de partida é a criação de uma população inicial composta por indivíduos, cada um representando uma solução potencial para o problema. Esses indivíduos são avaliados com base em uma função de aptidão, que mede o grau de adequação de cada solução ao problema em questão. Essa função é o elemento-chave que orienta o algoritmo, permitindo que ele identifique e priorize as áreas mais promissoras do espaço de busca (Wirsansky, 2020).

Após a avaliação inicial, um subconjunto de indivíduos é selecionado para participar do processo de reprodução, onde são aplicados operadores genéticos, como cruzamento e mutação. O cruzamento combina características de dois indivíduos para gerar novas soluções, enquanto a mutação introduz pequenas alterações aleatórias, promovendo diversidade genética. Esses mecanismos são essenciais para evitar que o algoritmo fique preso em ótimos locais, ao mesmo tempo em que permitem a exploração de novas regiões do espaço de busca, como destacado na Figura 3 (Singh e Prakash, 2022).

Figura 3 - Ilustração de ótimo local e ótimo global

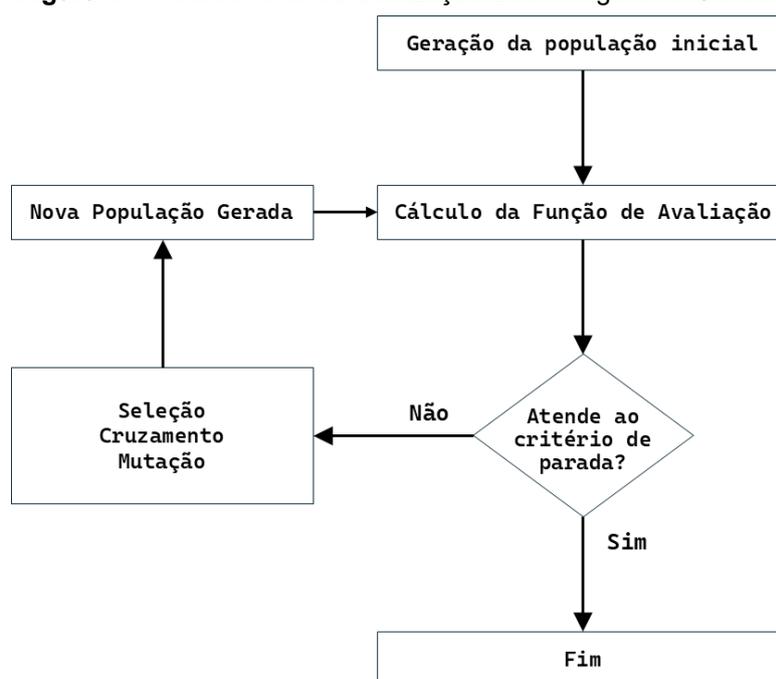


Fonte: Adaptado de Katoch et al. (2020)

Ao longo de diversas gerações, os AGs tendem a convergir para soluções otimizadas. O processo é interrompido quando um critério de parada é alcançado, como a estagnação na qualidade das soluções ou o atingimento de um número máximo de iterações. Esse comportamento iterativo, inspirado pela evolução biológica, torna os AGs uma ferramenta poderosa para enfrentar desafios computacionais complexos e amplamente aplicáveis em diversas áreas da ciência e engenharia (Szabó, 2023).

A Figura 4 a seguir demonstra por meio de fluxograma a lógica de funcionamento de um algoritmo genético de forma genérica.

Figura 4 - Procedimento de otimização de um Algoritmo Genético

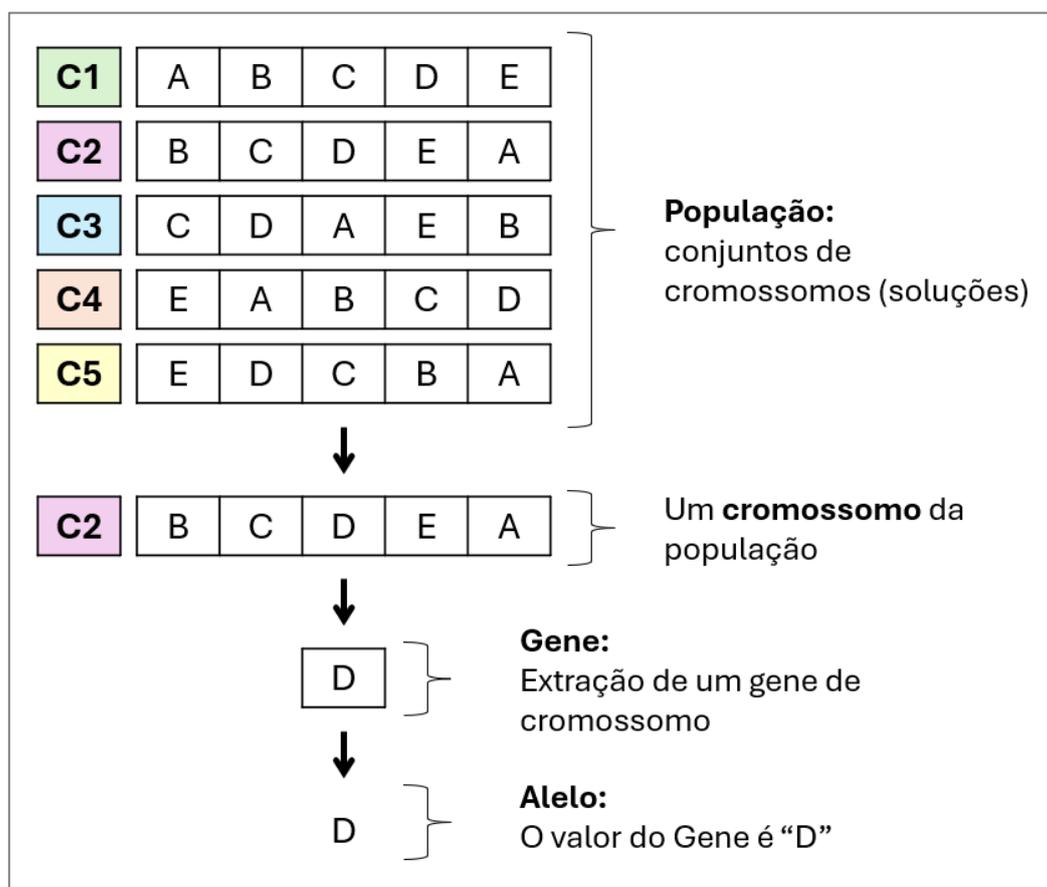


Fonte: Adaptado de Szabó (2023)

2.4.2. CODIFICAÇÃO

A escolha da codificação é um elemento essencial no projeto de um AG, pois define como as soluções são representadas e como os operadores genéticos atuam sobre elas. Em termos biológicos, o cromossomo corresponde à estrutura que armazena as informações genéticas, enquanto os genes representam suas unidades básicas. No contexto computacional, essas analogias são traduzidas para elementos estruturais do problema, conforme ilustrado na Figura 5 (Singh e Prakash, 2021).

Figura 5 – Terminologia dos Algoritmos Genéticos



Fonte: Adaptado de Singh e Prakash (2021)

A definição de um esquema de codificação apropriado é um aspecto fundamental nos algoritmos genéticos, pois influencia diretamente a eficiência das operações genéticas e a qualidade das soluções geradas. Contudo, não existe uma metodologia universal para determinar se um esquema específico é adequado para diferentes problemas, especialmente em aplicações práticas. Cada problema demanda sua própria abordagem de codificação, e é crucial que esses esquemas

sejam planejados para evitar formas redundantes, garantindo maior eficiência na busca por soluções (Katoch et al., 2020)

Entre as abordagens mais comuns de codificação em algoritmos genéticos estão as representações por cadeias binárias, números reais ou estruturas de dados específicas, cada uma escolhida com base na natureza do problema e nas características da solução desejada. Essas diferentes formas de representação não apenas determinam como os indivíduos serão manipulados pelos operadores genéticos, mas também influenciam diretamente a eficácia do processo evolutivo. A escolha do esquema de codificação ideal é, portanto, uma etapa essencial para garantir que as soluções geradas sejam válidas e que a exploração do espaço de busca ocorra de maneira eficiente. (Murthy e Mankame, 2024).

No caso do Problema do Caixeiro Viajante, a codificação permutacional é a mais indicada, pois garante que as soluções representem rotas válidas. Cada indivíduo é modelado como uma sequência de cidades, onde a ordem dos genes determina o percurso. Por exemplo, um cromossomo com a sequência [1, 3, 2, 4] indica que o caixeiro visitará as cidades 1, 3, 2 e 4, nessa ordem. Essa abordagem exige que os operadores genéticos sejam cuidadosamente adaptados, para evitar a geração de indivíduos inviáveis, como rotas com cidades repetidas ou ausentes (Alam et al., 2020).

Além disso, a codificação também influencia a eficiência do algoritmo. Estruturas mal projetadas podem dificultar a aplicação dos operadores genéticos, reduzindo a eficácia do processo evolutivo. Por isso, a escolha da codificação deve considerar tanto as características do problema quanto a capacidade de integração com os operadores genéticos (Alam et al., 2020).

2.4.3. GERAÇÃO DA POPULAÇÃO INICIAL

A geração da população inicial é um aspecto que impacta diretamente a capacidade do algoritmo de explorar o espaço de soluções. Uma população inicial diversificada oferece uma base sólida para a evolução, aumentando a probabilidade de o AG explorar regiões amplas do espaço de busca. Por outro lado, populações iniciais homogêneas tendem a limitar essa exploração, levando à convergência prematura (Obi, 2020).

No PCV, a geração da população inicial é geralmente realizada de forma aleatória. Contudo, estratégias baseadas em heurísticas podem ser empregadas para

acelerar a convergência. No entanto, essas estratégias são menos comuns, pois podem criar preferências por soluções específicas, reduzindo a capacidade do algoritmo de explorar outras áreas do espaço de busca. A diversidade inicial é fundamental nesse contexto, já que o espaço de busca cresce exponencialmente com o aumento do número de cidades. Para problemas de grande escala, é crucial definir um tamanho de população que mantenha o equilíbrio entre diversidade e custos computacionais. Populações muito pequenas podem prejudicar a eficácia do algoritmo, enquanto populações excessivamente grandes podem tornar a execução inviável devido às limitações de tempo e recursos (Ahmed, Choudhary e Al-Dayel, 2024).

2.4.4. FUNÇÃO DE APTIDÃO

A função objetivo, também chamada de função de aptidão ou fitness, desempenha um papel central nos algoritmos genéticos, sendo essencial para avaliar a qualidade de cada indivíduo dentro da população. Essa avaliação é realizada atribuindo uma pontuação que reflete o grau de adequação de cada solução candidata ao problema que se deseja resolver. Em resumo, a função objetivo é o mecanismo que permite ao AG diferenciar entre soluções melhores e piores, orientando o processo evolutivo (Khdeir e Awad, 2025).

De forma prática, a função objetivo serve como o elo que conecta o problema real ao funcionamento do AG. Sua principal tarefa é converter os objetivos do problema em valores numéricos, de modo que cada solução representada por um indivíduo receba um valor que indique o quanto ela atende às condições desejadas. Essa representação numérica facilita a comparação entre diferentes soluções e é determinante para que o algoritmo identifique quais indivíduos possuem maior potencial de contribuir para as próximas gerações (Joshi, 2021).

Além disso, a função objetivo influencia diretamente o processo de seleção dentro do AG. Indivíduos que apresentam valores mais altos na avaliação são mais propensos a serem escolhidos para reprodução, garantindo que características vantajosas tenham maior probabilidade de serem herdadas. Esse processo conduz o algoritmo ao aprimoramento progressivo das soluções ao longo das gerações (Wirsansky, 2020).

2.4.5. OPERADORES GENÉTICOS

Segundo Damia, Esnaashari e Parvizimosaed (2021), os operadores genéticos desempenham um papel fundamental nos algoritmos genéticos, pois são responsáveis por introduzir variação e promover a recombinação das soluções, contribuindo para a exploração eficiente do espaço de busca e a convergência para soluções de qualidade, conforme. A seguir, serão apresentados os operadores genéticos de seleção, cruzamento e mutação.

2.4.5.1. SELEÇÃO

A seleção é um dos operadores fundamentais dos algoritmos genéticos (AGs) e desempenha um papel crucial no processo evolutivo. Sua principal função é regular a transmissão de características genéticas entre as gerações, garantindo que os indivíduos mais aptos tenham maior probabilidade de contribuir com suas características para a composição da próxima população. Esse mecanismo simula, de forma computacional, o princípio darwiniano da sobrevivência do mais apto. (Kaur, 2023).

O papel da seleção vai além de simplesmente escolher os melhores indivíduos. Ela também atua como uma força direcionadora dentro do algoritmo, favorecendo a amplificação das características vantajosas presentes na população. Indivíduos com alta aptidão têm maior chance de serem selecionados, o que aumenta a propagação de suas qualidades para as gerações subsequentes. Essa dinâmica contribui para o refinamento contínuo das soluções, ajudando o AG a explorar o espaço de busca de forma mais eficiente (Sikora e Gryglewicz-Kacerka, 2023).

Existem diversas estratégias de seleção que podem ser empregadas, cada uma com características específicas que afetam o desempenho do algoritmo. Entre as mais comuns, estão:

1. **Seleção por Roleta:** Nesse método, a probabilidade de seleção de cada indivíduo é proporcional ao seu valor de aptidão. Indivíduos com maior aptidão têm maior chance de serem escolhidos, mas soluções menos aptas também podem ser selecionadas ocasionalmente, promovendo diversidade genética. (Szabó, 2023).
2. **Seleção por Torneio:** Uma quantidade fixa de indivíduos é selecionada aleatoriamente, e o mais apto dentro desse grupo é escolhido. Esse método é

eficiente e simples, além de permitir maior controle sobre a pressão seletiva ajustando o tamanho do torneio (Ahmed, Choudhary e Al-Dayel, 2024).

3. **Seleção Aleatória:** Nesse método, os indivíduos são selecionados aleatoriamente da população existente, sem considerar diretamente o valor de sua aptidão. Embora promova alta diversidade genética, essa abordagem carece de pressão seletiva, o que pode resultar na escolha de soluções menos aptas e na demora para alcançar a convergência (Szabó, 2023).
4. **Seleção por Rank:** Os indivíduos são ordenados de acordo com sua aptidão, e as chances de seleção são baseadas na posição no ranking, não no valor absoluto da aptidão. Essa abordagem evita o domínio excessivo de indivíduos extremamente aptos, equilibrando a pressão seletiva (Kaur, 2023).

A escolha da estratégia de seleção adequada depende das características do problema a ser resolvido e do equilíbrio desejado entre exploração e exploração intensiva. Um processo de seleção bem projetado é fundamental para garantir que o algoritmo genético possa convergir para soluções otimizadas de maneira eficiente e robusta.

2.4.5.2. CRUZAMENTO

O cruzamento, também conhecido como crossover, é um dos operadores centrais nos algoritmos genéticos e o principal mecanismo responsável pela recombinação genética. Sua função é gerar novos indivíduos (ou descendentes) que combinam características genéticas de dois indivíduos da população atual, com o objetivo de explorar novas regiões do espaço de busca e aumentar a diversidade das soluções (Dockhorn e Lucas, 2022).

Esse operador é inspirado no processo biológico de reprodução sexual, no qual o material genético dos pais é combinado para formar descendentes com características únicas. No contexto dos AGs, o cruzamento é fundamental para promover a troca de informações entre soluções existentes, possibilitando a criação de indivíduos mais adaptados que incorporam as melhores características dos pais (Khdeir e Awad, 2025).

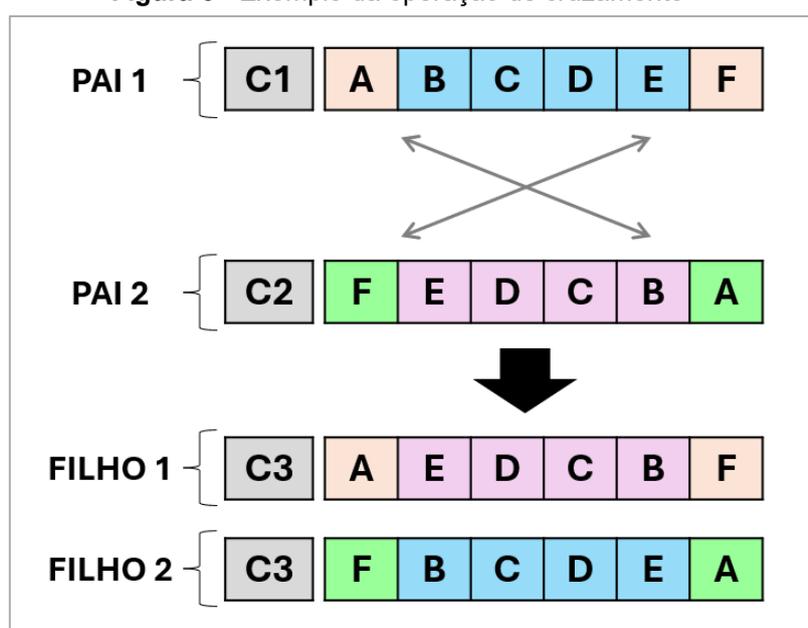
O processo de cruzamento geralmente ocorre de acordo com uma probabilidade pré-definida, chamada de taxa de crossover, que determina a frequência com que esse operador será aplicado. Durante o cruzamento, dois indivíduos, chamados de pais, são selecionados com base em sua aptidão, e uma ou

mais posições no material genético (ou cromossomo) são escolhidas como ponto(s) de divisão. A partir desses pontos, as informações genéticas dos pais são trocadas, resultando em descendentes que apresentam combinações únicas das características originais (Khdeir e Awad, 2025; Mosayebi e Sodhi, 2020).

Existem diversas estratégias para realizar o cruzamento nos AGs, cada uma com vantagens específicas dependendo do problema a ser resolvido. A seguir estão algumas das estratégias de cruzamento que podem ser utilizadas no AG, conforme descrito por Murthy e Mankame (2024) e Rosa et al. (2020):

1. **Cruzamento de Ponto Único:** Um único ponto de corte é escolhido no cromossomo, dividindo-o em duas partes. Os descendentes são formados trocando as porções correspondentes entre os pais.
2. **Cruzamento de Dois Pontos:** Dois pontos de corte são definidos, criando três segmentos no cromossomo. Os descendentes são gerados trocando os segmentos intermediários entre os pais, promovendo maior recombinação.
3. **Cruzamento Uniforme:** Em vez de utilizar pontos fixos, a troca de material genético ocorre em cada posição do cromossomo de maneira probabilística, com cada gene sendo selecionado aleatoriamente de um dos pais.
4. **Cruzamento baseado em Ordem:** Comumente utilizado em problemas de permutação, como o Problema do Caixeiro Viajante, esse método preserva a ordem relativa dos elementos no cromossomo enquanto realiza a recombinação, como demonstrado na Figura 6.

Figura 6 - Exemplo da operação de cruzamento



Fonte: Adaptado de Didden et al. (2023)

2.4.5.3. MUTAÇÃO

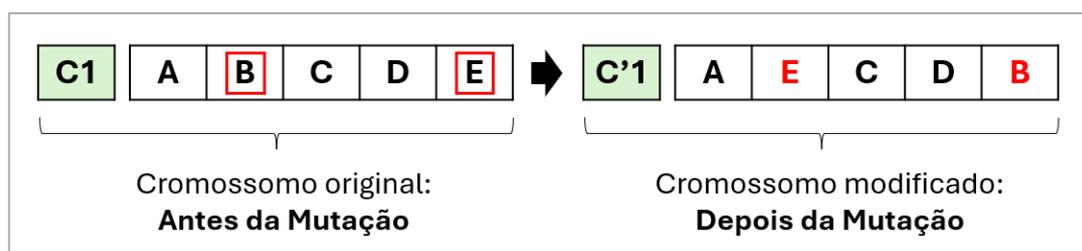
A mutação é um operador fundamental nos algoritmos genéticos e desempenha um papel crucial na introdução de variações na população. Sua principal função é modificar aleatoriamente os "genes" de um indivíduo, garantindo que novas possibilidades sejam exploradas ao longo das gerações. Essa variabilidade genética é essencial para evitar que o algoritmo fique preso em ótimos locais, ou seja, soluções subótimas que podem impedir a descoberta de resultados melhores (Dockhorn e Lucas, 2022).

Conforme abordado por Didden et al. (2023), durante o processo evolutivo, a mutação atua como um mecanismo complementar ao cruzamento. Enquanto o cruzamento combina características de indivíduos selecionados, a mutação altera pequenos segmentos do material genético, permitindo a introdução de soluções inéditas. Essas mudanças ocorrem de maneira aleatória, com uma probabilidade pré-definida chamada taxa de mutação, geralmente mantida em valores baixos. Esse controle é importante para equilibrar a introdução de diversidade sem comprometer a convergência do algoritmo.

A principal vantagem da mutação é sua capacidade de escapar de armadilhas no espaço de busca, como a convergência prematura para ótimos locais. Quando toda a população começa a apresentar características muito similares, a mutação funciona como uma força disruptiva, gerando novos caminhos e ampliando as possibilidades de busca. Esse aspecto é especialmente relevante em problemas complexos, onde o espaço de soluções pode ser vasto e irregular (Han e Xiao, 2022; Ochelska-Mierzejewska, Poniszewska-Marańda e Marańda, 2021).

Existem diversas estratégias para implementar a mutação em algoritmos genéticos, variando de acordo com a representação do problema e o tipo de dados utilizados. Em problemas como o PCV, por exemplo, a mutação pode ser realizada alterando a ordem dos elementos em uma permutação, como ao trocar a posição de duas cidades em uma rota (Figura 7), garantindo a exploração de novas soluções no espaço de busca (Sikora e Gryglewicz-Kacerka, 2023).

Figura 7 - Exemplo da operação de mutação



Fonte: Adaptado de Sikora e Gryglewicz-Kacerka (2023)

Embora a mutação seja uma etapa simples, sua importância não pode ser subestimada. Ela é o principal responsável por manter a diversidade genética na população ao longo das gerações, promovendo um equilíbrio entre exploração de novas áreas do espaço de busca e o refinamento de soluções já existentes. Quando bem ajustada, a mutação garante que o AG seja robusto, eficiente e capaz de encontrar soluções de alta qualidade (Sikora e Gryglewicz-Kacerka, 2023; T. Mzili, I. Mzili e Riffi, 2023).

2.4.6. CRITÉRIO DE PARADA

O critério de parada é um elemento fundamental nos algoritmos genéticos (AGs), responsável por determinar o momento adequado para encerrar o processo de otimização. Ele atua como um mecanismo de controle que evita a execução desnecessária do algoritmo, garantindo eficiência computacional e prevenindo desperdício de recursos (Obi, 2020; Mustafa, 2023).

O principal objetivo do critério de parada é identificar quando o algoritmo atingiu um ponto em que continuar as iterações não trará melhorias significativas ou quando as condições predefinidas para o término foram satisfeitas. A escolha do critério adequado é crucial, pois impacta diretamente a qualidade da solução obtida e o tempo de execução do AG (Alam et al., 2020).

Existem diversos tipos de critérios de parada que podem ser utilizados, dependendo das características do problema e das exigências do sistema. Os mais comuns, descritos por Murthy e Mankame (2024) e Alam et al. (2020), são:

1. **Número Máximo de Gerações:** Nesse critério, o algoritmo é configurado para parar após atingir um número fixo de gerações. É uma abordagem simples e amplamente utilizada, especialmente em casos em que o tempo de execução é uma preocupação importante.

2. **Convergência da População:** O algoritmo para quando a diversidade genética da população diminui significativamente, indicando que os indivíduos estão se tornando muito semelhantes. Isso sugere que o AG convergiu para uma solução e que novas iterações terão pouco impacto.
3. **Estagnação da Aptidão:** Esse critério é baseado na análise da função de aptidão. O processo é encerrado se, após um número pré-determinado de gerações, não houver melhora significativa no valor da melhor solução.
4. **Valor Ótimo Pré-definido:** O algoritmo termina quando encontra uma solução que atenda a um objetivo específico ou alcance um valor de aptidão pré-definido. Esse critério é útil em problemas onde se conhece previamente o resultado desejado.
5. **Tempo Máximo de Execução:** O AG é configurado para encerrar após um limite de tempo definido, garantindo que o processo seja concluído dentro de um prazo específico, independentemente do estado atual da população.

Cada um desses critérios possui vantagens e desvantagens, e a escolha do mais adequado depende do contexto e dos objetivos do problema a ser resolvido.

2.4.7. AMBIENTES PARA IMPLEMENTAÇÃO DO AG

A implementação de AGs exige a criação e configuração de estruturas específicas, como operadores genéticos e critérios de parada, o que pode ser facilitado por uma ampla variedade de ferramentas disponíveis. Os ambientes para implementação de AGs desempenham um papel crucial na escolha de soluções que garantam eficiência, flexibilidade e acessibilidade na resolução de problemas de otimização. Estas ferramentas abrangem desde bibliotecas de código aberto e softwares proprietários até suplementos integrados em planilhas, cada uma com características específicas para atender a diferentes demandas. Neste capítulo, serão apresentadas as principais opções disponíveis, acompanhadas de uma breve descrição dessas ferramentas e alguns exemplos de suas aplicações práticas, oferecendo uma visão abrangente que servirá como base para orientar na escolha do ambiente mais adequado para a implementação do AG no contexto deste trabalho.

2.4.7.1. DISTRIBUTED EVOLUTIONARY IN PYTHON

O Distributed Evolutionary Algorithms in Python (DEAP) é uma biblioteca flexível e modular desenvolvida em Python para a implementação de algoritmos evolutivos. Com suporte a diversas estruturas para representar soluções, como strings, números inteiros e arrays, ela oferece ferramentas robustas para a personalização de indivíduos e operadores genéticos. Um de seus principais diferenciais é a capacidade de paralelizar tarefas, aumentando significativamente o desempenho em problemas computacionalmente complexos. Além disso, por ser de código aberto, o DEAP é amplamente acessível e bem documentado, com compatibilidade para diferentes sistemas operacionais, como Linux, Windows e macOS (DEAP, 2025).

O DEAP tem sido utilizado em diversos estudos para resolver problemas complexos por meio de algoritmos genéticos. No trabalho de Maroof, Ayvaz e Naeem (2024), ele foi integrado a um algoritmo genético híbrido para otimizar o problema de roteamento de veículos com janela de tempo, uma variação do PCV que incorpora restrições adicionais, como janelas de tempo e capacidade dos veículos. Essa abordagem aprimorou a eficiência logística ao reduzir custos e distâncias. Em outro estudo, Raju et al. (2023) aplicaram o DEAP como parte de um modelo que combina algoritmos genéticos e aprendizado ativo para localizar configurações de energia mínima em nanoclusters, acelerando cálculos computacionais complexos. Já no trabalho de Hamida e Benjelloun (2021), o DEAP foi ampliado com técnicas de amostragem ativa para solucionar problemas de aprendizado supervisionado, como detecção de pulsar e ocupação predial, aumentando a eficiência em grandes bases de dados.

Dessa forma, o DEAP se destaca como uma ferramenta poderosa e versátil para a implementação de algoritmos genéticos, atendendo a diversas áreas de pesquisa com eficiência e flexibilidade. Sua capacidade de personalização, paralelismo e acessibilidade o torna ideal para resolver problemas complexos, evidenciando sua relevância acadêmica e prática.

2.4.7.2. MATLAB

O MATLAB, por meio do *Global Optimization Toolbox*, oferece uma solução para a implementação de algoritmos genéticos, permitindo resolver problemas de

otimização em diversos contextos. A biblioteca disponibiliza funcionalidades abrangentes, como configuração de funções de fitness, operadores genéticos, restrições e critérios de parada, além de recursos de visualização que auxiliam no monitoramento do progresso do algoritmo. Apesar de ser um software proprietário da empresa MathWorks, o MATLAB é amplamente reconhecido por sua interface intuitiva e pela integração com outros módulos, o que possibilita análises completas e eficientes em um único ambiente (MathWorks, 2025).

A flexibilidade do MATLAB permite sua aplicação em diversos estudos que utilizam algoritmos genéticos para diferentes finalidades. Por exemplo, no trabalho de Kulenović e Hošić (2021), ele foi empregado para otimizar PCV, criando uma aplicação que calcula rotas otimizadas entre cidades da Bósnia e Herzegovina, com operadores genéticos configurados diretamente no ambiente do MATLAB. Em outro estudo, Fraser e Okonkwo (2021) implementaram um algoritmo genético no planejamento de rotas cíclicas otimizadas entre capitais estaduais na região do Delta do Níger, Nigéria, obtendo eficiência significativa na minimização de distâncias. Além disso, Avgerinos, Manikaros e Vlachou (2023) utilizaram o MATLAB para desenvolver o método GA-MUSA, que integra algoritmos genéticos na análise de satisfação de usuários de plataforma educacional, evidenciando melhorias em relação a métodos tradicionais ao avaliar critérios como funcionalidade e usabilidade.

Esses estudos demonstram a versatilidade e eficácia do MATLAB na aplicação de algoritmos genéticos, destacando sua capacidade de adaptação a diferentes problemas de otimização. Integrando ferramentas avançadas e suporte para personalização, o MATLAB consolida-se como um recurso valioso tanto para pesquisas acadêmicas quanto para aplicações práticas.

2.4.7.3. MICROSOFT EXCEL SOLVER

O Solver é um suplemento do Microsoft Excel projetado para resolver problemas de otimização ajustando os valores de células variáveis, conhecidas como variáveis de decisão. Além de lidar com problemas de otimização tradicionais, ele oferece a funcionalidade de busca evolutiva, permitindo a aplicação de algoritmos genéticos para explorar soluções em problemas onde as relações entre as variáveis não são lineares. Nesse contexto, o algoritmo genético trabalha de forma iterativa, utilizando operadores como mutação e crossover para ajustar as variáveis de decisão, a fim de alcançar o valor ideal na célula de objetivo, sempre respeitando as restrições

definidas. Combinando a simplicidade da interface do Excel com a flexibilidade dos algoritmos evolutivos, o Solver se apresenta como uma ferramenta prática e acessível para problemas de otimização moderados (Microsoft, 2025; Al-Romema e Makarem, 2023).

O Solver do Excel, com sua funcionalidade de algoritmo evolutivo, foi utilizado em diferentes estudos para resolver problemas de otimização. No trabalho de Boru et al. (2024), o Solver foi aplicado ao Problema de Roteamento de Veículos (VRP), uma variação do PCV. O estudo otimizou a distribuição de livros entre bibliotecas de uma universidade, determinando o percurso mais curto e reduzindo custos operacionais. Em outro contexto, Ramsey, Gallemore e Bowling (2024) usaram o Solver para otimizar o agrupamento de alvos em tratamentos de radiocirurgia, melhorando a separação entre os alvos tratados em frações diferentes e aumentando a eficiência do tratamento de metástases cerebrais. Já no trabalho de Al-Romema e Makarem (2023), o Solver foi utilizado para minimizar atrasos no escalonamento de tarefas em uma máquina única, ajustando parâmetros genéticos para melhorar a eficiência na alocação de recursos.

O Solver do Excel é uma ferramenta prática e acessível para resolver problemas de otimização, especialmente em contextos moderados. No entanto, sua capacidade limitada de personalização e restrições de flexibilidade podem dificultar sua aplicação em cenários mais complexos. Ainda assim, sua integração ao Excel e suporte a algoritmos evolutivos garantem sua utilidade em diversas situações.

2.4.7.4. HEURISTICLAB

O HeuristicLab é um software desenvolvido pelo *Heuristic and Evolutionary Algorithms Laboratory* (HEAL), em uso desde 2002, voltado para o desenvolvimento e aplicação de algoritmos heurísticos e evolutivos. Projetado para facilitar a experimentação com diferentes métodos de otimização, ele é amplamente utilizado em contextos acadêmicos e práticos. Sua interface gráfica oferece recursos que permitem a configuração e execução de experimentos de maneira direta, mas sua utilização eficiente pode exigir uma curva de aprendizado inicial, especialmente para usuários menos familiarizados com os conceitos de algoritmos evolutivos (HEAL, 2025).

A ferramenta tem sido empregada em diversos estudos para a aplicação de algoritmos genéticos (AG). Por exemplo, no trabalho de Carreres-Prieto et al. (2022),

o HeuristicLab foi utilizado para desenvolver modelos simbólicos baseados em AG, possibilitando a estimativa de parâmetros como demanda bioquímica de oxigênio e sólidos suspensos totais em estações de tratamento de águas residuais, otimizando o monitoramento em tempo real. Já no estudo de Kinast, Doerner e Rinderle-Ma, 2021, o software foi aplicado para resolver problemas de escalonamento em oficinas industriais, incluindo a alocação de robôs colaborativos. Nessa pesquisa, foi utilizada uma codificação genética especial para otimizar custos de produção e tempos de execução.

Embora o HeuristicLab seja uma ferramenta útil e flexível para experimentação com heurísticas e meta-heurísticas, sua curva de aprendizado inicial e algumas limitações em personalização podem restringir seu uso em cenários mais específicos. Ainda assim, ele se destaca como uma solução relevante e eficaz para problemas de otimização em diferentes áreas.

2.4.7.5. EVOLVER

O Evolver é um suplemento para o Microsoft Excel, desenvolvido pela empresa Palisade, projetado para resolver uma ampla variedade de problemas de otimização modelados em planilhas. Com o uso de algoritmos genéticos e programação linear, ele ajusta variáveis de entrada em modelos configurados no Excel para encontrar combinações ótimas. Suas aplicações abrangem cenários como maximização de lucros, redução de custos e minimização de tempos de execução (Palisade, 2025).

O Evolver tem sido empregado em estudos para a resolução de problemas de otimização com algoritmos genéticos. No trabalho de Chaudhry et al. (2022), ele foi utilizado para escalonar máquinas e veículos guiados automatizados em oficinas flexíveis, otimizando recursos e reduzindo os tempos de produção, com foco na eficiência em ambientes industriais dentro do contexto da Indústria 4.0. Em outro estudo, Aidy et al. (2022) aplicaram AG através do software Evolver para otimizar custos na construção de hospitais, ajustando variáveis como dimensões de colunas e resistência do concreto. O estudo alcançou uma redução significativa nos custos de materiais enquanto cumpria as exigências normativas estruturais. Esses exemplos demonstram a versatilidade do Evolver em atender demandas específicas de diferentes setores.

Embora o Evolver tenha um uso menos disseminado devido a algumas restrições, ele continua sendo uma ferramenta relevante em cenários que valorizam

sua integração direta com o Excel. Sua aplicação prática destaca sua funcionalidade em problemas de otimização, apesar de seu alcance ser mais limitado em relação a ferramentas mais acessíveis e adaptáveis existentes.

3. TRABALHOS RELACIONADOS

Diversos trabalhos têm destacado os algoritmos genéticos como uma abordagem eficiente para resolver problemas complexos de sequenciamento e ordenação, incluindo o PCV e suas variações. Estudos recentes demonstram como os AGs são capazes de lidar com variáveis interdependentes, restrições complexas e altos níveis de dinamismo, otimizando soluções em diferentes cenários. Essas contribuições evidenciam a eficácia dos AGs como ferramentas amplamente aplicáveis e relevantes. A seguir, são apresentados estudos que ilustram a aplicação dessa abordagem na literatura recente.

Um estudo aplicou um AG para resolver problemas de sequenciamento flexível em sistemas job-shop com múltiplas máquinas e tarefas interligadas. O AG foi capaz de identificar sequências produtivas que equilibravam o uso de recursos e reduziam significativamente os tempos de execução das operações, mostrando-se eficaz na resolução de problemas de alta complexidade (Liang et al., 2022).

Outro trabalho aplicou AGs para resolver o problema de sequenciamento de pigmentos em linhas industriais, onde as frequentes trocas de cores geravam atrasos e desperdícios. O AG identificou sequências que reduziram significativamente o tempo de troca entre cores, otimizando o fluxo de produção. Como resultado, houve uma melhora expressiva na eficiência operacional e uma redução nos custos, evidenciando a eficácia do AG na resolução de problemas de sequenciamento de atividades (Houndji e Gna, 2024).

Bojic et al. (2023) aplicaram AGs em um ambiente de planejamento de produção multiobjetivo. O método identificou sequências produtivas eficientes, equilibrando prazos e flexibilidade, com impacto direto na produtividade e no atendimento às demandas do mercado.

Na indústria de calçados, Sadeghi Rebelo e Ferreira (2021) aplicaram AGs para solucionar o problema de sequenciamento em sistemas de montagem de modelos mistos, onde a diversidade de produtos exige ajustes frequentes nas estações de trabalho. O AG foi usado para determinar a melhor ordem de montagem, minimizando

os tempos de troca entre diferentes modelos e reduzindo o impacto das interrupções no fluxo produtivo. A abordagem também ajudou a equilibrar a carga de trabalho entre as máquinas, eliminando gargalos e aumentando a eficiência global das linhas de montagem.

Outro estudo de Wang e Zhu (2021) utilizou AGs para abordar problemas de job-shop flexíveis onde os tempos de preparação variavam conforme a sequência de tarefas. A aplicação do AG permitiu identificar sequências mais eficientes, otimizando o uso das máquinas e reduzindo atrasos acumulados durante a produção. A resolução proporcionou maior fluidez no processo produtivo, comprovando a eficácia dos AGs em gerenciar cenários dinâmicos e complexos da indústria.

No campo da pintura industrial, um AG foi aplicado para otimizar o sequenciamento de cores em linhas automatizadas, reduzindo os tempos de setup entre diferentes tintas. O método organizou a produção para processar cores similares consecutivamente, minimizando trocas e aumentando a eficiência operacional. O estudo destacou a adaptabilidade do AG a diversos tipos de processos produtivos (Yang et al., 2021).

De maneira semelhante, nos trabalhos de Wan et al. (2024), os AGs foram aplicados para explorar diferentes soluções em problemas de sequenciamento de montagem. A abordagem otimizou a ordem de produção, reduzindo significativamente os tempos totais de montagem ao minimizar trocas e ajustes entre etapas. Com isso, houve um aumento expressivo na produtividade das linhas, destacando a eficiência do método em ambientes industriais com alta variabilidade.

Ainda no âmbito de linhas de produção, um AG combinado com busca por vizinhança crítica foi aplicado para sequenciamento distribuído em sistemas job-shop, abordando a alocação eficiente de tarefas entre estações de trabalho. A abordagem conseguiu equilibrar as cargas de trabalho, minimizar atrasos e otimizar o uso dos recursos, demonstrando sua eficácia em ambientes com múltiplas restrições e alta complexidade (Tian et al., 2024).

Além disso, AGs foram utilizados para otimizar o sequenciamento de atividades em máquinas paralelas, considerando tempos de setup dependentes da sequência de tarefas. O estudo demonstrou que o método reduz atrasos ao organizar as tarefas de forma mais eficiente, aproveitando ao máximo a capacidade das máquinas e diminuindo tempos de inatividade (Adan, 2022).

Outro exemplo relevante é o uso de AGs para sequenciamento estocástico, onde variabilidades como tempos de processamento irregulares e mudanças inesperadas no fluxo de produção foram abordadas. O AG foi capaz de ajustar as sequências de forma adaptativa, considerando essas incertezas e gerando soluções robustas que garantiram eficiência e minimização de atrasos em cenários altamente dinâmicos. (Bari Karande e Menezes, 2022).

Problemas relacionados à organização de tarefas em ambientes produtivos foram abordados com AGs adaptativos, que consideraram máquinas reconfiguráveis e módulos auxiliares limitados. O método ajustou a ordem de execução das atividades às capacidades variáveis de cada máquina, otimizando o uso de recursos, reduzindo atrasos e trazendo soluções eficazes para cenários industriais dinâmicos (Fan et al., 2022).

Nos trabalhos de Pedrosa e Puig (2023), os AGs foram utilizados para otimizar o sequenciamento de tarefas em ambientes com *buffers* de capacidade limitada, organizando a ordem de execução de forma a evitar congestionamentos e maximizar a eficiência do fluxo de trabalho. O método ajustou o sequenciamento para que as tarefas fossem processadas dentro das restrições de armazenamento temporário, reduzindo interrupções e garantindo maior equilíbrio e produtividade no sistema produtivo.

A pesquisa de Bashir e Srivastava (2021) destacou como o AG pode ser implementado para o PCV em um ambiente de computação em nuvem, oferecendo uma abordagem robusta para otimizar tarefas de roteamento. O estudo demonstra que, ao combinar métodos clássicos do PCV com a flexibilidade do AG, é possível alcançar soluções eficientes para problemas computacionalmente intensivos em tempo reduzido.

O estudo de Bao et al. (2023) aborda uma variação do clássico PCV, conhecida como *Min-Max Clustered Traveling Salesmen Problem*. Nesse trabalho, o AG é utilizado para minimizar o tempo máximo de percurso entre múltiplos vendedores, uma abordagem que destaca a capacidade do AG de lidar com problemas que exigem equilíbrio de carga entre diferentes agentes. Essa aplicação demonstra a flexibilidade do AG ao adaptar o PCV para cenários com múltiplas restrições e objetivos.

Cahyani e Wiradinata (2023) apresentou a aplicação de um AG integrado ao PCV para recomendar rotas multi-destino, utilizando a API do Google Maps. O estudo destaca a utilização do PCV como base para otimizar trajetos em cenários urbanos,

evidenciando a relevância do AG ao combinar planejamento de rotas com dados geoespaciais em tempo real, proporcionando soluções práticas e eficientes para problemas de transporte e sequenciamento.

Já Gomes et al. (2021) explorou o uso do AG no contexto do problema de roteamento de veículos, adaptando o PCV para atender às demandas logísticas de uma empresa de bebidas. O estudo demonstrou como o AG pode estruturar rotas otimizadas, considerando múltiplos veículos, com foco na eficiência operacional e na melhor utilização da frota.

Em outro estudo, Gözüaçık, Altıok e Gökrem (2021) propuseram uma aplicação inovadora do AG no contexto de navegação indoor utilizando comunicação por luz visível. Embora o foco do trabalho não seja diretamente resolver o PCV, ele faz uso da analogia com o problema para projetar rotas otimizadas em ambientes internos. Isso demonstra como os conceitos do PCV podem ser adaptados para solucionar desafios específicos, destacando a versatilidade do AG em problemas variados.

Ochelska-Mierzejewska, Poniszewska-Maranda e Maranda (2021) aplicaram AGs ao problema de roteamento de veículos, utilizando o PCV como analogia para estruturar e otimizar soluções com restrições operacionais específicas. O estudo destaca como a aplicação de AGs em um problema baseado no PCV pode ser eficaz na obtenção de soluções que conciliam eficiência e viabilidade em contextos complexos.

Em uma abordagem similar, Kulenović e Hošić (2021) utilizaram o AG como ferramenta para resolver diretamente o PCV, demonstrando sua eficácia na busca por soluções otimizadas para problemas de roteamento. O trabalho analisou como os operadores genéticos foram configurados para explorar o espaço de soluções, garantindo um desempenho aprimorado na resolução do PCV.

Placido, Archetti e Cerrone (2022) utilizaram o PCV como uma analogia para resolver desafios relacionados à inspeção de painéis solares. O estudo utilizou um algoritmo genético modificado para lidar com o "*close-enough TSP*", uma variação do PCV original, que considera rotas viáveis sem a necessidade de visitar todos os pontos diretamente, desde que estejam dentro de uma distância aceitável. Essa abordagem permitiu otimizar as trajetórias de inspeção, garantindo maior eficiência no tempo de execução e na cobertura dos locais analisados, atendendo às especificidades do diagnóstico de painéis solares.

Por fim, o estudo de Yuan et al. (2021) explorou o uso de um algoritmo genético para resolver o problema do caixeiro viajante no contexto da logística autônoma com veículos aéreos não tripulados. O trabalho adaptou o TSP para otimizar rotas, considerando restrições operacionais específicas desses veículos, como alcance limitado e capacidade de carga, destacando a flexibilidade do AG em abordar variações práticas do PCV.

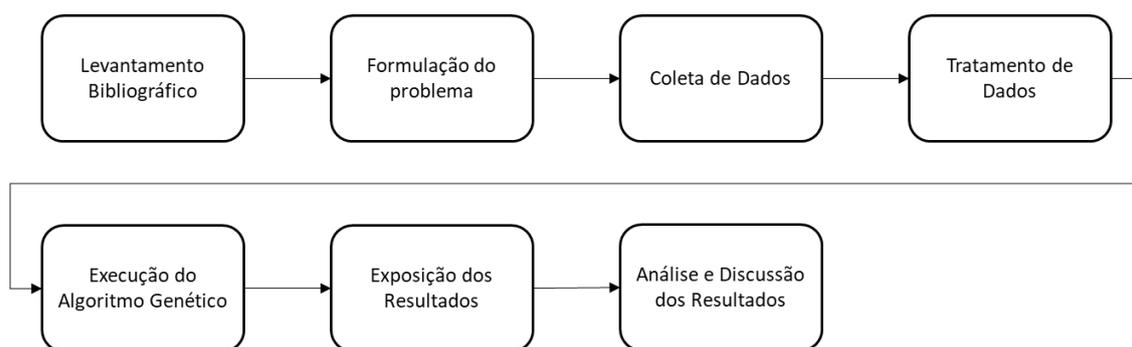
Os estudos analisados demonstram a flexibilidade e a eficácia dos algoritmos genéticos como método para resolver o PCV e suas variantes, além de aplicá-los a desafios complexos de sequenciamento e ordenação em diversos contextos. Essas contribuições fornecem uma base sólida para embasar a aplicação do AG no contexto proposto por este trabalho, destacando sua eficiência e aplicabilidade.

4. METODOLOGIA

4.1 ETAPAS DE DESENVOLVIMENTO

Seguindo a estrutura metodológica proposta por Cauchick-Miguel et al. (2018), que recomenda o encadeamento lógico e sistemático das fases de uma pesquisa aplicada, este trabalho foi conduzido por meio de etapas bem definidas, de modo a garantir rigor metodológico para assegurar a coerência entre os objetivos propostos, os métodos utilizados e os resultados obtidos. Alinhado a essa perspectiva, o presente trabalho foi estruturado em seis etapas principais (Figura 8), descritas a seguir.

Figura 8 - Etapas metodológicas da pesquisa



Fonte: Adaptado de Cauchick-Miguel et al. (2018)

A primeira etapa consistiu no levantamento bibliográfico, com o objetivo de fundamentar teoricamente o estudo, explorando os conceitos relacionados ao Problema do Caixeiro Viajante, ao sequenciamento de produção e aos Algoritmos Genéticos.

Na sequência, foi realizada a formulação do problema, etapa em que se definiram as variáveis relevantes, as restrições e os objetivos da otimização. Com base nessa estrutura, procedeu-se à coleta de dados junto à empresa estudada, englobando informações como o tempo de produção de cada modelo de motocicleta e a quantidade de modelos de motocicletas produzidos em cada linha de montagem.

Os dados obtidos foram então organizados e tratados para viabilizar sua inserção no modelo computacional. Na etapa seguinte, deu-se início à execução do Algoritmo Genético, implementado em linguagem Python com o uso da biblioteca DEAP.

Concluída a execução, os resultados foram expostos e analisados, permitindo a comparação entre diferentes configurações e a identificação dos melhores arranjos de sequenciamento. Por fim, foi realizada a discussão dos resultados, na qual se interpretaram os impactos da solução proposta sobre o processo produtivo, evidenciando as contribuições da abordagem adotada.

4.2 DESCRIÇÃO DO PROBLEMA

A empresa objeto deste estudo enfrenta atualmente a ausência de um método definido para determinar a sequência ideal de produção dos modelos de motocicletas que devem ser montados. Essa falta de padronização no planejamento da sequência produtiva tem gerado ineficiências significativas no processo industrial. Entre as principais consequências estão a subutilização da mão de obra, atrasos no fluxo de trabalho e uma redução na eficiência geral das linhas de montagem.

O problema que será abordado neste estudo está relacionado ao tempo necessário para realizar as trocas de modelos de motocicletas nas linhas de montagem. A empresa opera com três linhas distintas, identificadas como Linha de Montagem A, Linha de Montagem B e Linha de Montagem C, sendo cada uma dedicada à montagem de diferentes grupos de modelos de motocicletas, conforme apresentado na Tabela 1.

Tabela 1 - Modelos de motocicletas e linhas de montagem

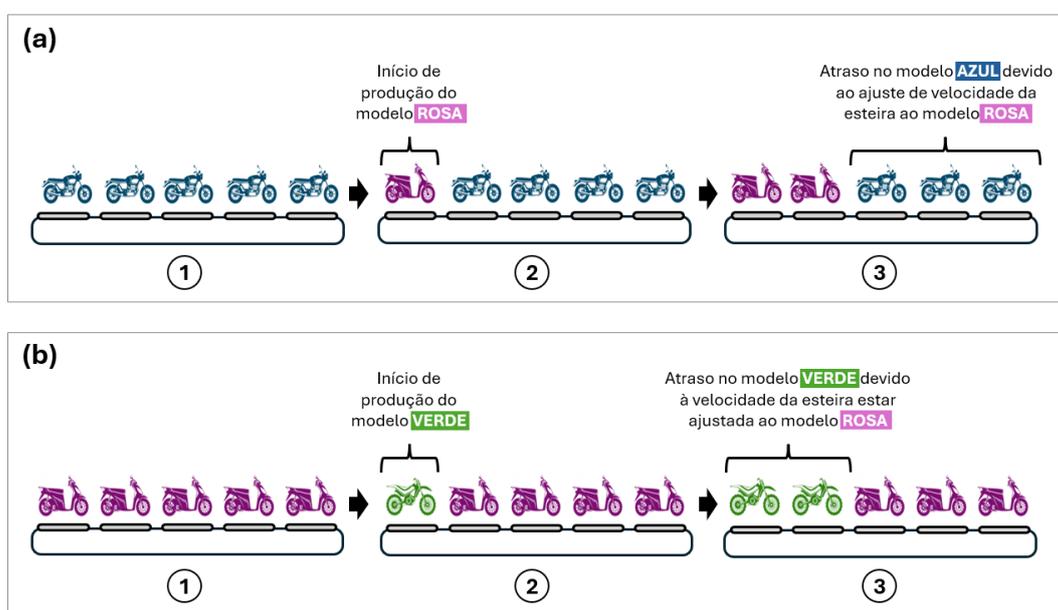
Linha de Montagem	Modelo
A	M1 à M33
B	M34 à M56
C	M57 à M72

Fonte: Elaborado pelo autor (2025)

Cada modelo de motocicleta produzido nas linhas A, B e C possui um ritmo de produção distinto. Por exemplo, o modelo M1 possui um ritmo diferente do modelo M2, que por sua vez possui um ritmo diferente do modelo M3. Essas diferenças de ritmo têm impacto direto no processo de troca de modelos nas linhas de produção.

Devido aos diferentes ritmos de produção, sempre que um novo modelo é introduzido em uma linha de produção, é necessário realizar um ajuste de velocidade na esteira onde as motocicletas são montadas. Esse procedimento pode ocasionar atrasos tanto na montagem do modelo que já está em produção quanto no modelo que está iniciando sua produção, conforme ilustrado na Figura 9.

Figura 9 - Ilustração do impacto entre as trocas de modelos de motocicletas
 (a) Situação 1: Ritmo de produção do modelo rosa maior que o modelo azul.
 (b) Situação 2: Ritmo de produção do modelo verde menor que o modelo rosa.



Fonte: Elaborado pelo autor (2025)

Esses atrasos resultantes dos ajustes de velocidade na esteira representam um desafio significativo para a eficiência da produção. A troca frequente de modelos e os atrasos associados a este processo impactam negativamente o fluxo de trabalho, aumentam o tempo total de produção e reduzem a produtividade geral das linhas de montagem.

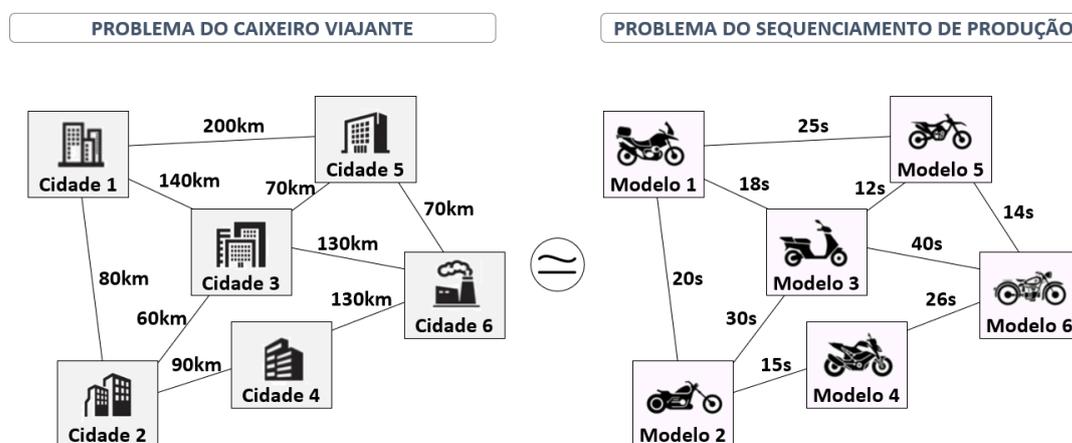
Diante desse cenário, a pesquisa apresenta uma proposta para otimizar a sequência de produção dos modelos de motocicletas, com o objetivo de minimizar os tempos de troca entre os modelos, considerando as particularidades dos diferentes ritmos de produção. Essa abordagem busca oferecer uma base teórica que contribua

para a eficiência do fluxo de trabalho e o aprimoramento do processo produtivo, atendendo às demandas operacionais de forma mais eficiente.

4.3 RELAÇÃO DO PCV COM O PROBLEMA DO SEQUENCIAMENTO DE PRODUÇÃO DE MOTOCICLETAS

O problema do caixeiro viajante e o sequenciamento de produção de motocicletas, apesar de estarem inseridos em contextos distintos, compartilham uma base conceitual semelhante. No PCV, o objetivo é determinar a rota mais curta para visitar todas as cidades, representadas como vértices, considerando as distâncias entre elas, que correspondem às arestas de um grafo. De forma análoga, no sequenciamento de produção, os modelos de motocicletas são os vértices, enquanto os tempos de troca entre eles representam as arestas, conforme demonstrado na Figura 10. Nesse cenário, o foco é encontrar a ordem de produção que minimize o tempo total dos períodos de transição entre as trocas dos diferentes modelos de motocicletas.

Figura 10 - Analogia ilustrativa entre o PCV e o problema do sequenciamento de produção de motocicletas



Fonte: Elaborado pelo autor (2025)

Por conta dessa equivalência estrutural, a modelagem matemática dos dois problemas segue a mesma lógica. Assim, os métodos desenvolvidos para resolver o PCV podem ser diretamente aplicados para resolver o problema do sequenciamento de produção de motocicletas. Essa analogia direta permite adaptar estratégias de otimização, já que ambos os problemas compartilham o mesmo princípio fundamental: minimizar uma soma total, seja de distâncias no PCV ou de tempos no

sequenciamento de produção de motocicletas. Dessa forma, a abordagem lógica e metodológica de solução é válida e eficaz para ambos os contextos.

4.3.1. COMPLEXIDADE DO PROBLEMA

O problema do sequenciamento de modelos de motocicletas proposto neste trabalho utiliza uma analogia direta com o clássico Problema do Caixeiro Viajante, compartilhando os mesmos desafios computacionais associados à sua resolução. Assim como o PCV, o problema do sequenciamento de produção pode ser classificado como um problema NP-difícil, pertencente à categoria de problemas computacionalmente complexos para os quais não existem algoritmos de tempo polinomial conhecidos que os resolvam em tempo hábil (Khdeir e Awad, 2025).

À medida que o número de variáveis ou instâncias do problema aumenta — neste caso, o número de modelos de motocicletas a serem sequenciados —, o espaço de soluções possíveis cresce exponencialmente. Isso significa que o tempo necessário para explorar todas as combinações possíveis torna-se inviável para métodos exatos, especialmente em problemas de grande escala (T. Mzili, I. Mzili e Riffi, 2023)

Para encontrar a solução ótima do problema, é necessário considerar todas as possíveis combinações, o que implica em um elevado custo computacional. Conforme ilustrado na Figura 11, para um problema com um número reduzido de instâncias, como cinco, o tempo de cálculo é extremamente curto. No entanto, à medida que o número de instâncias aumenta, o tempo de processamento cresce rapidamente. Para o problema do sequenciamento de produção analisado neste estudo, com 16, 23 e 33 modelos, os tempos estimados para resolver o problema por métodos exatos seriam, respectivamente, aproximadamente 8 dias, 17,8 milhões de anos e 4,17 sextilhões de anos. Esses valores evidenciam a inviabilidade prática de métodos exatos para resolver problemas dessa magnitude (Al-Romema e Makarem, 2023).

Figura 11 - Relação entre número de instâncias e tempo de processamento

n	Number of paths	Time ($1 \mu s / chemin$)
5	12	$12 \mu s$
10	181,440	0.18 s
15	4.359×10^{10}	12 h
20	6.082×10^{16}	1,928 years
61	4.160×10^{81}	13.19×10^{67} years

Fonte: Dutot e Olivier (2017)

Essa escalabilidade exponencial torna evidente a necessidade de abordar o problema com técnicas alternativas, como os algoritmos genéticos, que podem oferecer boas soluções de forma mais eficiente e viável para problemas complexos. Diante dessa realidade, o problema do sequenciamento de produção proposto neste trabalho é caracterizado por sua alta complexidade, especialmente em função do crescimento exponencial do número de combinações possíveis, como demonstrado pelos cálculos apresentados.

4.4 ESCOLHA DO MÉTODO DE RESOLUÇÃO

Para resolver o problema de sequenciamento de produção de motocicletas, foi necessário selecionar um método que equilibrasse eficiência computacional e qualidade das soluções, considerando o caráter combinatório e de grande escala do problema. Nesse contexto, compararam-se três abordagens: métodos exatos, heurísticos e meta-heurísticos, conforme resumido na Tabela 2.

Tabela 2 - Comparativo entre métodos de resolução

Critério		Método		
Item	Descrição	Meta-Heurísticos	Heurísticos	Exatos
Escalabilidade	Capacidade de lidar com problemas de grande escala de maneira eficiente	Alta	Moderado	Baixa
Qualidade da solução	Garantia de que o método forneça soluções ótimas ou próximas do ótimo	Moderado	Baixa	Alta
Exploração do espaço de busca	Capacidade de investigação do espaço de busca para evitar soluções subótimas.	Alta	Baixa	Alta
Tempo de execução	Capacidade de encontrar soluções dentro de prazos viáveis	Alta	Alta	Baixa
Facilidade de implementação	Simplicidade na aplicação prática e ferramentas ou bibliotecas disponíveis	Moderado	Alta	Baixa

Fonte: Elaborado pelo autor (2025)

Os métodos exatos, embora garantam a solução ótima, possuem baixa escalabilidade e requerem um tempo computacional inviável para problemas de grande escala, enquanto os métodos heurísticos, apesar de rápidos, têm exploração limitada e frequentemente resultam em soluções subótimas e de baixa qualidade. Em

contrapartida, os métodos meta-heurísticos oferecem uma combinação ideal de alta escalabilidade, capacidade de evitar ótimos locais e flexibilidade para lidar com múltiplas restrições. Observando a Tabela 2, fica evidente que os métodos meta-heurísticos apresentam o melhor desempenho geral em critérios fundamentais para problemas como o proposto neste trabalho, tornando-se a escolha mais adequada.

Dentre as meta-heurísticas analisadas, destacam-se quatro algoritmos amplamente utilizados para resolver o PCV, conforme apresentado na Tabela 3. Como o problema de sequenciamento de produção abordado neste estudo possui uma formulação análoga ao PCV, esses algoritmos também podem ser aplicados à sua resolução. A seguir, é feita uma comparação entre essas meta-heurísticas, considerando critérios essenciais para a implementação no contexto específico deste trabalho.

Tabela 3 - Comparativo entre métodos meta-heurísticos

Critério	Método Meta-Heurístico			
	Algoritmo Genético (AG)	Otimização por colônia de formigas (ACO)	Recozimento Simulado (SA)	Busca Tabu (TS)
Capacidade de lidar com o PCV	Atende	Atende	Atende	Atende
Facilidade de implementação	Atende	Não atende	Atende parcialmente	Não atende
Qualidade das soluções para grandes instâncias	Atende	Atende	Atende	Atende

Fonte: Elaborado pelo autor (2025)

A partir dessa análise, observa-se que o ACO e a TS não atendem ao critério de "Facilidade de implementação", pois sua aplicação ao problema desta pesquisa exigiria ajustes específicos para cada uma das três linhas de montagem analisadas, que possuem tamanhos de instância distintos (16, 23 e 33 modelos). Para que esses algoritmos operassem de forma eficiente, seria necessária uma parametrização diferenciada para cada instância, o que tornaria sua implementação complexa e trabalhosa.

O SA, por sua vez, apresenta um comportamento mais flexível, sendo possível configurá-lo uma única vez para todas as instâncias. No entanto, ajustes finos ainda poderiam ser necessários para garantir um melhor desempenho na busca por soluções, o que poderia demandar esforço adicional na calibração dos parâmetros.

Já os AGs se destacam pela capacidade de operar com a mesma parametrização para todas as instâncias. Ao configurar o AG para resolver a maior instância (33 modelos), a mesma configuração pode ser utilizada para as instâncias menores sem perda significativa de eficiência, garantindo simplicidade na implementação sem comprometer a qualidade das soluções. Essa flexibilidade torna o AG a meta-heurística mais eficiente dentre as analisadas.

Outro fator determinante para a escolha dos AGs é o amplo suporte na literatura, que comprova sua eficácia na resolução de problemas similares em diferentes contextos, incluindo em aplicações industriais. Além disso, a ampla disponibilidade de estudos e implementações reforçam sua confiabilidade. Essa combinação de características garante que os AGs sejam uma escolha robusta e eficiente para o problema estudado.

Assim, os algoritmos genéticos foram selecionados como método de resolução por sua robustez, flexibilidade e desempenho comprovado, atendendo de forma eficiente às demandas e à complexidade do problema estudado.

4.5 ESCOLHA DO AMBIENTE DE DESENVOLVIMENTO

A seleção do ambiente de desenvolvimento para a implementação do algoritmo genético neste trabalho foi baseada na análise de ferramentas amplamente utilizadas na literatura recente: DEAP, MATLAB, Microsoft Excel Solver, HeuristicLab e Evolver. Para determinar a opção mais adequada ao problema estudado, foram estabelecidos quatro critérios essenciais: flexibilidade, eficiência, usabilidade e custo.

A Tabela 4 a seguir apresenta a comparação das ferramentas avaliadas, indicando o nível de atendimento de cada uma aos critérios estabelecidos:

Tabela 4 - Comparativo entre as ferramentas para implementação do AG

Critério		Ferramenta				
Item	Descrição	DEAP	MATLAB	Excel Solver	Heuristic-Lab	Evolver
Flexibilidade	Ajuste de parâmetros e operadores	Atende	Atende parcialmente	Não atende	Atende parcialmente	Atende parcialmente
Eficiência	Tempo de processamento de problemas complexos	Atende	Atende	Não atende	Atende	Atende parcialmente
Usabilidade	Facilidade de uso e implementação	Atende parcialmente	Atende parcialmente	Atende	Atende parcialmente	Atende
Custo	Disponibilidade gratuita ou com custo acessível	Atende	Não atende	Atende	Atende	Não atende

Fonte: Elaborado pelo autor (2025)

Com base nessa análise, optou-se pela utilização do DEAP (Distributed Evolutionary Algorithms in Python), por apresentar o melhor equilíbrio entre os critérios avaliados. Ainda que suas capacidades já tenham sido abordadas na Seção 2.4.7.1, é importante destacar aqui que a escolha se fundamenta também em sua ampla aplicação em estudos recentes, que atestam sua robustez prática.

Por exemplo, Raju et al. (2023) demonstraram a eficácia do DEAP ao utilizá-lo em conjunto com aprendizado ativo para localizar configurações globais de energia mínima em nanoclusters metálicos, o que permitiu acelerar significativamente a convergência com menor custo computacional.

De modo semelhante, Pistore (2017) mostrou como a flexibilidade no ajuste de operadores foi essencial para aumentar a assertividade na predição de promotores em sequências genéticas, reforçando a necessidade de ambientes configuráveis como o DEAP.

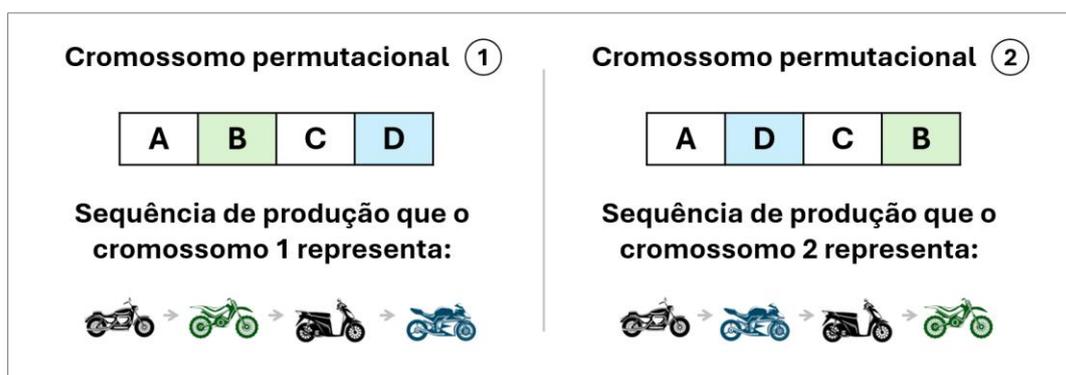
Assim, a escolha do DEAP nesta etapa metodológica não se restringe às suas qualidades intrínsecas — já detalhadas anteriormente —, mas se justifica principalmente por seu histórico de aplicações bem-sucedidas em contextos semelhantes ao do presente estudo, unindo capacidade técnica, flexibilidade e respaldo acadêmico.

4.6 FUNCIONAMENTO DO AG NO PROBLEMA DE SEQUENCIAMENTO DE PRODUÇÃO DE MOTOCICLETAS

Para entender o funcionamento do AG no contexto do problema de sequenciamento de produção de motocicletas, é essencial decompor o processo em suas principais etapas: codificação, inicialização, avaliação, seleção, cruzamento, mutação e critério de parada.

Primeiramente, em relação à codificação do problema, no AG, as possíveis soluções são representadas como cromossomos. No caso do problema de sequenciamento de produção, cada cromossomo é uma permutação de números inteiros, onde cada número representa um modelo de motocicleta. Por exemplo, um cromossomo pode ser representado como [A, B, C, D], indicando a ordem na qual os modelos serão produzidos (Figura 12). Essa codificação permite que o algoritmo explore diferentes sequências produtivas aceitáveis na busca por uma solução otimizada (Sikora e Gryglewicz-Kacerka, 2023).

Figura 12 - Exemplo de cromossomo permutacional e sua sequência representativa



Fonte: Elaborado pelo autor (2025)

A etapa seguinte consiste na inicialização da população, que envolve a criação de múltiplos cromossomos gerados aleatoriamente. A geração de uma população inicial diversificada é fundamental para garantir que o algoritmo tenha um amplo espaço de busca, aumentando a probabilidade de encontrar boas soluções. No estudo em questão, essa população inicial será composta por diversas sequências de produção de modelos de motocicletas criadas de forma aleatória, permitindo que o AG inicie o processo de otimização com diferentes possibilidades de sequenciamento de produção dos diferentes modelos de motocicletas (Obi, 2020; Szabó, 2023).

Uma vez gerada a população, cada cromossomo é avaliado utilizando uma função de aptidão, cuja finalidade é medir a eficiência do sequenciamento em termos

do tempo de troca entre modelos. Essa função calcula o tempo total necessário para a transição de um modelo para outro ao longo da linha de montagem, considerando os intervalos de ajuste necessários entre os diferentes tipos de motocicletas. Soluções que resultam em menores tempos de troca são consideradas mais aptas, pois garantem maior eficiência produtiva e redução de desperdícios no processo.

Para garantir que a busca por soluções mais eficientes continue ao longo das gerações, são aplicados os operadores de seleção, cruzamento e mutação. O processo de seleção tem como objetivo escolher os cromossomos mais aptos para reprodução, aumentando a probabilidade de que suas características sejam transmitidas às gerações seguintes. O cruzamento ocorre combinando partes dos cromossomos selecionados para gerar novas soluções, promovendo a exploração do espaço de busca. Já a mutação introduz pequenas alterações aleatórias nos cromossomos, aumentando a diversidade da população e reduzindo o risco de convergência prematura para soluções subótimas.

Todo esse processo evolutivo se repete até que seja atendido um critério de parada, que determina o momento em que o algoritmo deve encerrar a busca por soluções aprimoradas. Quando o critério de parada é atingido, o AG retorna a melhor solução encontrada para o sequenciamento da produção de motocicletas.

Abaixo, o fluxo do algoritmo genético para o problema de sequenciamento da produção de motocicletas é apresentado de forma sistematizada:

- 1) Inicialização: Gerar uma população inicial de cromossomos aleatórios;
- 2) Avaliação: Calcular a aptidão de cada cromossomo utilizando a função de avaliação;
- 3) Seleção: Selecionar os cromossomos mais aptos para reprodução;
- 4) Cruzamento e Mutação: Aplicar os operadores genéticos para gerar novos cromossomos filhos.
- 5) Nova População: Substituir a população atual pelos novos cromossomos gerados.
- 6) Critério de Parada: Repetir os passos de 2 a 5 até que o critério de parada seja atendido.

4.7 CÁLCULO DO TEMPO ENTRE AS TROCAS DE MODELOS

Para determinar o tempo necessário entre as trocas de modelos de motocicletas nas linhas de produção, utiliza-se a seguinte equação:

$$T_{maior} - T_{menor} \times QC \quad (1)$$

Nessa equação, T_{maior} representa o tempo de produção do modelo com maior duração, T_{menor} corresponde ao tempo do modelo com menor duração, e QC refere-se à quantidade de cavaletes (espaços na linha de montagem onde as motos são montadas) disponíveis na linha de produção. Essa fórmula permite calcular com precisão o intervalo entre as trocas de modelos, considerando as variações nos tempos de produção e a estrutura específica de cada linha de montagem.

A aplicação desse cálculo exige a consideração das características individuais de cada linha de produção, incluindo o número de cavaletes e os tempos de produção de cada modelo de motocicleta. A Tabela 5 apresenta a quantidade de cavaletes disponíveis em cada linha de montagem, enquanto a Tabela 6 exibe os tempos de produção de cada modelo.

Com base nesses dados e na equação fornecida, é possível construir a matriz de tempos entre as trocas de modelos, semelhante à matriz de distâncias utilizada no PCV. Essa matriz organiza de forma estruturada os tempos de transição entre cada par de modelos. Essa informação é fundamental para modelar o problema de forma adequada, possibilitando a aplicação do método de otimização proposto neste estudo.

Tabela 5 - Quantidade de cavaletes por linha de montagem

Linha de Montagem	Número de Cavaletes
A	20
B	80
C	118

Fonte: Elaborado pelo autor (2025)

Tabela 6 - Tempo de produção dos modelos de motocicletas

Modelo	Tempo (s)	Modelo	Tempo (s)	Modelo	Tempo (s)
M1	142	M25	268	M49	46
M2	143	M26	226	M50	47
M3	143,3	M27	266	M51	43,2
M4	145	M28	267	M52	43,7
M5	145,8	M29	286	M53	50,7
M6	147,8	M30	300	M54	47,6
M7	149,1	M31	300,5	M55	54,6
M8	149,4	M32	305,8	M56	48
M9	150,1	M33	310	M57	19,7

Modelo	Tempo (s)	Modelo	Tempo (s)	Modelo	Tempo (s)
M10	150,7	M34	35	M58	22
M11	151,4	M35	21	M59	19,1
M12	151,4	M36	36	M60	27
M13	152,9	M37	39	M61	29
M14	206	M38	34	M62	19
M15	207	M39	24	M63	31
M16	216	M40	23	M64	30
M17	222	M41	40	M65	28,5
M18	349,7	M42	38	M66	19,4
M19	350	M43	43	M67	28
M20	357,5	M44	42,8	M68	20
M21	359	M45	44	M69	33,2
M22	379,6	M46	40,6	M70	33
M23	381	M47	42	M71	32
M24	261	M48	42,5	M72	27,6

Fonte: Elaborado pelo autor (2025)

4.8 DEFINIÇÃO DOS PARÂMETROS DO ALGORITMO GENÉTICO (CONDIÇÕES DE CONTORNOS)

Na implementação de um algoritmo genético, é necessário levar em consideração alguns fatores essenciais. Entre eles, destacam-se a função de avaliação, responsável pela classificação dos indivíduos, além dos métodos de seleção, cruzamento e mutação escolhidos. Também é preciso definir outros parâmetros, como o tamanho da população e o critério de parada a ser adotado neste estudo.

Serão discutidos os aspectos técnicos da implementação, bem como o detalhamento dos parâmetros selecionados. Para o desenvolvimento deste trabalho, foi utilizada a linguagem de programação Python, com suporte da biblioteca DEAP. A execução do código ocorreu em um computador equipado com processador Intel Core i7-9750H e 16 GB de memória RAM.

4.4.1 FUNÇÃO DE AVALIAÇÃO

Para abordar o problema do sequenciamento de produção dos modelos de motocicletas, será implementada uma função de avaliação focada na minimização do tempo total de troca entre os modelos na linha de montagem. Esta função é crucial para avaliar a aptidão dos indivíduos gerados pelo algoritmo genético e determinar a eficiência de cada solução proposta.

Matematicamente, conforme descrito por Khdeir e Awad (2025), essa função de avaliação pode ser definida como:

$$eval(P) = \sum_{\forall(x,y) \in P} dist(x_i, y_i) \quad (2)$$

Onde:

- P representa uma sequência de produção específica;
- (x_i, y_i) é um par de modelos consecutivos na sequência P;
- $dist(x_i, y_i)$ corresponde ao tempo necessário para realizar a troca entre os modelos x_i e y_i .

Essa equação permite obter o valor total dos tempos de troca de uma determinada sequência de produção dos modelos de motocicleta na linha de montagem. No algoritmo genético, a função de avaliação fornece o valor necessário para a função de fitness, que é usada para comparar as diferentes soluções candidatas e identificar qual delas apresenta a menor soma dos tempos de troca.

Ao minimizar $eval(P)$, o algoritmo genético prioriza as sequências mais eficientes. Dessa forma, a função de avaliação desempenha um papel crucial na orientação do processo de busca do AG, permitindo que o algoritmo encontre soluções mais adequadas para o problema de sequenciamento da produção dos modelos de motocicletas deste estudo.

4.4.2 TAMANHO DA POPULAÇÃO

O tamanho da população é um fator crucial em algoritmos genéticos, impactando diretamente a capacidade de exploração e a convergência para soluções ótimas. Neste estudo, optou-se por uma população de 100 indivíduos, buscando equilibrar a diversidade genética com a eficiência computacional, essencial para problemas de sequenciamento de produção de complexidade moderada.

Pistore (2017) sugere que populações de 200 indivíduos oferecem maior diversidade e melhoram a qualidade das soluções, enquanto Sikora e Gryglewicz-Kacerka (2023) indicam que 50 indivíduos são suficientes para obter bons resultados em termos de tempo de execução. No entanto, escolher 100 indivíduos como um valor intermediário permite um equilíbrio adequado, conforme demonstrado por Al-Furhud e Ahmed (2020), que destacam o sucesso desse tamanho de população em seus

estudos. Essa escolha evita tanto a limitação de diversidade quanto o aumento exagerado no tempo de processamento, proporcionando uma execução eficiente do algoritmo.

4.4.3 SELEÇÃO

O método de seleção por torneio com tamanho dois foi escolhido para definir os indivíduos da próxima geração devido à sua simplicidade e eficiência no algoritmo genético. Nesse processo dois indivíduos são selecionados aleatoriamente da população para competir e o mais apto entre eles é escolhido como pai. Esse procedimento é repetido até que o número necessário de pais seja alcançado garantindo uma seleção eficiente e mantendo a diversidade (Murthy e Mankame, 2024).

A escolha de um torneio com tamanho dois oferece um equilíbrio ideal entre pressão seletiva e manutenção da diversidade. Ele promove a seleção de indivíduos mais aptos sem excluir totalmente aqueles com menor desempenho permitindo maior exploração do espaço de busca e reduzindo o risco de convergência prematura, conforme descrito por Kuroswiski et al. (2021).

Portanto, a seleção por torneio com tamanho dois foi adotada para combinar simplicidade, controle da pressão seletiva e diversidade, garantindo eficiência na busca por soluções de alta qualidade.

4.4.4 CRUZAMENTO

O método de cruzamento adotado é o *Partially Matched Crossover* (PMX), escolhido por sua eficiência na resolução de problemas de permutação, como o PCV. Nesse método, segmentos correspondentes de dois pais são trocados para gerar dois novos indivíduos, garantindo que a posição relativa dos elementos seja preservada e mantendo a viabilidade das soluções geradas (Murthy e Mankame, 2024).

A taxa de cruzamento definida é de 80%, o que significa que a maioria dos indivíduos da população passará pelo processo de recombinação. Esse valor foi escolhido para promover uma ampla diversidade entre os descendentes, favorecendo a exploração do espaço de busca e evitando que o algoritmo fique preso em soluções subótimas prematuramente.

O PMX se destaca por sua capacidade de manter características estruturais dos pais, minimizando a introdução de soluções inviáveis, algo essencial em

problemas de permutação onde a repetição de elementos deve ser evitada (Khdeir e Awad, 2025).

Portanto, a escolha do PMX como operador de cruzamento se justifica por sua robustez, eficiência na manipulação de sequências de modelos de produção e capacidade de preservar a estrutura das soluções, minimizando a geração de configurações inviáveis, fatores essenciais para a otimização do problema abordado neste trabalho (Kuroswiski et al., 2021).

4.4.5 MUTAÇÃO

O método de mutação adotado é o *Order Based Mutation* (OBM), no qual uma subsequência de genes é selecionada e reordenada aleatoriamente, preservando a estrutura do problema sem gerar soluções inválidas.

A taxa de mutação de 10% foi definida para equilibrar a exploração de novas soluções e a preservação das melhores características já encontradas. Esse valor evita que o algoritmo fique preso em ótimos locais sem comprometer a convergência para boas soluções (Mosayebi e Sodhi, 2020).

Esse método é especialmente adequado para problemas de permutação, como o PCV, pois altera a ordem dos elementos sem invalidar a solução. Além disso, contribui para a diversidade da população e reduz o risco de estagnação, garantindo uma busca mais eficiente por soluções otimizadas (Nikfarjam, A. Neumann e F. Neumann, 2021)

4.4.6 CRITÉRIO DE PARADA

O critério de parada será definido pelo número máximo de gerações, estabelecido em 500 gerações. Esse limite foi escolhido para garantir um equilíbrio entre exploração e convergência, permitindo que o algoritmo percorra um espaço amplo de soluções sem prolongar excessivamente o tempo de processamento.

Um número maior de gerações aumenta a probabilidade de encontrar a solução ótima, pois permite que o algoritmo continue refinando as soluções ao longo do tempo. No entanto, definir um limite é essencial para evitar execuções desnecessariamente longas e garantir um custo computacional viável. Segundo Wirsansky (2020), a escolha de um número máximo entre 200 e 500 gerações é comumente eficaz para problemas como o PCV, pois proporciona um refinamento suficiente das soluções sem impactar significativamente a eficiência do algoritmo.

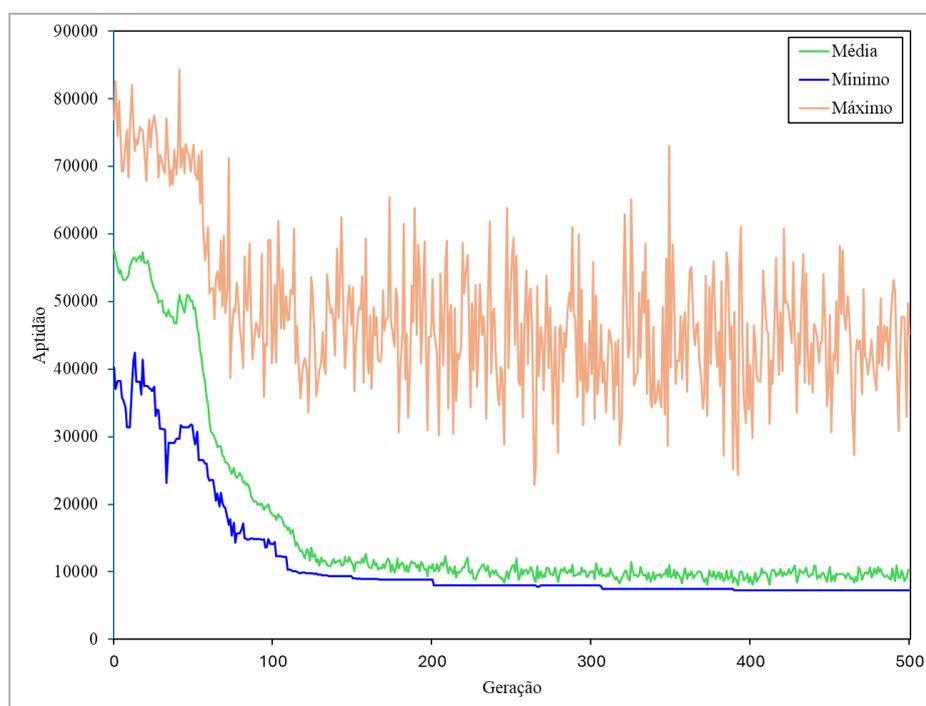
Ao adotar 500 gerações, busca-se garantir que o processo evolutivo alcance um nível adequado de refinamento, permitindo que as melhores soluções sejam preservadas e continuamente aprimoradas. Esse critério também evita a convergência prematura, aumentando a chance de escapar de mínimos locais e encontrar soluções de melhor qualidade. Além disso, esse número foi definido considerando um bom custo-benefício entre tempo de execução e qualidade das soluções, garantindo que o algoritmo produza resultados robustos e eficientes para o problema analisado.

5. RESULTADOS E DISCUSSÃO

5.1 LINHA DE MONTAGEM A

Para a Linha de Montagem A, os resultados são apresentados na Figura 13, que ilustra a evolução das aptidões máxima, média e mínima ao longo das gerações. O AG passou por um período inicial de grande variação, característico da exploração do espaço de busca, seguido por uma estabilização progressiva. A convergência foi alcançada na geração 390, momento em que a aptidão se estabilizou, indicando que o algoritmo encontrou uma solução robusta e consistente.

Figura 13 - Evolução das soluções do AG para a Linha de Montagem A



Fonte: Elaborado pelo autor (2025)

Durante o processo, a aptidão média apresentou um declínio acentuado nas gerações iniciais, refletindo o aprimoramento progressivo das soluções. A aptidão

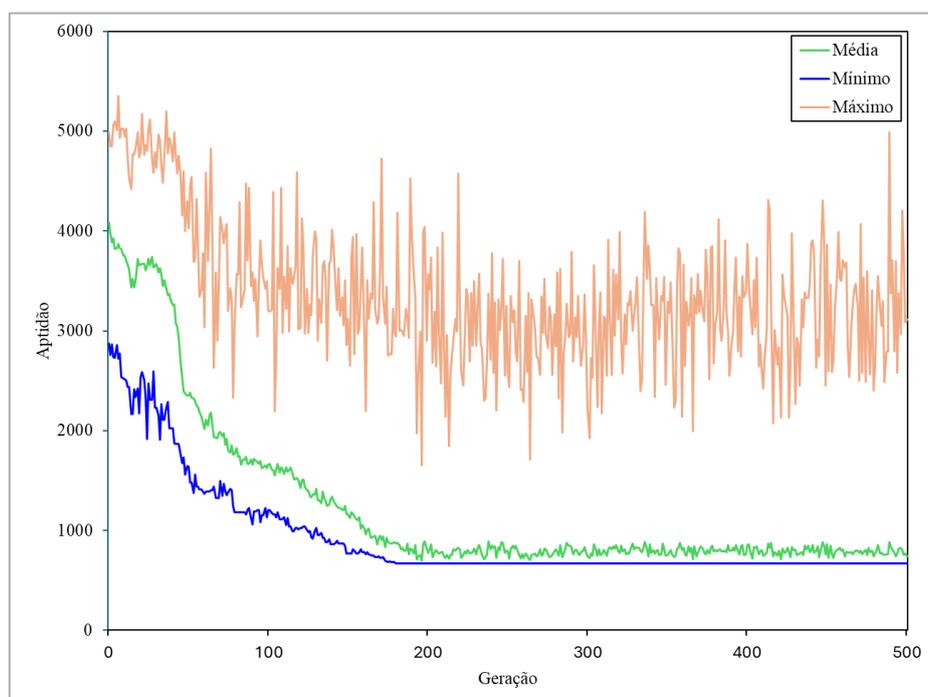
máxima, por outro lado, manteve certa variabilidade, sugerindo a presença de indivíduos fora do padrão otimizado, mas que contribuíram para a diversidade da população.

A sequência de produção otimizada foi definida como: M1, M2, M3, M4, M9, M8, M6, M5, M7, M10, M11, M12, M13, M14, M15, M16, M17, M26, M24, M27, M29, M30, M31, M32, M33, M18, M19, M21, M23, M22, M20, M25, M28. Essa sequência foi projetada para minimizar o tempo necessário para as trocas entre modelos, totalizando 7.232 segundos, um valor que demonstra a eficiência do AG na otimização deste processo produtivo.

5.2 LINHA DE MONTAGEM B

Na Linha de Montagem B, os resultados estão representados na Figura 14, que também mostra a evolução das aptidões máxima, média e mínima ao longo das gerações. O AG apresentou uma convergência rápida, estabilizando-se na geração 180, quando a aptidão deixou de ser aprimorada. Esse comportamento reflete a capacidade do algoritmo de encontrar soluções eficientes em um número reduzido de gerações para um problema com menor número de instâncias (modelos de motocicletas).

Figura 14 - Evolução das soluções do AG para a Linha de Montagem B



Fonte: Elaborado pelo autor (2025)

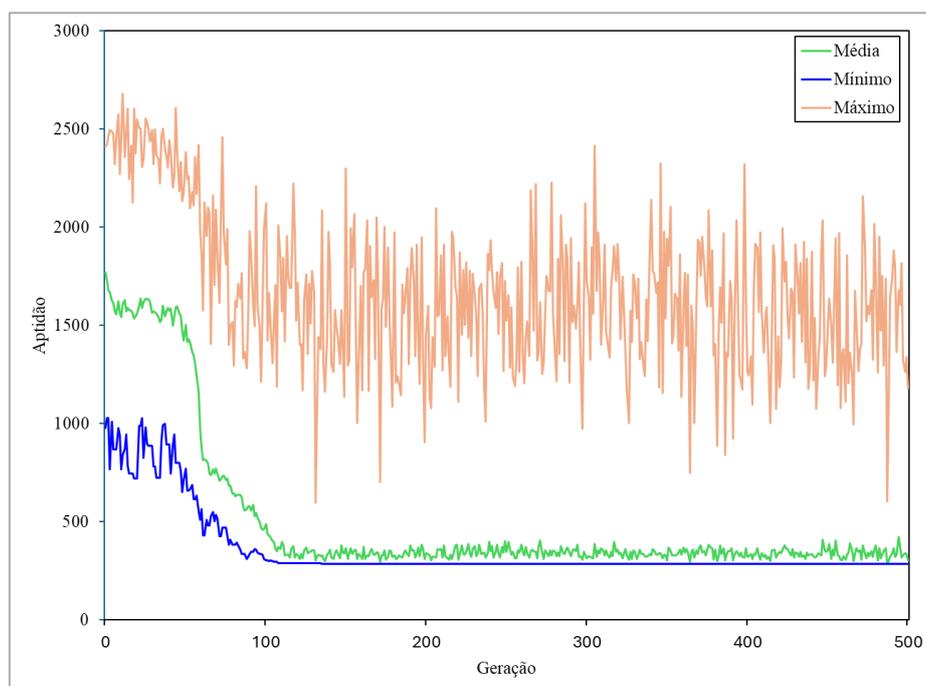
O gráfico evidencia um ajuste inicial intenso, com grandes variações nas aptidões, especialmente na máxima, que permaneceu mais oscilante. Em contrapartida, as aptidões média e mínima declinaram rapidamente, estabilizando-se em valores que indicam a obtenção de uma solução robusta.

A sequência de produção otimizada gerada foi: M55, M53, M56, M54, M50, M49, M45, M52, M51, M43, M44, M48, M47, M46, M41, M37, M42, M36, M34, M38, M39, M40, M35. O tempo total necessário para as trocas entre os modelos foi de 672 segundos, demonstrando uma otimização expressiva e reforçando a eficácia do AG em promover eficiência temporal no processo produtivo.

5.3 LINHA DE MONTAGEM C

Os resultados para a Linha de Montagem C são apresentados na Figura 15, que retrata a evolução das aptidões ao longo das gerações. O AG demonstrou um desempenho de rápida convergência, alcançando a estabilização na geração 135. A aptidão mínima estabilizou-se rapidamente, indicando que o algoritmo foi capaz de encontrar uma solução otimizada em um curto intervalo de tempo. Esse comportamento é particularmente notável quando se considera um menor número de modelos envolvidos, o que reduz a complexidade do espaço de busca.

Figura 15 - Evolução das soluções do AG para a Linha de Montagem C



Fonte: Elaborado pelo autor (2025)

As curvas no gráfico revelam um declínio acentuado nas aptidões média e mínima durante as primeiras gerações, representando o refinamento das soluções. A estabilização da aptidão reforça a robustez da solução encontrada, alinhada aos critérios estabelecidos.

A sequência de produção otimizada para a Linha de Montagem C foi: M62, M59, M66, M57, M68, M58, M60, M72, M67, M65, M61, M64, M63, M71, M70, M69. O tempo total necessário para as trocas entre modelos foi de 284 segundos, o que representa uma otimização significativa, evidenciando o impacto positivo do AG na redução de tempos de trocas entre os modelos.

5.4 COMPARATIVO ENTRE SEQUÊNCIA OTIMIZADA E ALEATÓRIA

Para avaliar a eficácia do AG na otimização do sequenciamento de produção, foi realizado um comparativo entre os resultados obtidos a partir do sequenciamento otimizado pelo AG e os resultados gerados por uma sequência aleatória para cada uma das três linhas de montagem analisadas. A Tabela 7 apresenta, para cada linha de montagem, a ordem de produção dos modelos de motocicletas obtidos para ambas as abordagens.

Tabela 7 - Comparação das sequências de produção: AG vs. Sequência Aleatória

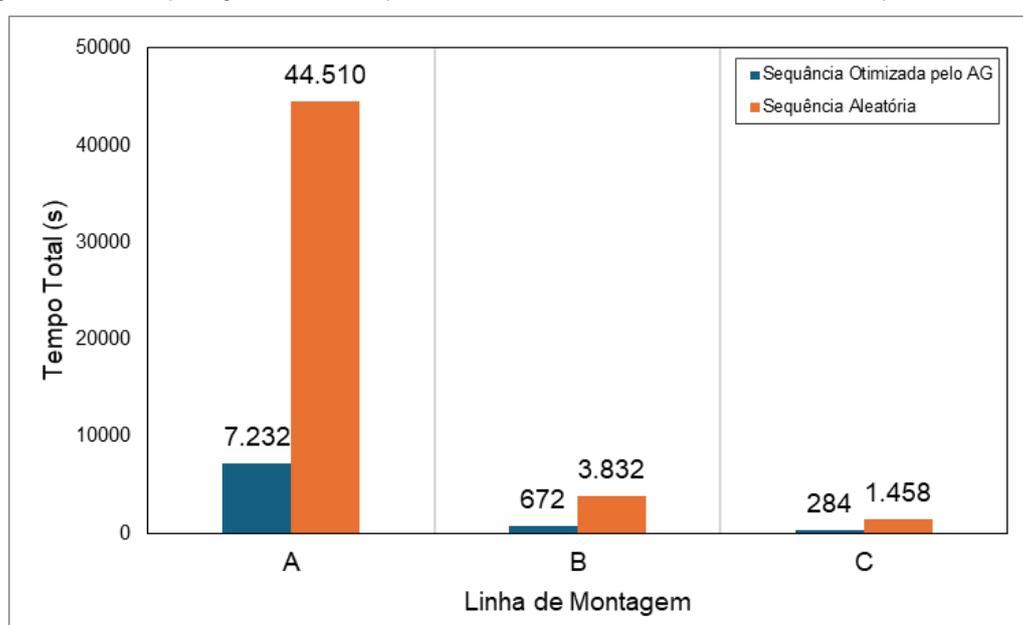
Linha de Montagem	Sequência Otimizada pelo AG	Sequência Aleatória
A	M1→M2→M3→M4→M9→M8→M6→M5→M7→M10→M11→M12→M13→M14→M15→M16→M17→M26→M24→M27→M29→M30→M31→M32→M33→M18→M19→M21→M23→M22→M20→M25→M28	M17→M5→M23→M26→M28→M19→M24→M18→M30→M15→M4→M2→M12→M11→M22→M6→M7→M10→M13→M3→M20→M21→M33→M8→M14→M1→M16→M31→M29→M27→M32→M25
B	M55→M53→M56→M54→M50→M49→M45→M52→M51→M43→M44→M48→M47→M46→M41→M37→M42→M36→M34→M38→M39→M40→M35	M41→M45→M37→M39→M52→M42→M40→M38→M34→M48→M43→M46→M56→M55→M54→M51→M47→M36→M53→M35→M49→M50→M44
C	M62→M59→M66→M57→M68→M58→M60→M72→M67→M65→M61→M64→M63→M71→M70→M69	M58→M67→M70→M66→M62→M60→M57→M63→M72→M68→M65→M59→M61→M69→M71→M64

Fonte: Elaborado pelo autor (2025)

Observando-se a Figura 16, nota-se que a aplicação do AG proporcionou uma redução expressiva no tempo total entre as trocas de modelos de motocicletas, em todas as linhas de montagem analisadas. Na Linha de Montagem A, o tempo reduziu

de 44.510 segundos (sequência aleatória) para 7.232 segundos com o uso do AG, representando uma redução de tempo superior a 83%. Na Linha de Montagem B, o tempo foi reduzido de 3.832 segundos para 672 segundos, uma melhoria de aproximadamente 82%. Por fim, na Linha de Montagem C, a redução foi de 1.458 segundos para 284 segundos, o que equivale a cerca de 80% de redução do tempo.

Figura 16 - Comparação entre tempo total de trocas de modelos: AG vs. Sequência Aleatória



Fonte: Elaborado pelo autor (2025)

Esses resultados demonstram que o AG é significativamente mais eficiente, pois consegue identificar sequências produtivas que minimizam os tempos de troca entre os modelos de motocicletas, otimizando o fluxo da produção. Em contraste, a sequência aleatória não considera critérios de otimização, gerando tempos improdutivos elevados e menor eficiência operacional.

Dessa forma, os dados apresentados confirmam a eficácia do Algoritmo Genético como uma ferramenta robusta de apoio à decisão no sequenciamento de produção, especialmente em ambientes industriais complexos que envolvem múltiplas variáveis e restrições operacionais.

5.5 ANÁLISE DE SENSIBILIDADE DOS PARÂMETROS DO AG

Com o objetivo de avaliar a eficácia dos parâmetros escolhidos para o AG utilizado neste estudo, foram realizados testes de sensibilidade, que consistem em alterar os valores desses parâmetros e observar o impacto nos resultados obtidos. Essa abordagem permite verificar se as configurações iniciais representam as

melhores escolhas ou se há combinações mais adequadas para melhorar o desempenho do AG.

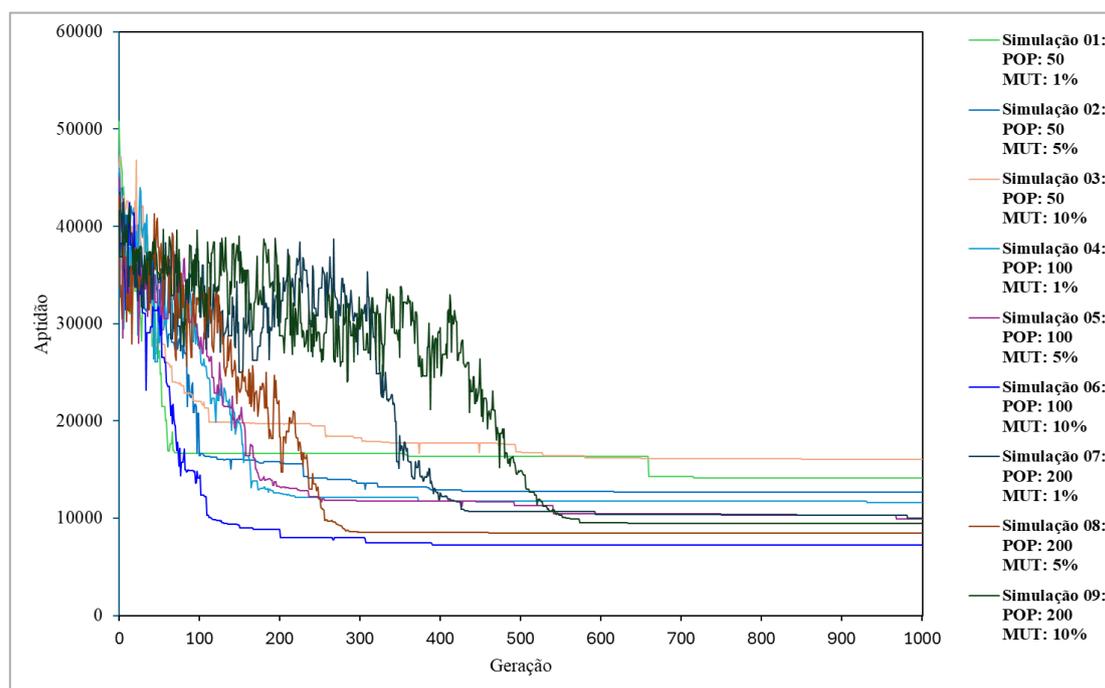
Entre os parâmetros que exercem maior influência nos resultados do AG, destacam-se o tamanho da população e a taxa de mutação. O tamanho da população afeta diretamente a exploração do espaço de busca, pois um número maior de indivíduos possibilita uma cobertura mais ampla de busca das possíveis soluções. Por outro lado, a taxa de mutação está associada à variabilidade genética, sendo essencial para evitar a estagnação em ótimos locais e para manter a diversidade no conjunto de soluções ao longo das gerações. Ambos os parâmetros, portanto, desempenham papéis complementares no equilíbrio entre exploração e exploração local.

Para investigar os efeitos dessas variáveis, foram realizados experimentos simulando diferentes combinações de tamanho da população e taxa de mutação. Foram definidos três tamanhos populacionais distintos: 50, 100 e 200 indivíduos. Adicionalmente, foram consideradas três taxas de mutação: 1%, 5% e 10%. A escolha desses valores foi motivada pela intenção de explorar cenários com baixa, moderada e alta variabilidade genética, bem como populações pequenas, médias e grandes, cobrindo uma ampla gama de possibilidades práticas.

Como o problema estudado consiste na aplicação do AG em três linhas de montagem com tamanhos de instâncias diferentes, para realizar essas simulações, optou-se por utilizar o problema mais desafiador, correspondente à linha de montagem 1, que possui 33 modelos. Essa escolha garante que os resultados obtidos sejam representativos e consistentes, mesmo em cenários com maior complexidade.

Dessa maneira, ao considerar as três opções de tamanho populacional e as três taxas de mutação, obteve-se um total de nove combinações distintas de parâmetros, resultando em nove simulações. Cada simulação foi executada com o mesmo número de gerações e com um critério de parada pré-definido de mil gerações, garantindo uma comparação justa entre os diferentes cenários. Os valores otimizados, representados pelos valores mínimos alcançados em cada geração para cada combinação de parâmetros, estão apresentados na Figura 17.

Figura 17 - Comparativo do AG para diferentes parâmetros de tamanho populacional (POP) e taxa de mutação (MUT)



Fonte: Elaborado pelo autor (2025)

A análise dos resultados das simulações mostrou que a configuração mais eficiente para o Algoritmo Genético foi a combinação de uma população de 100 indivíduos e uma taxa de mutação de 10%. Essa configuração se destacou por equilibrar o tamanho da população, que garante uma exploração adequada do espaço de busca e uma convergência eficaz, com uma taxa de mutação suficientemente alta para evitar estagnação e promover diversidade ao longo das gerações. Esse equilíbrio permitiu ao algoritmo alcançar soluções otimizadas de forma mais eficiente, oferecendo um desempenho superior em comparação com as outras combinações testadas.

Ao comparar os diferentes tamanhos populacionais, verificou-se que uma população de 50 indivíduos levou a uma convergência mais rápida, mas com maior risco de estagnação em soluções subótimas, devido à baixa diversidade. Por outro lado, uma população de 200 indivíduos permitiu uma exploração mais ampla do espaço de busca, porém tornou o processo mais lento, dificultando a rápida identificação de soluções otimizadas. A população intermediária de 100 indivíduos mostrou-se a mais eficaz ao manter a diversidade necessária sem comprometer a eficiência computacional.

A taxa de mutação também desempenhou um papel determinante. Com valores mais baixos, como 1%, o algoritmo apresentou uma convergência estável, mas em alguns casos limitou sua capacidade de escapar de ótimos locais. Já a mutação de 10% garantiu um nível maior de variabilidade ao longo das gerações, permitindo que o algoritmo continuasse explorando novas soluções e evitasse estagnação. Embora taxas muito altas possam prejudicar a convergência, no contexto do problema estudado, que envolve elevada quantidade de instâncias, essa variabilidade se mostrou benéfica ao contribuir para a melhoria progressiva das soluções.

Dado o desempenho satisfatório dessa configuração do AG para o problema com 33 modelos, a mesma abordagem pode ser estendida para instâncias menores, contendo 16 e 23 modelos. Embora essas instâncias tenham menos elementos, elas ainda apresentam desafios de médio e grande porte, exigindo um equilíbrio entre diversidade e convergência. A configuração de 100 indivíduos mantém variedade genética suficiente para evitar convergência prematura, enquanto a taxa de mutação em 10% preserva a capacidade do algoritmo de continuar refinando as soluções ao longo das gerações, garantindo sua eficácia em diferentes tamanhos de problema.

Dessa forma, a configuração de 100 indivíduos e mutação de 10% mostrou-se a mais equilibrada entre eficiência computacional e qualidade das soluções, tornando-se a escolha mais apropriada para este estudo.

5.6 DISCUSSÃO DOS RESULTADOS OBTIDOS

A implementação do algoritmo genético nas Linhas de Montagem A, B e C revelou padrões distintos de comportamento para cada linha, refletindo a complexidade inerente a cada uma delas. Essas diferenças estão diretamente relacionadas ao número de modelos produzidos em cada linha, o que influencia a dificuldade do problema de otimização. Apesar das especificidades de cada linha de montagem, os gráficos demonstram tendências consistentes, evidenciando a eficácia do AG na busca por soluções eficientes.

Em todas as três linhas, o AG apresentou uma forte tendência à convergência, atingindo soluções ótimas ou subótimas em um número finito de gerações. Esse comportamento confirma a capacidade do algoritmo de refinar iterativamente suas soluções ao longo do tempo, ajustando-se às demandas específicas de cada linha de montagem. No entanto, a velocidade dessa convergência variou conforme a complexidade do problema enfrentado em cada linha de montagem.

Por exemplo, a Linha de Montagem C obteve convergência mais rapidamente do que as Linhas de Montagem A e B, o que pode ser explicado pelo menor número de modelos produzidos nessa linha. Com um espaço de busca menor a ser explorado, o AG conseguiu encontrar soluções viáveis em um número reduzido de gerações. Em contrapartida, as Linhas de Montagem A e B, que possuem um maior número de modelos apresentam um espaço de busca mais amplo e, conseqüentemente, um desafio de otimização mais complexo, o que resultou em uma convergência mais gradual, exigindo mais gerações para alcançar um estado de estabilidade.

Os gráficos analisados indicam que, independentemente da linha de montagem, a curva correspondente ao menor tempo de troca (curva azul) apresenta uma redução acentuada logo nas primeiras gerações. Esse comportamento sugere que o algoritmo rapidamente identifica caminhos promissores, ajustando a sequência de produção de forma eficiente. Além disso, as variações nas curvas de aptidão média e máxima (curvas verde e laranja) indicam uma contínua exploração do espaço de soluções. Esse fenômeno é essencial para evitar mínimos locais e garantir que o AG explore novas possibilidades antes da convergência final.

Outro aspecto relevante observado foi a manutenção da diversidade genética ao longo das gerações. Apesar das flutuações nas aptidões máximas, a variabilidade genética permaneceu alta o suficiente para evitar a estagnação prematura para as três linhas de montagem analisadas. Esse fator é fundamental para garantir que o AG não fique preso em ótimos locais, mas continue explorando soluções mais vantajosas.

A obtenção de bons resultados foi diretamente influenciada pela configuração dos parâmetros do algoritmo. Os métodos de seleção e cruzamento foram aplicados de forma estratégica, permitindo uma recomposição eficiente da população e a introdução de novas características genéticas, o que ampliou a exploração do espaço de busca.

Além disso, a taxa e o tipo de mutação adotados tiveram um papel fundamental no equilíbrio entre a introdução de novas variabilidades e a preservação das soluções mais promissoras. A mutação permitiu a incorporação contínua de novas possibilidades, sem comprometer os atributos positivos já adquiridos ao longo das gerações. Esse fator foi essencial para aumentar a robustez das soluções encontradas, minimizando o risco de convergência prematura para soluções subótimas.

A definição do critério de parada com base no número de gerações permitiu que o algoritmo continuasse sua evolução até que uma estabilização significativa das soluções fosse atingida. Isso garantiu que as soluções não fossem interrompidas precocemente, permitindo um maior refinamento das sequências encontradas.

Por fim, o sequenciamento de produção otimizado obtido para as três linhas de montagem demonstrou que o AG foi altamente eficaz em reduzir o tempo entre trocas de modelos. As sequências identificadas não apenas validam as expectativas teóricas do estudo, mas também oferecem insights operacionais valiosos. A implementação dessas sequências pode trazer melhorias significativas para a eficiência produtiva, reduzindo desperdícios de tempo e maximizando o rendimento das linhas de montagem.

6. CONCLUSÃO

Este estudo apresentou uma proposta para a utilização de algoritmos genéticos na otimização do sequenciamento de produção, destacando sua capacidade de minimizar os tempos de troca entre modelos de motocicletas em linhas de montagem. A abordagem adotada permitiu explorar a influência dos principais parâmetros do algoritmo, como os operadores genéticos, tamanho da população e o critério de parada, demonstrando que a definição adequada desses elementos é essencial para garantir um desempenho eficiente e consistente.

Ao longo da pesquisa, verificou-se que o algoritmo genético é capaz de encontrar sequenciamentos eficientes, reforçando seu potencial como uma alternativa promissora para problemas de sequenciamento da produção. Embora o estudo tenha se concentrado na formulação teórica do modelo, os resultados obtidos por meio das simulações indicam que a metodologia proposta pode ser aplicada em contextos reais, contribuindo para uma melhor otimização das linhas de montagem e para a redução de tempos operacionais.

Outro ponto relevante observado foi o impacto da configuração dos parâmetros na qualidade das soluções encontradas. Pequenos ajustes nos valores desses parâmetros podem resultar em melhorias significativas, garantindo uma convergência mais rápida e precisa para soluções ótimas ou subótimas. Isso demonstra a importância de um ajuste fino do algoritmo para adaptar-se a diferentes cenários produtivos, tornando o processo mais previsível e otimizado.

A pesquisa também abre caminho para estudos futuros que possam validar a aplicação prática do modelo em ambientes industriais, permitindo a análise de sua viabilidade em condições reais de operação. Além disso, explorar a combinação de algoritmos genéticos com outras técnicas de otimização, como métodos híbridos, pode ampliar ainda mais a eficiência da abordagem. Outras possibilidades incluem a incorporação de novas restrições e variáveis ao modelo, tornando-o mais robusto e adaptável a diferentes necessidades do setor produtivo.

Em síntese, este estudo reforça a relevância dos algoritmos genéticos como uma ferramenta eficaz para a otimização do sequenciamento de produção, contribuindo para o avanço de soluções inteligentes no campo da engenharia de produção. Apesar do caráter teórico da pesquisa, os resultados obtidos demonstram que essa abordagem pode trazer benefícios expressivos para a indústria, sendo uma alternativa viável para a melhoria da eficiência operacional e da competitividade no setor. A continuidade dessas investigações poderá consolidar ainda mais o uso de técnicas de computação evolutiva como um diferencial estratégico para processos produtivos cada vez mais dinâmicos e otimizados.

7. CONTRIBUIÇÕES DA PESQUISA

7.1 IMPACTOS ACADÊMICOS

A presente pesquisa representa uma contribuição significativa para o avanço do conhecimento no campo dos algoritmos genéticos aplicados ao Problema do Caixeiro Viajante, com foco específico no problema do sequenciamento de produção no setor industrial de duas rodas. Ao integrar conceitos de otimização combinatória com desafios reais da manufatura, este estudo amplia a compreensão sobre a viabilidade e eficiência dos algoritmos genéticos em problemas complexos, fornecendo um referencial robusto para futuras investigações.

Além disso, a pesquisa promove o desenvolvimento de novas abordagens metodológicas para o problema do sequenciamento de produção. A utilização do AG conferiu uma perspectiva inovadora à otimização de processos industriais, ressaltando o papel estratégico dos métodos computacionais avançados no contexto da Engenharia de Produção.

Os resultados obtidos incentivam novas pesquisas acadêmicas, fomentando a produção de dissertações, teses e artigos científicos que aprofundem o tema e

explorem novas aplicações da computação evolutiva em ambientes produtivos. Dessa forma, este estudo não apenas contribui para a literatura acadêmica, mas também reforça a relevância dos algoritmos genéticos como ferramentas eficazes para a solução de problemas práticos da indústria.

7.2 IMPACTOS ECONÔMICOS

Do ponto de vista econômico, esta pesquisa apresenta um potencial relevante para a redução de custos operacionais e a melhoria da eficiência produtiva no setor industrial de duas rodas. A proposta de aplicação de algoritmos genéticos no sequenciamento de produção busca identificar soluções que minimizem os tempos de troca entre modelos de motocicletas, contribuindo para uma operação mais eficiente e com menor desperdício de recursos.

A otimização do sequenciamento tem o potencial de gerar economia de tempo e, conseqüentemente, de recursos financeiros, possibilitando uma maior produtividade das linhas de montagem. Esses ganhos podem se refletir na competitividade das empresas do setor, especialmente no contexto do Polo Industrial de Manaus, onde a eficiência produtiva é um fator estratégico.

Além disso, a metodologia proposta pode ser considerada para estudos futuros ou adaptações em outros contextos industriais que enfrentam desafios semelhantes no sequenciamento da produção.

7.3 IMPACTOS SOCIAIS

Os impactos sociais desta pesquisa estão diretamente ligados à melhoria da organização e da estabilidade do setor industrial. A otimização do sequenciamento de produção contribui para um ambiente de trabalho mais estruturado, reduzindo a variabilidade nas operações e permitindo um planejamento mais eficiente das atividades. Isso favorece melhores condições de trabalho, tornando os processos mais previsíveis e reduzindo a pressão sobre os trabalhadores decorrente de falhas no fluxo produtivo.

Além disso, a maior eficiência operacional pode fortalecer a competitividade da indústria, fator essencial para a manutenção da atividade produtiva e, conseqüentemente, para a preservação dos empregos na região. Um setor industrial mais estável gera impactos positivos na economia local, proporcionando maior segurança para os trabalhadores e para a comunidade ao redor.

No contexto social mais amplo, o fortalecimento da indústria impulsiona o desenvolvimento da região ao consolidar um ambiente produtivo mais robusto e menos suscetível a oscilações. Isso contribui para a continuidade das operações industriais e para a geração de oportunidades econômicas, beneficiando tanto os trabalhadores quanto a população que depende indiretamente da atividade industrial.

Dessa forma, a pesquisa não apenas busca aprimorar processos produtivos, mas também promove efeitos positivos na organização do trabalho, na estabilidade do setor e na segurança socioeconômica da região.

REFERÊNCIAS

ABDULKARIM, H. A.; ALSHAMMARI, I. F. Comparison of Algorithms for Solving Traveling Salesman Problem. **International Journal of Engineering and Advanced Technology**, v. 4, n. 6, p. 76–79, 2015.

ADAN, J. A hybrid genetic algorithm for parallel machine scheduling with setup times: A comparative study of metaheuristics on large problem instances. **Journal of Intelligent Manufacturing**, v. 33, n. 7, p. 2059–2073, 2022. Springer US.

AHMED, Z. H.; CHOUDHARY, M. T.; AL-DAYEL, I. Effects of crossover operator combined with mutation operator in genetic algorithms for the generalized travelling salesman problem. **International Journal of Industrial Engineering Computations**, v. 15, n. 3, p. 627–644, 2024.

AHMED, Z. H.; HARON, H.; AL-TAMEEM, A. Appropriate Combination of Crossover Operator and Mutation Operator in Genetic Algorithms for the Travelling Salesman Problem. **Computers, Materials and Continua**, v. 79, n. 2, p. 2399–2425, 2024.

AIDY, A.; RADY, M.; MASHHOUR, I. M.; MAHFOUZ, S. Y. Structural Design Optimization of Flat Slab Hospital Buildings Using Genetic Algorithms. **Buildings**, v. 12, n. 12, 2022.

AL-FURHUD, M. A.; AHMED, Z. H. Experimental Study of a Hybrid Genetic Algorithm for the Multiple Travelling Salesman Problem. (G. S. Dulikravich, Org.) **Mathematical Problems in Engineering**, v. 2020, p. 1–13, 2020.

AL-ROMEMA, M.; MAKAREM, M. A. Enhancing Evolutionary Solver Efficiency for NP-Hard Single Machine Scheduling Problems. **International Journal of Science and Research (IJSR)**, v. 12, n. 11, p. 2015–2025, 2023.

ALAM, T.; QAMAR, S.; DIXIT, A.; BENAIDA, M. Genetic Algorithm: Reviews, Implementations, and Applications. **International Journal of Engineering Pedagogy (iJEP)**, v. 10, n. 6, p. 57, 2020.

ALHIJAWI, B.; AWAJAN, A. Genetic algorithms: theory, genetic operators, solutions, and applications. **Evolutionary Intelligence**, v. 17, n. 3, p. 1245–1256, 2024.

ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. **Pesquisa Operacional**. 2º ed. Rio de Janeiro: Elsevier: ABEPRO, 2015.

AVGERINOS, E.; MANIKAROS, N.; VLACHOU, R. Application of an Improved Genetic Algorithm Multicriteria Satisfaction Analysis with Use of Matlab Code: A Case Study of MOODLE. **Proceedings of World Multi-Conference on Systemics, Cybernetics and Informatics, WMSCI**, v. 2023-Septe, n. 2, p. 129–136, 2023.

BAO, X.; WANG, G.; XU, L.; WANG, Z. Solving the Min-Max Clustered Traveling Salesmen Problem Based on Genetic Algorithm. **Biomimetics**, v. 8, n. 2, p. 238, 2023.

BARI, P.; KARANDE, P.; MENEZES, J. Simulation of Job Sequencing for Stochastic Scheduling With a Genetic Algorithm. **Operational Research in Engineering Sciences: Theory and Applications**, v. 5, n. 3, p. 17–39, 2022.

BASHIR, A.; SRIVASTAVA, M. K. Implementation of Genetic Algorithm Using the Traveling Salesman Problem in Cloud. **African Journal of Science and Research**, v. 1, n. 1, p. 1–7, 2021.

BAUMANN, R. Globalização, desglobalização e o Brasil. **Brazilian Journal of Political Economy**, v. 42, n. 3, p. 592–618, 2022.

BOJIC, S.; MASLARIC, M.; MIRCETIC, D.; NIKOLICIC, S.; TODOROVIC, V. Simulation and Genetic Algorithm-based approach for multi-objective optimization of production planning: A case study in industry. **Advances in Production Engineering And Management**, v. 18, n. 2, p. 250–262, 2023.

BORU, M.; BUTAR, B.; MEDIKANO, A.; ZULKARNAIN, Y. Determining Book Distribution Routes by Implementing Vehicle Routing Problems Using Excel Solver at Gunadarma University. , v. 1, n. 4, p. 7–15, 2024.

BRASIL. Ministério do Desenvolvimento, Indústria, Comércio e Serviços. Indicadores do Polo Industrial de Manaus. Disponível em: <<https://www.gov.br/suframa/pt-br/publicacoes/indicadores/IndicaOUT24/view>>. Acesso em: 8/1/2025.

CAHYANI, C. A. M.; WIRADINATA, T. Traveling Salesman Problem Multi-destination Route Recommendation System Using Genetic Algorithm and Google Maps API.

International Journal of Computer Applications, v. 185, n. 18, p. 35–43, 2023.

CARRERES-PRIETO, D.; GARCÍA, J. T.; CERDÁN-CARTAGENA, F.; SUARDIAZ-MURO, J.; LARDÍN, C. Implementing Early Warning Systems in WWTP. An investigation with cost-effective LED-VIS spectroscopy-based genetic algorithms. **Chemosphere**, v. 293, 2022.

CAUCHICK-MIGUEL, P. A.; FLEURY, A.; MELLO, C. H. P.; et al. **Metodologia de Pesquisa em Engenharia de Produção e Gestão de Operações**. 3º ed. Rio de Janeiro: LTC, 2018.

CAVANI, S.; IORI, M.; ROBERTI, R. Exact methods for the traveling salesman problem with multiple drones. **Transportation Research Part C: Emerging Technologies**, v. 130, p. 103280, 2021. Elsevier.

CHANDRA, A.; NATALIA, C.; NARO, A. Comparative Solutions of Exact and Approximate Methods for Traveling Salesman Problem. **Lámpsakos**, , n. 25, p. 1–12, 2021.

CHAUDHRY, I. A.; RAFIQUE, A. F.; ELBADAWI, I. A. Q.; et al. Integrated scheduling of machines and automated guided vehicles (AGVs) in flexible job shop environment using genetic algorithms. **International Journal of Industrial Engineering Computations**, v. 13, n. 3, p. 343–362, 2022.

CORRÊA, H. L.; CORRÊA, C. A. **Administração de Produção e Operações: manufatura e serviços: uma abordagem estratégica**. 5º ed. Barueri: Atlas, 2022.

DAMIA, A.; ESNAASHARI, M.; PARVIZIMOSAED, M. Adaptive Genetic Algorithm Based on Mutation and Crossover and Selection Probabilities. 7th International Conference on Web Research (ICWR). **Anais...** . p.86–90, 2021. IEEE.

DARWIN, C. **A Origem das Espécies**. Leça da Palmeira: Planeta Vivo, 2009.

DEAP. DEAP: Distributed Evolutionary Algorithms in Python. Disponível em: <<https://github.com/DEAP/deap>>. Acesso em: 24/1/2025.

DIABY, M.; KARWAN, M. H. **Advances in Combinatorial Optimization**. Toh Tuck L: World Scientific, 2016.

DIDDEN, J. B. H. C.; LEFEBER, E.; ADAN, I. J. B. F.; PANHUIJZEN, I. W. F. Genetic algorithm and decision support for assembly line balancing in the automotive industry. **International Journal of Production Research**, v. 61, n. 10, p. 3377–3395, 2023. Taylor & Francis.

DOCKHORN, A.; LUCAS, S. Choosing Representation, Mutation, and Crossover in Genetic Algorithms. **IEEE Computational Intelligence Magazine**, v. 17, n. 4, p. 52–53, 2022. IEEE.

DUTOT, A.; OLIVIER, D. Swarm Problem-Solving. **Agent-based Spatial Simulation with NetLogo, Volume 2**. p.117–172, 2017. Elsevier.

FAN, J.; ZHANG, C.; LIU, Q.; SHEN, W.; GAO, L. An improved genetic algorithm for flexible job shop scheduling problem considering reconfigurable machine tools with limited auxiliary modules. **Journal of Manufacturing Systems**, v. 62, n. November 2021, p. 650–667, 2022. Elsevier Ltd.

FRASER, E. A.; OKONKWO, O. R. Genetic Algorithm Approach for Optimal Cyclic Tour Round the State Capitals in Nigeria'S Niger Delta Region. **International Journal of Research -GRANTHAALAYAH**, v. 9, n. 5, p. 171–186, 2021.

FREITAS, M. G. DE; FUCHIGAMI, H. Y. A new technology implementation via mathematical modeling for the sequence-dependent setup times of industrial problems. **Computers and Industrial Engineering**, v. 172, n. PB, p. 108624, 2022. Elsevier Ltd.

GAO, Y.; WANG, Z.; GAO, L.; LI, X. A Matheuristic Approach for the No-Wait Flowshop Scheduling Problem with Makespan Criterion. 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD). **Anais...** . p.1028–1033, 2022. IEEE.

GOMES, D. E.; IGLÉSIAS, M. I. D.; PROENÇA, A. P.; LIMA, T. M.; GASPAR, P. D. Applying a Genetic Algorithm to a m-TSP: Case Study of a Decision Support System for Optimizing a Beverage Logistics Vehicles Routing Problem. **Electronics**, v. 10, n. 18, p. 2298, 2021.

GOMES, U. T.; PINHEIRO, P. R.; SARAIVA, R. D. Dye schedule optimization: A case

study in a textile industry. **Applied Sciences (Switzerland)**, v. 11, n. 14, p. 1–18, 2021.

GÖZÜAÇIK, E.; ALTIOK, M.; GÖKREM, L. Indoor Navigation with Visible Light Communication using Genetic Algorithm. **European Journal of Science and Technology**, , n. 26, p. 185–190, 2021.

HAMIDA, S. BEN; BENJELLOUN, G. Extending DEAP with active sampling for evolutionary supervised learning. **Proceedings of the 16th International Conference on Software Technologies, ICSOFT 2021**, , n. Icssoft, p. 574–582, 2021.

HAN, S.; XIAO, L. An improved adaptive genetic algorithm. (S. Zhao, Org.) **SHS Web of Conferences**, v. 140, p. 01044, 2022.

HARIYADI; MUTIRA, P.; NGUYEN, P. T.; ISWANTO, I.; SUDRAJAT, D. Traveling Salesman Problem Solution using Genetic Algorithm. **Journal of Critical Reviews**, v. 7, n. 1, p. 56–61, 2020.

HE, X.; PAN, Q.-K.; GAO, L.; NEUFELD, J. S. An asymmetric traveling salesman problem based matheuristic algorithm for flowshop group scheduling problem. **European Journal of Operational Research**, v. 310, n. 2, p. 597–610, 2023. Elsevier B.V.

HEURISTIC AND EVOLUTIONARY ALGORITHMS LABORATORY (HEAL). HeuristicLab documentation. Disponível em: <<https://dev.heuristiclab.com/trac.fcgi/wiki/Documentation>>. Acesso em: 24/1/2025.

HILLIER, F. S.; LIEBERMAN, G. J. **Introdução à Pesquisa Operacional**. 9º ed. Porto Alegre: AMGH, 2013.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence**. London: MIT, 1992.

HOUNDJI, V. R.; GNA, T. Genetic Algorithms for Solving the Pigment Sequencing Problem. **Revue Africaine de Recherche en Informatique et Mathématiques Appliquées**, v. Volume 40, p. 1–15, 2024.

JOSHI, D. Genetic Algorithm and its Applications - A Brief Study. **Asian Journal of**

Convergence in Technology, v. 7, n. 3, p. 8–12, 2021.

KATOCH, S.; CHAUHAN, S. S.; KUMAR, V. A review on genetic algorithm: past, present, and future. **Multimedia Tools and Applications**, v. 80, n. 5, p. 8091–8126, 2020. Multimedia Tools and Applications.

KAUR, P. Study of the Various Selection Techniques in Genetic Algorithms. **Interantional Journal of Scientific Research in Engineering and Management**, v. 07, n. 03, p. 1–4, 2023.

KHDEIR, Y.; AWAD, A. A Comparison of Heuristic Algorithms for Solving the Traveling Salesman Problem. **An-Najah University Journal for Research - A (Natural Sciences)**, v. 39, n. 1, p. 1–8, 2025.

KINAST, A.; DOERNER, K. F.; RINDERLE-MA, S. Biased random-key genetic algorithm for cobot assignment in an assembly/disassembly job shop scheduling problem. **Procedia Computer Science**, v. 180, n. 2019, p. 328–337, 2021. Elsevier B.V.

KORTE, B.; VYGEN, J. **Combinatorial Optimization: Theory and Algorithms**. 6^o ed. Berlin: Springer, 2018.

KOWALSKI, M.; KUGI, A.; STEINBOECK, A. Job Scheduling for a Multi-Line Steel Hot Rolling Mill With Selectable Furnaces. **IFAC-PapersOnLine**, v. 56, n. 2, p. 5376–5381, 2023. Elsevier Ltd.

KULENOVIĆ, F.; HOŠIĆ, A. Application of genetic algorithm in solving the travelling Salesman problem. **IOP Conference Series: Materials Science and Engineering**, v. 1208, n. 1, p. 012032, 2021.

KUROSWISKI, A. R.; PIRES, H. B.; PASSARO, A.; GUIMARÃES, L. N. F.; SENNE, E. L. F. Abordagens Exata e Heurística na Otimização do Problema do Caixeiro Viajante com Drone. **Sige**, p. 1–6, 2021.

LIANG, X.; LIU, Y.; GU, X.; HUANG, M.; GUO, F. Adaptive Genetic Algorithm Based on Individual Similarity to Solve Multi-Objective Flexible Job-Shop Scheduling Problem. **IEEE Access**, v. 10, p. 45748–45758, 2022. IEEE.

MANYAM, S. G.; SUNDAR, K.; CASBEER, D. W. Cooperative Routing for an Air–Ground Vehicle Team—Exact Algorithm, Transformation Method, and Heuristics. **IEEE Transactions on Automation Science and Engineering**, v. 17, n. 1, p. 537–547, 2020. IEEE.

MAROOOF, A.; AYVAZ, B.; NAEEM, K. Logistics Optimization Using Hybrid Genetic Algorithm (HGA): A Solution to the Vehicle Routing Problem With Time Windows (VRPTW). **IEEE Access**, v. 12, n. February, p. 36974–36989, 2024. IEEE.

MATHWORKS. MATLAB. Disponível em: <<https://www.mathworks.com/products/matlab.html>>. Acesso em: 24/1/2025.

MICROSOFT. Define and solve a problem by using Solver. Disponível em: <<https://support.microsoft.com/en-us/office/define-and-solve-a-problem-by-using-solver-5d1a388f-079d-43ac-a7eb-f63e45925040>>. Acesso em: 24/1/2025.

MOSAYEBI, M.; SODHI, M. Tuning genetic algorithm parameters using design of experiments. Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion. **Anais...** . p.1937–1944, 2020. New York, NY, USA: ACM.

MURTHY, A. S.; MANKAME, D. D. P. Genetic Algorithms - A Brief Study. **International Journal of Science and Research (IJSR)**, v. 13, n. 7, p. 1195–1200, 2024.

MUSTAFA, K. Applying Developed Genetic Algorithm Operators to the Knapsack Problems. **Global Journal of Computer Sciences: Theory and Research**, v. 13, n. 2, p. 75–91, 2023.

MZILI, T.; MZILI, I.; RIFFI, M. E. Artificial rat optimization with decision-making: A bio-inspired metaheuristic algorithm for solving the traveling salesman problem. **Decision Making: Applications in Management and Engineering**, v. 6, n. 2, p. 150–176, 2023.

NIKFARJAM, A.; NEUMANN, A.; NEUMANN, F. On the Use of Quality Diversity Algorithms for The Traveling Thief Problem. **Proceedings of the Genetic and Evolutionary Computation Conference**, p. 260–268, 2021. New York: ACM.

OBI, C. J. Using genetic algorithm to solve multiple traveling salesman problem and

considering Carbon emissions. **Indian Journal of Science and Technology**, v. 13, n. 36, p. 3707–3715, 2020.

OCHELSKA-MIERZEJEWSKA, J.; PONISZEWSKA-MARAÑDA, A.; MARAÑDA, W. Selected Genetic Algorithms for Vehicle Routing Problem Solving. **Electronics**, v. 10, n. 24, p. 3147, 2021.

OSABA, E.; YANG, X.-S.; DEL SER, J. Traveling salesman problem: a perspective review of recent research and new results with bio-inspired metaheuristics. **Nature-Inspired Computation and Swarm Intelligence**. p.135–164, 2020. Elsevier.

PALISADE. About Evolver. Disponível em: <<https://help.palisade.com/v8/en/Evolver/About.htm>>. Acesso em: 24/1/2025.

PEDROSA, J.; PUIG, V. Job Shop Scheduling with Limited-Capacity Buffers using Constraint Programming and Genetic Algorithms. **IFAC-PapersOnLine**, v. 56, n. 2, p. 953–958, 2023.

PISTORE, G. **Uso de Algoritmos Genéticos para o aumento da assertividade no reconhecimento e identificação de promotores no BACPP**, 2017. TCC, Caxias do Sul: UCS. Disponível em: <<https://repositorio.ucs.br/xmlui/handle/11338/3721>>. Acesso em: 22/10/2024.

PLACIDO, A. DI; ARCHETTI, C.; CERRONE, C. A genetic algorithm for the close-enough traveling salesman problem with application to solar panels diagnostic reconnaissance. **Computers & Operations Research**, v. 145, n. July 2021, p. 105831, 2022. Elsevier Ltd.

RAJU, R. K.; SIVAKUMAR, S.; WANG, X.; ULISSI, Z. W. Cluster-MLP: An Active Learning Genetic Algorithm Framework for Accelerated Discovery of Global Minimum Configurations of Pure and Alloyed Nanoclusters. **Journal of Chemical Information and Modeling**, v. 63, n. 20, p. 6192–6197, 2023.

RAMSEY, C.; GALLEMORE, S.; BOWLING, J. Optimization of target grouping in distributive stereotactic radiosurgery using the excel evolutionary solver. **Journal of Applied Clinical Medical Physics**, , n. November, p. 1–7, 2024.

RAO, B. M.; XANTHOPOULOS, P.; ZHENG, Q. P. DeLand Crayon Company: An Application of the Traveling Salesman Problem to Production Scheduling with Sequence-Dependent Setup Times. **INFORMS Transactions on Education**, v. 20, n. 2, p. 93–98, 2020.

RESENDE, M. G. C.; RIBEIRO, C. C.; GLOVER, F.; MARTÍ, R. Scatter Search and Path-Relinking: Fundamentals, Advances, and Applications. **Handbook of Metaheuristics**. 3º ed, p.87–107, 2018. Montreal: Springer.

ROSA, P. H. P.; SRIWINDONO, H.; NUGROHO, R. A.; POLINA, A. M.; PINARYANTO, K. Comparison of Crossover and Mutation Operators to Solve Teachers Placement Problem by Using Genetic Algorithm. **Journal of Physics: Conference Series**, v. 1566, n. 1, p. 012021, 2020.

SADEGHI, P.; REBELO, R. D.; FERREIRA, J. S. Using variable neighbourhood descent and genetic algorithms for sequencing mixed-model assembly systems in the footwear industry. **Operations Research Perspectives**, v. 8, p. 100193, 2021. Elsevier Ltd.

SATHYA, N.; MUTHUKUMARAVEL, A. A Review of the Optimization Algorithms on Traveling Salesman Problem. **Indian Journal of Science and Technology**, v. 8, n. 29, p. 1–4, 2015. Disponível em: <<https://indjst.org/articles/a-review-of-the-optimization-algorithms-on-traveling-salesman-problem>>. .

SHAHAB, M. L. New heuristic algorithm for traveling salesman problem. **Journal of Physics: Conference Series**, v. 1218, n. 1, p. 02–10, 2019.

SIKORA, T.; GRYGLEWICZ-KACERKA, W. Application of Genetic Algorithms to the Traveling Salesman Problem. **Applied Computer Science**, v. 19, n. 2, p. 55–62, 2023.

SILVA, F. A. M. DA. **Estudo de formulações do problema do caixeiro viajante e sequenciamento de tarefas e suas aplicações em um problema prático de produção na indústria de pães e bolos**, 2019. Tese de doutorado, Campinas: UNICAMP. Disponível em: <<https://repositorio.unicamp.br/jspui/handle/REPOSIP/335600>>. Acesso em: 22/10/2024.

SINGH, D.; PRAKASH, R. Understanding Evolutionary Based Genetic Algorithm and Implementation using R. **SSRN Electronic Journal**, p. 1–5, 2021.

SLACK, N.; BRANDON-JONES, A.; BURGESS, N. **Administração da Produção**. 10^o ed. Barueri: Atlas, 2023.

SZABÓ, K. A Brief Overview of Genetic Algorithms. **Design of Machines and Structures**, v. 13, n. 2, p. 113–120, 2023.

TIAN, S.; ZHANG, C.; FAN, J.; LI, X.; GAO, L. A genetic algorithm with critical path-based variable neighborhood search for distributed assembly job shop scheduling problem. **Swarm and Evolutionary Computation**, v. 85, n. January, p. 101485, 2024. Elsevier B.V.

TOAZA, B.; ESZTERGÁR-KISS, D. A review of metaheuristic algorithms for solving TSP-based scheduling optimization problems. **Applied Soft Computing**, v. 148, p. 110908, 2023.

VIANNA, B. L. **Metodos exato e heurístico para resolução do problema do caixeiro viajante em famílias**, 2022. Instituto Tecnológico de Aeronáutica e Universidade Federal de São Paulo. Disponível em: <<https://repositorio.unifesp.br/server/api/core/bitstreams/94ab9953-3cb3-42e5-a5fa-7c0bed0853fb/content>>. Acesso em: 22/10/2024.

WAN, X.; LIU, K.; QIU, W.; KANG, Z. An Assembly Sequence Planning Method Based on Multiple Optimal Solutions Genetic Algorithm. **Mathematics**, v. 12, n. 4, 2024.

WANG, Y.; ZHU, Q. A Hybrid Genetic Algorithm for Flexible Job Shop Scheduling Problem with Sequence-Dependent Setup Times and Job Lag Times. **IEEE Access**, v. 9, p. 104864–104873, 2021. IEEE.

WIRSANSKY, E. **Hands-on genetic algorithms with Python**. Birmingham: Packt, 2020.

YANG, J.; SUN, T.; HUANG, X.; et al. Optimizing painting sequence scheduling based on adaptive partheno-genetic algorithm. **Processes**, v. 9, n. 10, p. 1–17, 2021.

YUAN, X.; ZHU, J.; LI, Y.; HUANG, H.; WU, M. An enhanced genetic algorithm for

unmanned aerial vehicle logistics scheduling. **IET Communications**, v. 15, n. 10, p. 1402–1411, 2021.

ZEROVNIK, J.; HERAKOVIC, N. A new application of the generalized traveling salesman problem in industry 4.0 and 5.0. **Multiple Criteria Decision Making**, v. 16, p. 153–163, 2021.

ZHONG, L.; LI, W.; QIAN, B.; HE, L. Improved discrete cuckoo-search algorithm for mixed no-idle permutation flow shop scheduling with consideration of energy consumption. **IET Collaborative Intelligent Manufacturing**, v. 3, n. 4, p. 345–355, 2021.