



UFOP

Universidade Federal
de Ouro Preto

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas**

**Implementação de meta-heurísticas
para o Problema de
Sequenciamento de Tarefas em
uma Máquina com Penalidades por
Antecipação e Atraso**

Mariana de Sousa Vieira

**TRABALHO DE
CONCLUSÃO DE CURSO**

ORIENTAÇÃO:
Fernando Bernardes de Oliveira

**Março, 2026
João Monlevade–MG**

Mariana de Sousa Vieira

**Implementação de meta-heurísticas para o
Problema de Sequenciamento de Tarefas em
uma Máquina com Penalidades por
Antecipação e Atraso**

Orientador: Fernando Bernardes de Oliveira

Monografia apresentada ao curso de Sistemas de Informação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Universidade Federal de Ouro Preto

João Monlevade

Março de 2026



FOLHA DE APROVAÇÃO

Mariana de Sousa Vieira

Implementação de meta-heurísticas para o Problema de Sequenciamento de Tarefas em uma Máquina com Penalidades por Antecipação e Atraso

Monografia apresentada ao Curso de Sistemas de Informação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação

Aprovada em 05 de março de 2026

Membros da banca

Prof. Dr. Fernando Bernardes de Oliveira - Orientador (Universidade Federal de Ouro Preto)
Profª. Drª. Geiza Cristina da Silva - Avaliadora (Universidade Federal do ABC - UFABC)
Profª. Drª. Tatiana Alves Costa - Avaliadora (Universidade Federal de Ouro Preto)

Fernando Bernardes de Oliveira, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 01/04/2026



Documento assinado eletronicamente por **Fernando Bernardes de Oliveira, PROFESSOR DE MAGISTERIO SUPERIOR**, em 01/04/2026, às 11:24, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1083344** e o código CRC **CB7DD182**.

Agradecimentos

Sou profundamente grata a Deus por me sustentar em todos os momentos e por me permitir alcançar a conclusão deste curso. Este percurso acadêmico, marcado por desafios e superações, evidenciou que cada etapa contribuiu de forma significativa para meu desenvolvimento pessoal e profissional.

Agradeço à **Universidade Federal de Ouro Preto (UFOP)**, pela excelência na formação acadêmica e pelo ambiente propício à aprendizagem, à pesquisa e ao crescimento intelectual. Aos docentes que atuaram em minha trajetória, registro minha sincera gratidão pelo compartilhamento de conhecimento, dedicação e comprometimento com a educação de qualidade. Em especial, agradeço ao meu orientador, **Professor Fernando**, pela orientação precisa, pelas contribuições relevantes e pelo suporte contínuo durante o desenvolvimento deste trabalho.

Aos meus pais, Alex e Marilene, expresse minha profunda gratidão pelo apoio, incentivo e pela base sólida que sempre me proporcionaram. À minha irmã, Maria, pelo estímulo constante e exemplo diário de determinação. Aos meus familiares — tios, tias e primos — pelo acompanhamento e incentivo ao longo desta trajetória. Aos meus padrinhos, pelo cuidado e suporte contínuo. Ao meu namorado, Ronaldo, pelo apoio constante e parceria durante todo o processo. Aos meus amigos, pelo auxílio e estímulo em momentos de desafio.

Dedico também esta conquista à memória dos meus avós, cujos ensinamentos e valores permanecem como referência em minha formação acadêmica e pessoal.

Resumo

Este trabalho apresenta a implementação e análise comparativa de meta-heurísticas aplicadas ao Problema de Sequenciamento de Tarefas em uma Máquina com Penalidades por Antecipação e Atraso (PSUMAA). Foram avaliadas estratégias de Busca Local (Busca Local com *First Improvement* (BL-FI) e Busca Local com *Best Improvement* (BL-BI)), um Algoritmo Genético (Algoritmos Genéticos (AG)) clássico e versões híbridas combinando AG com Busca Local. O objetivo foi investigar a capacidade dessas técnicas em produzir soluções de alta qualidade considerando o custo total de penalidades e princípios *Just-in-Time*. Os experimentos abrangeram múltiplas instâncias do problema, permitindo análise do desempenho em termos de função objetivo, *gap* médio e tempo de execução. Os resultados indicaram que as abordagens híbridas superaram consistentemente os métodos isolados, apresentando menor *gap* médio e maior robustez, embora com maior custo computacional. As estratégias de Busca Local isoladas mostraram eficiência temporal, mas tendência à estagnação em ótimos locais, enquanto o AG clássico apresentou maior variabilidade e escalabilidade limitada. Conclui-se que, considerando o contexto experimental estabelecido, a hibridização entre exploração global e intensificação local constitui estratégia eficaz para o PSUMAA, evidenciando o trade-off entre desempenho computacional e qualidade das soluções.

Palavras-chaves: Sequenciamento de Tarefas, Máquina Única, Penalidades por Antecipação e Atraso, Algoritmos Genéticos, Meta-heurísticas, Busca Local.

Abstract

This work presents the implementation and comparative analysis of meta-heuristics applied to the Single Machine Scheduling Problem with Earliness and Tardiness Penalties (**PSUMAA**). Local Search strategies (**BL-FI** and **BL-BI**), a classical Genetic Algorithm (*Genetic Algorithm* (**GA**)), and hybrid versions combining **GA** with Local Search were evaluated. The objective was to investigate the ability of these techniques to produce high-quality solutions in terms of total penalty cost and Just-in-Time principles. Experiments included multiple problem instances, allowing performance analysis in terms of objective function, average gap, and computational time. Results showed that hybrid approaches consistently outperformed standalone methods, achieving lower average gaps and greater robustness, albeit with higher computational cost. Isolated Local Search strategies were temporally efficient but prone to local optima, while the classical **GA** exhibited higher variability and limited scalability. Considering the established experimental context, it is concluded that hybridization between global exploration and local intensification constitutes an effective strategy for **PSUMAA**, highlighting the trade-off between computational effort and solution quality.

Keywords: Job Scheduling, Single Machine, Earliness and Tardiness Penalties, Genetic Algorithms, Meta-heuristics, Local Search.

Lista de figuras

Figura 1 – Fluxo de informações em um sistema produtivo.	20
Figura 2 – Sistema de produção com máquina única.	21
Figura 3 – Sistema de produção com máquinas paralelas.	21
Figura 4 – Sistema de produção do tipo flow shop.	22
Figura 5 – Sistema de produção do tipo job shop.	23
Figura 6 – Diagrama de Gantt com tempos de setup dependentes da sequência	25
Figura 7 – Exemplo ilustrativo de aprisionamento em mínimo local em busca local	30
Figura 8 – Comparação do tempo médio de execução em função do número de gerações para os métodos avaliados.	64
Figura 9 – Convergência dos métodos em função do número de gerações (Gap Médio \times Gerações).	66
Figura 10 – Distribuição do gap (%) por método.	67
Figura 11 – Distribuição do gap (%) por método considerando todas as gerações.	68
Figura 12 – Convergência do gap médio (%) por instância considerando 100 gerações.	69
Figura 13 – Convergência do gap médio (%) por instância considerando 1000 gerações.	71
Figura 14 – Convergência do gap médio (%) por instância considerando 10000 gerações.	72

Lista de tabelas

Tabela 1 – Características consideradas na formulação do PSUMAA	24
Tabela 2 – Principais operadores dos Algoritmos Genéticos	32
Tabela 3 – Exemplo de dados de entrada para uma instância com 8 tarefas.	42
Tabela 4 – Matriz de tempos de setup S_{ij}	43
Tabela 5 – Instâncias utilizadas nos experimentos.	53
Tabela 6 – Resultados obtidos pelas estratégias de BL-FI e BL-BI.	55
Tabela 7 – Comparação entre o AG e o AG+BL-FI.	57
Tabela 8 – Comparação entre AG (AG) e o (AG+BL-FI), Versão Final. . .	58
Tabela 9 – Comparação entre o AG (AG) e o AG com Busca Local Best Improvement (AG+BL-BI).	59
Tabela 10 – Comparação entre o AG e o AG AG+BL-BI Versão final. . . .	60
Tabela 11 – Comparação entre o AG+BL-FI e AG+BL-BI.	61
Tabela 12 – Comparação entre o AG+BL-FI e AG+BL-BI Versão Final. . .	61
Tabela 13 – Estatísticas do tempo de execução (em segundos) para os méto- dos avaliados.	63
Tabela 14 – Resumo comparativo dos métodos avaliados	73

Lista de abreviaturas e siglas

AG Algoritmos Genéticos

BL-FI Busca Local com *First Improvement*

BL-BI Busca Local com *Best Improvement*

EDD *Earliest Due Date*

GA *Genetic Algorithm*

GRASP *Greedy Randomized Adaptive Search*

HBC Hipótese dos Blocos Construtivos

PLIM Programação Linear Inteira Mista

PSUMAA Problema de Sequenciamento de Tarefas em uma Máquina com Penalidades por Antecipação e Atraso

SA *Simulated Annealing*

SPT *Shortest Processing Time*

TCS Teoria de Controle Supervisório

TE Teoria dos Esquemas

TS *Tabu Search*

VNS *Variable Neighborhood Search*

Sumário

1	INTRODUÇÃO	12
1.1	Motivação e Justificativa	14
1.2	Objetivos Geral e Específicos	15
1.3	Estrutura da Monografia	16
2	REVISÃO DA LITERATURA	17
2.1	Trabalhos Relacionados	17
2.2	Fundamentação Teórica	19
2.2.1	Problemas de Sequenciamento da Produção	19
2.2.2	O Problema de Sequenciamento em Uma Máquina com Penalidades por Antecipação e Atraso	23
2.2.3	Complexidade Computacional e Problemas NP-Difíceis	26
2.2.3.1	Definição do Problema	26
2.2.4	Métodos de Resolução	28
2.2.4.1	Heurísticas Construtivas	28
2.2.4.2	Busca Local	29
2.2.4.3	Meta-heurísticas	30
2.2.4.4	Algoritmo Genético	31
2.2.5	Combinação de Técnicas Heurísticas	33
2.3	Considerações Finais	34
3	DESENVOLVIMENTO	35
3.1	Tecnologias e Recursos Computacionais Utilizados	35
3.2	Modelagem	36
3.2.1	Representação Formal do Problema	37
3.2.1.1	Formulação Matemática do PSUMAA	37
3.2.2	Função de Avaliação	39
3.2.3	Representação da Solução	39
3.2.4	Definição da Vizinhança	40

3.3	Métodos de Resolução Aplicados	41
3.3.1	Construção da Solução Inicial	41
3.3.2	Busca Local	43
3.3.2.1	Busca Local – Best Improvement	44
3.3.2.2	Busca Local – <i>First Improvement</i>	45
3.3.3	Algoritmo Genético AG	46
3.3.4	Algoritmo Híbrido AG + Busca Local	49
3.4	Considerações Finais do Desenvolvimento	50
4	RESULTADOS	52
4.1	Conjunto de Instâncias utilizadas	52
4.2	Configuração do Ambiente Experimental	54
4.3	Análise dos Resultados Computacionais	54
4.4	Desempenho das Estratégias de Busca Local	55
4.4.1	Comparação entre AG Clássico e AG Híbrido	56
4.4.2	Análise Estatística do Tempo de Execução	62
4.5	Análise Gráfica e Estatística dos Resultados	65
4.5.1	Convergência em função do número de gerações	66
4.5.2	Distribuição do gap por método	67
4.5.3	Comparação do gap médio agregado	68
4.5.4	Impacto do Número de Gerações na Convergência por Instância	69
4.5.5	Resumo comparativo dos resultados	73
4.6	Considerações sobre os Resultados	74
5	CONCLUSÃO	76
5.1	Resultados e Contribuições	77
5.1.1	Análise dos Resultados Experimentais	77
5.1.2	Principais Contribuições	78
5.2	Limitações do trabalho	79
5.3	Trabalhos Futuros	79
	REFERÊNCIAS	81

1 Introdução

A crescente pressão da concorrência nos mercados tem levado as empresas a oferecerem uma ampla variedade de produtos personalizados, buscando atender às exigências cada vez maiores dos clientes, especialmente no que diz respeito à pontualidade das entregas. Nesse contexto, destaca-se o princípio do *Just-in-Time*, segundo o qual a produção e a entrega devem ocorrer exatamente no momento necessário, evitando tanto adiantamentos quanto atrasos (FELDMANN; BISKUP, 2003).

Diante dessa realidade, torna-se indispensável um controle rigoroso sobre os prazos de entrega, especialmente em ambientes produtivos complexos, nos quais é fundamental coordenar com precisão as diferentes etapas do processo. Esse tipo de organização temporal apresenta ampla aplicação em diversos setores industriais, como os segmentos têxtil, de tintas, de celulose e alimentício, que frequentemente enfrentam esse tipo de desafio. A literatura contempla diversos estudos voltados ao planejamento e ao sequenciamento da produção nesses contextos específicos (FILHO, 2007).

Nesse cenário, o Problema de Sequenciamento de Tarefas em uma Máquina com Penalidades por Antecipação e Atraso (PSUMAA) tem se mostrado particularmente relevante, tanto pela sua aplicabilidade prática quanto pela complexidade teórica envolvida. A relevância desse problema está relacionada ao fato de que concluir uma tarefa fora do tempo ideal pode acarretar custos adicionais – seja por antecipação, como capital imobilizado e necessidade de armazenagem, seja por atraso, comprometendo o cumprimento dos prazos acordados (PENNA et al., 2012).

Quanto aos prazos de entrega, Penna et al. (2012) classificam os problemas de sequenciamento em três categorias principais:

- **Data de entrega comum** (*common due date*): todas as tarefas devem ser concluídas até um mesmo instante;

- **Datas de entrega distintas** (*distinct due dates*): cada tarefa possui um prazo específico para sua finalização;
- **Janelas de entrega distintas** (*distinct due windows*): cada tarefa deve ser finalizada dentro de um intervalo de tempo previamente determinado.

Embora amplamente estudado, o **PSUMAA** é definido com um problema de elevada complexidade computacional, classificado como *NP-difícil*. O espaço de busca cresce de maneira fatorial com o número de tarefas ($n!$), tornando inviável a enumeração exaustiva mesmo para instâncias de médio porte. Desse modo, não há garantias de obtenção de soluções ótimas em tempo computacional viável para problemas de maior escala. Além disso, até onde foi identificado na literatura, não existe até então um algoritmo com custo polinomial para resolução do problema. Por esse motivo, a maioria das abordagens presentes na literatura utiliza técnicas heurísticas e meta-heurísticas, capazes de encontrar soluções satisfatórias de forma eficiente, mesmo sem assegurar a otimalidade. Já os métodos exatos, conforme apontado por [Oliveira e Simão \(2019\)](#), geralmente se limitam a instâncias de pequena escala, por não conseguirem lidar com a variabilidade e complexidade dos ambientes industriais reais.

Segundo [Júnior \(2007\)](#), as heurísticas, também denominadas métodos aproximativos, constituem estratégias voltadas à obtenção de boas soluções para problemas complexos, mesmo sem garantias formais de otimalidade. Sua principal vantagem está na capacidade de produzir respostas viáveis com baixo custo computacional, o que as torna especialmente atrativas para problemas de grande escala, mais próximos das condições encontradas na prática.

[Abualigah et al. \(2023\)](#) enfatizam que as meta-heurísticas, como algoritmos genéticos, otimização por enxame de partículas, recozimento simulado e colônia de formigas, têm se mostrado ferramentas eficazes para resolver problemas complexos de otimização. Entre essas abordagens, os algoritmos genéticos destacam-se por sua flexibilidade na representação de soluções baseadas em permutação, característica fundamental em problemas de sequenciamento. Além disso, permitem a incorporação natural de estratégias híbridas, como busca local e mecanismos adaptativos, favorecendo o equilíbrio entre exploração e intensificação do espaço de busca.

Diante dos desafios apresentados e da relevância do [PSUMAA](#), este trabalho propõe a implementação e avaliação experimental de um [AG](#) aplicado ao problema, bem como versões híbridas com mecanismos de intensificação e diversificação da busca. Busca-se analisar o impacto dessas estratégias na qualidade das soluções obtidas, no comportamento de convergência do método e no custo computacional associado, contribuindo para a compreensão do desempenho de abordagens meta-heurísticas no contexto do problema.

1.1 Motivação e Justificativa

Conforme destacado por [Filho \(2007\)](#), o interesse em problemas de programação de tarefas com penalidades por antecipação e atraso é relativamente recente quando comparado a métricas tradicionais como tempo médio de fluxo, atraso total ponderado ou *makespan*. No entanto, a crescente exigência por cumprimento rigoroso de prazos e redução de estoques tem ampliado a relevância prática dessas formulações.

No contexto de sistemas produtivos que operam com uma única máquina, a definição adequada da sequência de processamento das tarefas torna-se um fator crítico para o desempenho operacional. O alinhamento com os princípios do *Just-in-Time* reforça a necessidade de minimizar simultaneamente penalidades por antecipação e atraso, buscando maior sincronização entre produção e demanda.

O [PSUMAA](#) é classificado como NP-difícil ([DU; LEUNG, 1990](#)), o que implica que o esforço computacional necessário para a obtenção de soluções ótimas cresce rapidamente com o aumento do número de tarefas. Em ambientes industriais reais, nos quais o número de jobs pode ser elevado e as decisões precisam ser tomadas em tempo hábil, métodos exatos tornam-se impraticáveis.

Diante desse cenário, o emprego de meta-heurísticas apresenta-se como alternativa viável e estratégica. A motivação deste trabalho reside na necessidade de investigar estratégias computacionais capazes de produzir soluções de alta qualidade em tempo computacional reduzido, bem como analisar o impacto de mecanismos de intensificação e diversificação no desempenho de algoritmos evolucionários aplicados ao [PSUMAA](#).

1.2 Objetivos Geral e Específicos

O objetivo geral deste trabalho é investigar o desempenho de algoritmos genéticos aplicados ao [PSUMAA](#), analisando sua capacidade de produzir soluções de alta qualidade em termos de custo total de penalidades, bem como seu comportamento de convergência e custo computacional.

Os objetivos específicos são:

1. Implementar um [AG](#) clássico para resolução do [PSUMAA](#);
2. Desenvolver versões híbridas incorporando busca local como mecanismo de intensificação;
3. Introduzir estratégias adicionais, como elitismo, mutação adaptativa e inicialização heurística, visando aprimorar o desempenho do algoritmo;
4. Realizar experimentos computacionais com instância da literatura;
5. Avaliar e comparar os métodos em termos de melhor solução encontrada, gap médio, tempo médio de execução e estabilidade dos resultados.

A execução do trabalho compreende as seguintes etapas:

1. Revisão bibliográfica sobre o [PSUMAA](#) e métodos de solução;
2. Modelagem formal do problema;
3. Desenvolvimento e implementação das meta-heurísticas propostas;
4. Planejamento e execução de experimentos computacionais;
5. Análise crítica dos resultados obtidos;
6. Discussão das conclusões e identificação de possíveis extensões futuras.

1.3 Estrutura da Monografia

O restante deste trabalho está organizado de forma a apresentar, de maneira lógica e progressiva, as etapas que conduziram ao desenvolvimento da pesquisa e às conclusões alcançadas.

No [Capítulo 2](#), é apresentada a fundamentação teórica que sustenta o estudo, contemplando uma análise de trabalhos correlatos que aplicam técnicas heurísticas e meta-heurísticas, com ênfase em Algoritmos Genéticos, na resolução do Problema de Sequenciamento em Uma Máquina com Penalidades por Antecipação e Atraso (PSUMAA). São também discutidos os principais conceitos relacionados à modelagem matemática e à complexidade computacional do problema.

O [Capítulo 3](#) descreve detalhadamente a metodologia adotada, incluindo a modelagem do problema, a representação das soluções, a definição dos operadores genéticos, as estratégias de intensificação incorporadas e os critérios utilizados para definição dos parâmetros experimentais.

Em seguida, o [Capítulo 4](#) apresenta os experimentos computacionais realizados, bem como a análise quantitativa dos resultados obtidos. São avaliadas métricas como melhor solução encontrada, gap médio, tempo de execução e comportamento de convergência, permitindo a comparação entre as diferentes versões do algoritmo implementado.

Por fim, o [Capítulo 5](#) descreve as considerações finais acerca dos resultados obtidos, destacando as contribuições do trabalho, suas limitações e possíveis direções para pesquisas futuras.

2 Revisão da Literatura

Neste capítulo são apresentados os principais conceitos e estudos que fundamentam o desenvolvimento deste trabalho. O objetivo é contextualizar o [PSUMAA](#) no âmbito da otimização combinatória, destacando sua relevância prática e complexidade computacional.

A [Seção 2.1](#) discute pesquisas que empregaram meta-heurísticas — com ênfase em [AG](#) e abordagens híbridas — na resolução de problemas de sequenciamento e planejamento da produção. São analisadas suas principais contribuições, estratégias adotadas e limitações observadas.

A [Seção 2.2](#) apresenta os fundamentos teóricos necessários para compreensão das técnicas implementadas neste trabalho, abordando conceitos de heurísticas, meta-heurísticas, computação evolucionária e modelagem matemática do [PSUMAA](#).

Por fim, na [Seção 2.3](#), são sintetizadas as principais evidências encontradas na literatura, destacando lacunas que motivaram o desenvolvimento da proposta apresentada no capítulo seguinte.

2.1 Trabalhos Relacionados

Diferentes estudos exploram o uso de meta-heurísticas para a resolução do [PSUMAA](#), dada sua complexidade combinatória e classificação como um problema NP-difícil e o crescimento exponencial do espaço de busca conforme o número de tarefas aumenta.

Nesse contexto, destaca-se o trabalho de [Ribeiro, Souza e Souza \(2009\)](#), que propôs uma abordagem adaptativa baseada em [AG](#), incorporando *Greedy Randomized Adaptive Search* ([GRASP](#)) na geração da população inicial, operadores de cruzamento adaptativos com seleção baseada em desempenho histórico, busca local randômica utilizando movimentos de troca e realocação de tarefas, além da aplicação de *Path Relinking* para intensificação da busca. Os resultados computaci-

onais demonstraram a eficácia da abordagem, superando algoritmos tradicionais da literatura, especialmente em instâncias com grande número de tarefas, reforçando o potencial do AG na resolução de problemas complexos de otimização.

Seguindo uma linha semelhante de pesquisa, Sousa (2022) apresenta uma proposta voltada ao escalonamento de tarefas independentes em máquinas heterogêneas, utilizando também um AG híbrido. Embora trate de uma variação do problema, sua contribuição é relevante por combinar diferentes estratégias, como a geração inicial de soluções via programação linear relaxada, operadores genéticos clássicos e uma busca local voltada ao balanceamento de carga. Essa combinação favorece a diversidade das soluções e gera melhorias consistentes no *makespan*, especialmente em ambientes de grande escala.

Complementando esse panorama, o estudo de Spricigo (2018) amplia a aplicação dos AG ao integrá-los com a Teoria de Controle Supervisório (TCS), voltada ao controle de sistemas de manufatura. A principal contribuição desse trabalho está na proposta de um modelo híbrido, no qual os cromossomos são representados de três formas distintas e validados por supervisores automatizados, garantindo que as soluções geradas estejam de acordo com os requisitos de segurança da planta. Essa abordagem mostra-se eficaz para melhorar a produtividade em sistemas industriais automatizados, destacando o uso de AG também em contextos práticos e altamente controlados.

Além disso, Santos et al. (2010) realizaram um estudo de caso prático em uma indústria metalúrgica, aplicando AG ao PSUMAA em um cenário real. Ao comparar os resultados obtidos por meio de um AG e um Algoritmo Cultural, os autores observaram que o AG proporcionou melhor desempenho, gerando uma economia significativa na operação. O estudo se destaca por validar o uso de AG em situações reais de produção, reforçando sua aplicabilidade no setor industrial.

Por fim, aprofundando a base teórica por trás dos AG, Pinto, Martarelli e Nagano (2022) discute como a configuração dos operadores genéticos influencia diretamente o desempenho do algoritmo. Por meio da Teoria dos Esquemas (TE) e da Hipótese dos Blocos Construtivos (HBC), os autores alertam para os riscos de uma má parametrização, como a perda de diversidade populacional e a conver-

gência precoce. Essa análise oferece importantes diretrizes para pesquisadores e profissionais que desejam aplicar AG de maneira eficiente em problemas como o PSUMAA.

Apesar dos avanços apresentados, observa-se que grande parte das abordagens depende de parametrização fixa ou estruturas híbridas complexas, o que pode comprometer a escalabilidade e a simplicidade de implementação. Dessa forma, permanece relevante investigar estratégias que conciliem eficiência computacional, controle de diversidade e intensificação adaptativa de busca, mantendo uma estrutura algorítmica equilibrada.

2.2 Fundamentação Teórica

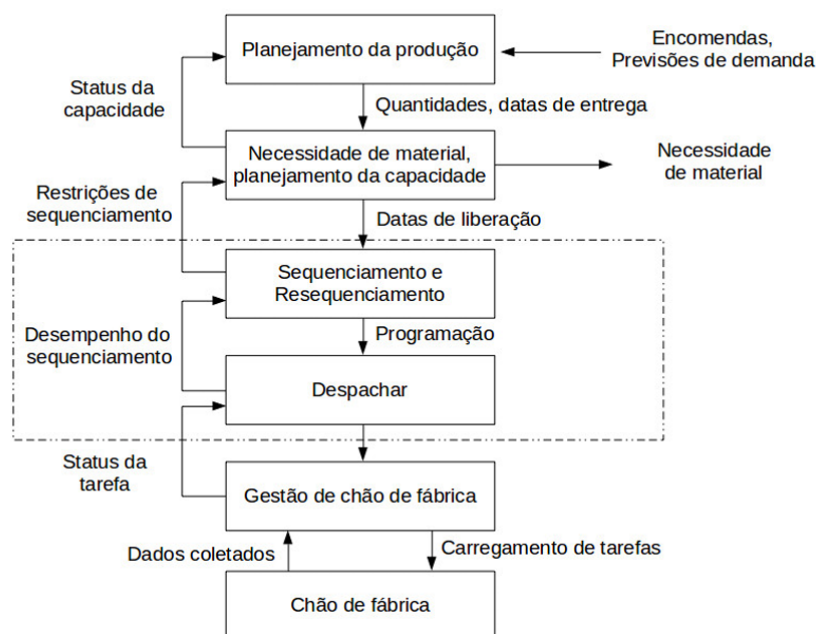
A fundamentação teórica apresenta os conceitos, métodos e ferramentas que sustentam o desenvolvimento deste trabalho, fornecendo a base necessária para a compreensão do problema abordado e das soluções propostas. Nesse contexto, são discutidos aspectos relacionados ao sequenciamento da produção, à complexidade computacional e às abordagens utilizadas para lidar com problemas de otimização de alta complexidade. Dessa forma, esta seção estabelece o suporte conceitual que orienta a modelagem do problema, bem como a implementação e a análise dos resultados apresentados posteriormente.

2.2.1 Problemas de Sequenciamento da Produção

O sequenciamento da produção é um tema central em Pesquisa Operacional, pois define a melhor ordem de execução de tarefas sujeitas a prazos, tempos de processamento, disponibilidade de máquinas e outras restrições. Segundo Pinedo (2012), esse processo faz parte de um fluxo mais amplo de planejamento e controle da manufatura, que envolve desde a previsão de demanda até o despacho final das atividades no chão de fábrica.

A Figura 1 ilustra esse fluxo de informação em um sistema produtivo, evidenciando como decisões de planejamento de capacidade e de materiais impactam diretamente na execução das tarefas.

Figura 1 – Fluxo de informações em um sistema produtivo.



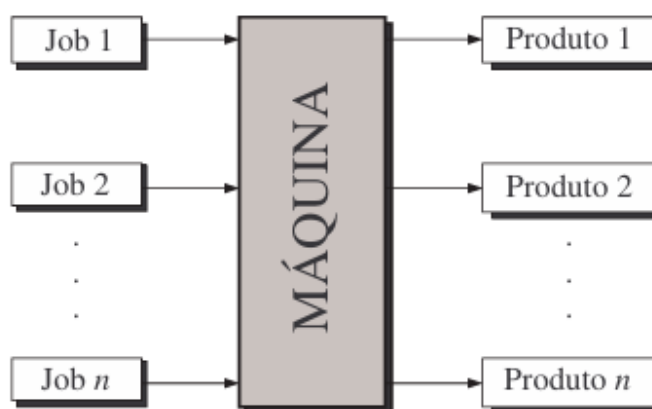
Fonte: Retirado de [Pinedo \(2012\)](#).

Diversos fatores influenciam a formulação dos problemas de sequenciamento, entre eles o tipo de recurso disponível no sistema produtivo. De maneira geral, os sistemas de produção podem ser classificados em quatro categorias principais: máquina única, máquinas paralelas, flow shop e job shop ([BAKER, 1974](#)). Cada uma dessas estruturas apresenta características específicas quanto à organização das operações, ao fluxo de processamento e ao nível de complexidade envolvido, conforme ilustrado a seguir.

• Máquina Única

No ambiente de máquina única, todos os jobs devem ser processados em um único recurso produtivo. Nesse tipo de sistema, apenas um job pode ser executado por vez, e o principal desafio consiste em definir a melhor ordem de processamento para minimizar critérios como tempo total de conclusão, atraso ou tempo médio de espera, conforme descrito por [Júnior \(2007\)](#).

Figura 2 – Sistema de produção com máquina única.

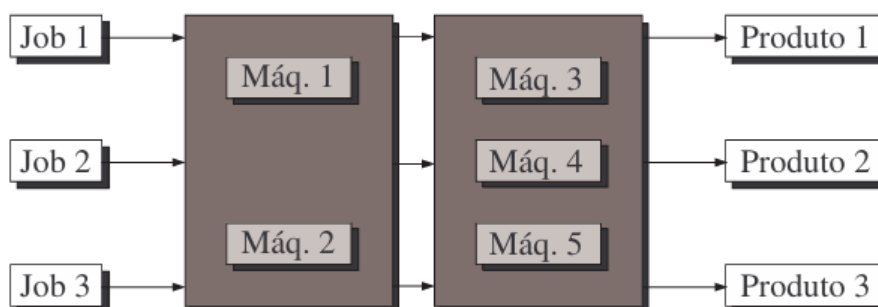


Fonte: (JúNIOR, 2007).

- **Máquinas Paralelas**

Os sistemas com máquinas paralelas possuem dois ou mais recursos produtivos capazes de processar os jobs de forma simultânea. Esses recursos podem ser idênticos ou apresentar pequenas diferenças de desempenho. O problema de sequenciamento, nesse caso, envolve não apenas a definição da ordem de processamento, mas também a alocação adequada dos jobs às máquinas disponíveis, conforme apresentado por Júnior (2007).

Figura 3 – Sistema de produção com máquinas paralelas.

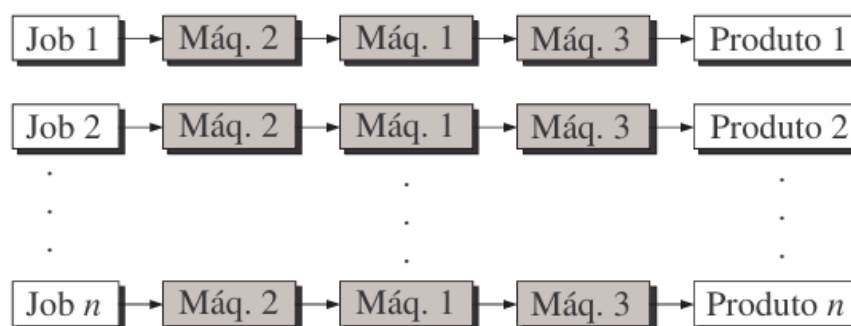


Fonte: (JúNIOR, 2007).

- **Flow Shop**

No ambiente *flow shop*, todos os jobs devem passar pelas mesmas máquinas, seguindo obrigatoriamente a mesma sequência de processamento. Esse tipo de sistema é comum em linhas de produção seriadas, nas quais o fluxo é padronizado. A principal dificuldade está em determinar a sequência dos jobs que minimize o tempo total de processamento ou evite gargalos ao longo da linha, conforme Júnior (2007).

Figura 4 – Sistema de produção do tipo flow shop.

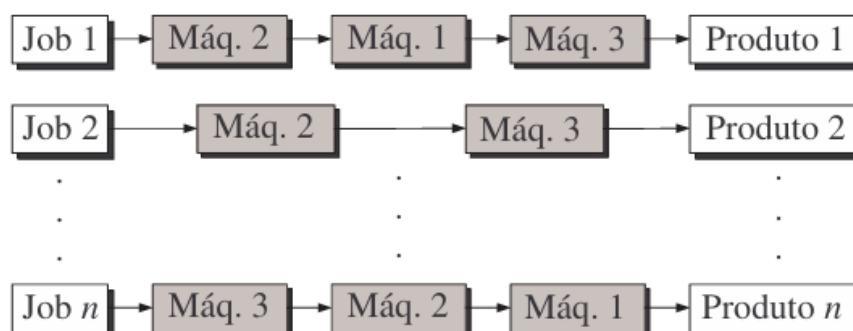


Fonte: (JúNIOR, 2007).

- **Job Shop**

O sistema *job shop* representa o cenário mais flexível e complexo entre os modelos apresentados. Nele, cada job possui uma sequência própria de máquinas, podendo diferir da ordem seguida por outros jobs. Essa flexibilidade torna o problema de sequenciamento computacionalmente mais difícil, exigindo frequentemente o uso de métodos heurísticos ou meta-heurísticos para obtenção de soluções de boa qualidade, conforme Júnior (2007).

Figura 5 – Sistema de produção do tipo job shop.



Fonte: (JúNIOR, 2007).

2.2.2 O Problema de Sequenciamento em Uma Máquina com Penalidades por Antecipação e Atraso

O **PSUMAA** caracteriza-se como uma extensão dos modelos clássicos de máquina única ao incorporar custos associados ao término antecipado ou tardio das tarefas. Seu objetivo é determinar uma sequência que minimize o total de penalidades, tornando-o adequado para sistemas que operam com forte restrição temporal e prazos rígidos (RODRIGUES; PIZZOLATO, 2014).

Nesse modelo, todas as tarefas estão disponíveis desde o início do horizonte de planejamento, e devem ser executadas sequencialmente por um único recurso. A Tabela 1 sintetiza os principais elementos do problema, como tempos de processamento, janelas de entrega, penalidades e tempos de setup dependentes da sequência.

Tabela 1 – Características consideradas na formulação do PSUMAA

Característica	Descrição
Disponibilidade das tarefas	Todas as tarefas podem ser iniciadas a partir do tempo zero.
Ambiente de processamento	Uma única máquina processa um conjunto de n tarefas.
Tempo de processamento	Cada tarefa i possui duração P_i .
Janela de entrega	Cada tarefa deve idealmente terminar no intervalo $[E_i, T_i]$.
Penalidade por antecipação	Finalizações antes de E_i acarretam penalidade proporcional ao adiantamento (α_i).
Penalidade por atraso	Conclusões após T_i geram penalidade proporcional ao atraso (β_i).
Conclusão dentro da janela	Não há penalidades quando o término ocorre entre E_i e T_i .
Preempção	O processamento não pode ser interrompido após iniciado.
Tempo ocioso	Podem existir períodos de ociosidade entre tarefas.
Tempo de setup	Entre duas tarefas consecutivas existe um setup S_{ij} ; para a primeira tarefa considera-se $S = 0$.

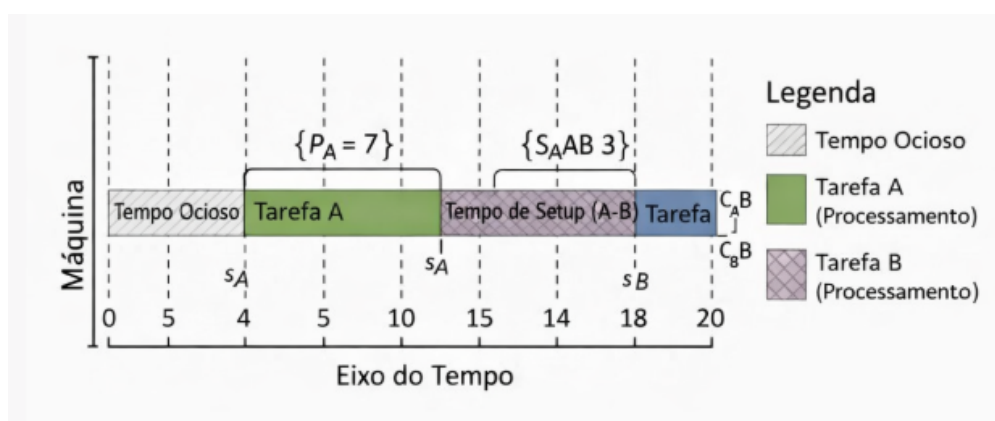
Fonte: Elaborada pela autora.

A aplicação desse modelo é comum em sistemas industriais sujeitos a variações operacionais, nos quais a transição entre tarefas pode impactar significativamente o desempenho do processo. Como discutido em [Ribeiro, Souza e Souza \(2009\)](#), o PSUMAA permite representar de forma mais fiel a dinâmica real de ambientes produtivos nos quais os tempos de preparação dependem da sequência executada, situação bastante recorrente em linhas de produção com ajustes frequentes.

A Figura 6 apresenta um diagrama de Gantt que exemplifica o sequenciamento de tarefas em uma única máquina considerando tempos de setup dependentes da sequência. Inicialmente, observa-se um período de tempo ocioso, no qual a máquina permanece disponível, porém sem processamento ativo. Em seguida, a Tarefa A é processada de forma contínua, iniciando no instante s_A e finalizando em C_A , que representa o tempo de conclusão da tarefa.

Após a conclusão da Tarefa A (C_A), ocorre um intervalo correspondente ao tempo de setup necessário para preparar a máquina para a execução da Tarefa B. Esse tempo de preparação depende da sequência das tarefas e não agrega valor produtivo, mas impacta diretamente o instante de início da próxima operação. Somente após o término do setup a Tarefa B é iniciada no instante s_B e processada de forma contínua até sua conclusão, representada por C_B .

Figura 6 – Diagrama de Gantt com tempos de setup dependentes da sequência



Fonte: Adaptado de Júnior (2007).

O diagrama evidencia que tanto os tempos de setup quanto os períodos de ociosidade influenciam diretamente o cronograma das tarefas, alterando seus instantes de início e término. Como consequência, esses fatores afetam o tempo de conclusão das tarefas e podem resultar em penalidades associadas à antecipação ou ao atraso em relação às datas estabelecidas.

Assim, a representação gráfica reforça o papel central dos tempos de preparação, das janelas de entrega e da ociosidade na definição da sequência ótima. Esses elementos influenciam diretamente o custo total associado ao cronograma final, o que torna o PSUMAA um problema de alta relevância tanto do ponto de vista acadêmico quanto industrial.

2.2.3 Complexidade Computacional e Problemas NP-Difíceis

Problemas de otimização podem ser classificados de acordo com o esforço computacional necessário para sua resolução. Enquanto alguns modelos admitem soluções exatas em tempo polinomial, outros apresentam elevado grau de dificuldade e pertencem à classe dos NP-difíceis, para os quais não existem algoritmos capazes de garantir solução ótima dentro de um tempo computacional viável. Nesses casos, o espaço de busca do problema cresce fatorialmente ($n!$), uma vez que todas as possíveis permutações das tarefas devem ser consideradas. Esse crescimento combinatorial torna impraticável a enumeração completa para instâncias de maior porte (PENNONE; MENDONÇA, 2019).

Nesse contexto, modelos que envolvem múltiplas restrições, como janelas de entrega, penalidades temporais e tempos de setup dependentes da sequência, tendem a apresentar maior complexidade combinatória. A literatura destaca que o PSUMAA enquadra-se como um problema NP-difícil (PENNA et al., 2012), de modo que a busca por soluções ótimas torna-se rapidamente impraticável conforme aumenta o número de tarefas.

Diante dessas limitações, torna-se necessária a adoção de métodos aproximativos capazes de equilibrar qualidade da solução e tempo de processamento. Estratégias como heurísticas, busca local e meta-heurísticas têm sido amplamente aplicadas, pois conseguem explorar o espaço de soluções de forma eficiente, produzindo resultados de boa qualidade em prazos compatíveis com aplicações reais (ROSA, 2019).

2.2.3.1 Definição do Problema

O PSUMAA consiste em determinar a melhor ordem de processamento de um conjunto finito de tarefas em uma única máquina, de modo a minimizar os custos associados à conclusão antecipada ou tardia das atividades.

Cada tarefa i possui:

- tempo de processamento P_i ;
- janela de entrega definida pelo intervalo $[E_i, T_i]$;

- custos unitários de antecipação c_i^e e atraso c_i^t .

Além disso, a transição da tarefa i para a tarefa j pode implicar um tempo de preparação dependente da sequência, representado por S_{ij} .

Seja C_i o instante de conclusão da tarefa i . As variáveis de antecipação e atraso são definidas pelas Equações (2.1) e (2.2), respectivamente:

$$e_i = \max(0, E_i - C_i) \quad (2.1)$$

$$t_i = \max(0, C_i - T_i) \quad (2.2)$$

A função objetivo, apresentada na Equação (2.3), visa minimizar o custo total associado às penalidades por antecipação, atraso e tempos de preparação dependentes da sequência:

$$\min \sum_{i \in J} (ce_i \cdot e_i + ct_i \cdot t_i) + \sum_{(i,j) \in S} c_{ij}^s \cdot S_{ij} \quad (2.3)$$

em que ce_i representa o custo unitário de antecipação da tarefa i , ct_i representa o custo unitário de atraso, e c_{ij}^s corresponde ao custo associado ao tempo de preparação entre as tarefas i e j .

Neste trabalho, considera-se um conjunto de tarefas J , no qual cada tarefa $i \in J$ possui um tempo de processamento P_i . Além disso, a transição entre duas tarefas consecutivas pode exigir um tempo de preparação dependente da sequência, representado por S_{ij} , que corresponde ao tempo necessário para preparar a máquina para executar a tarefa j imediatamente após a tarefa i .

O espaço de busca do problema corresponde ao conjunto de todas as permutações possíveis das tarefas, cujo tamanho é $n!$. Essa característica indica o crescimento fatorial do problema e reforça a necessidade do uso de técnicas aproximativas para sua resolução em instâncias de maior porte.

Assim, o **PSUMAA** configura-se como um problema clássico de apoio à decisão em ambientes produtivos, cuja solução eficiente contribui para a redução de custos operacionais e para a melhoria do nível de serviço.

2.2.4 Métodos de Resolução

A elevada complexidade do **PSUMAA** torna a aplicação de métodos exatos inviável para instâncias de maior escala, devido ao rápido crescimento do espaço de busca (MORAIS; ROCHA; CARVALHO, 2010). Nesse cenário, heurísticas e meta-heurísticas surgem como alternativas viáveis, oferecendo soluções suficientemente boas com esforço computacional reduzido.

Embora não garantam a otimalidade, essas técnicas apresentam desempenho adequado em ambientes industriais, especialmente quando o problema envolve janelas de entrega, penalidades de atraso e tempos de setup dependentes da sequência (SILVA; OLIVEIRA; MORAIS, 2007). Entre as heurísticas tradicionais destaca-se o método *Earliest Due Date* (**EDD**), amplamente utilizado na geração rápida de sequências iniciais, servindo como ponto de partida para métodos de melhoria posteriores.

2.2.4.1 Heurísticas Construtivas

As heurísticas construtivas constroem a solução passo a passo, inserindo tarefas até que a sequência esteja completa. Conforme discutem **Morais, Rocha e Carvalho (2010)**, esse tipo de abordagem permite gerar soluções iniciais viáveis com baixo custo computacional, mesmo que estas ainda possam ser aprimoradas posteriormente.

Devido à sua simplicidade e eficiência, heurísticas construtivas são frequentemente empregadas como etapa inicial de métodos de melhoria, como busca local e meta-heurísticas. Esse processo sequencial é especialmente útil em problemas combinatórios, nos quais o grande espaço de busca exige soluções iniciais consistentes e rapidamente geradas.

No âmbito do **PSUMAA**, heurísticas construtivas são frequentemente associadas a métodos de intensificação, tais como *Tabu Search* (**TS**), *Simulated*

Annealing (SA), GRASP, AG e VND, que refinam as sequências iniciais e promovem melhor qualidade de solução (SIMÕES; PEREIRA, 2016; FIDELIS; OLIVEIRA, 2017; ROSA, 2019).

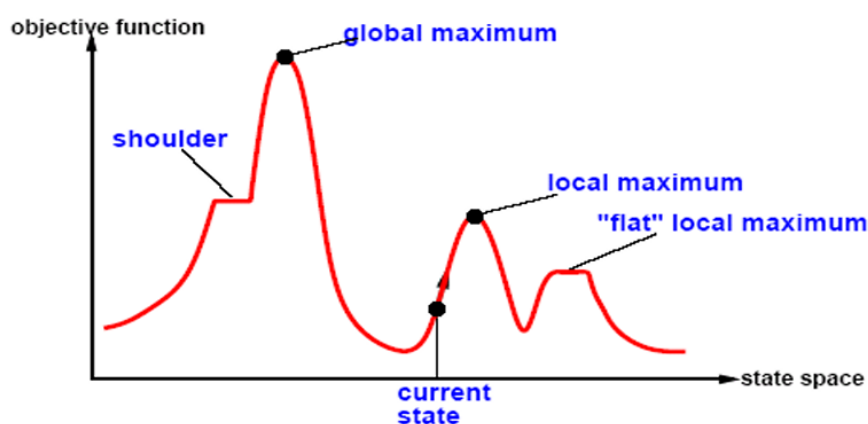
2.2.4.2 Busca Local

A busca local é uma técnica de aprimoramento que parte de uma solução viável e explora o espaço de vizinhança por meio de pequenas alterações na sequência, como trocas, inserções ou deslocamentos de tarefas Ferreira et al. (2018). A cada iteração, a solução é substituída por outra de menor custo, permitindo a redução progressiva da função objetivo.

No contexto do PSUMAA, pequenas alterações na ordem de execução podem afetar diretamente penalidades de atraso e tempos de setup, tornando a busca local uma estratégia eficaz. Seu desempenho depende da definição da vizinhança, dos movimentos permitidos e do critério de parada, que pode considerar número de iterações, limite de tempo ou ausência continuada de melhorias.

Uma das limitações da busca local é o risco de aprisionamento em mínimos locais, situação em que nenhuma solução vizinha oferece melhoria, resultando na estagnação do processo de otimização Ferreira et al. (2018). A Figura 7 ilustra esse comportamento, evidenciando a necessidade de mecanismos complementares, como reinicializações aleatórias, alteração dinâmica da vizinhança ou métodos probabilísticos, que permitem escapar de regiões subótimas e explorar áreas mais promissoras do espaço de soluções.

Figura 7 – Exemplo ilustrativo de aprisionamento em mínimo local em busca local



Fonte: Retirado de Russell e Norvig (2007).

Dessa forma, a busca local desempenha papel central na resolução do PSUMAA, aprimorando soluções inicialmente geradas por heurísticas construtivas e potencializando o desempenho de meta-heurísticas híbridas aplicadas ao problema.

2.2.4.3 Meta-heurísticas

As meta-heurísticas se diferenciam das heurísticas tradicionais por coordenarem mecanismos de busca capazes de realizar exploração mais abrangente do espaço de soluções, combinando intensificação e diversificação Morais, Rocha e Carvalho (2010). No caso específico do PSUMAA, o espaço de busca é composto por todas as possíveis permutações das tarefas, totalizando $n!$ sequências distintas. Esse crescimento fatorial torna impraticável a enumeração completa mesmo para valores moderados de n , justificando o uso de estratégias aproximativas.

Entre as meta-heurísticas mais utilizadas na literatura para o sequenciamento em máquina única destacam-se métodos consolidados, amplamente reconhecidos por sua capacidade de lidar com espaços de busca complexos e múltiplas restrições, entre eles:

- **SA** – inspirado no processo físico de têmpera, utiliza aceitação probabilística

de piores soluções para escapar de ótimos locais;

- **TS** – emprega memória adaptativa para impedir retornos a movimentos já explorados, ampliando a diversificação da busca;
- **Algoritmos Genéticos (AG)** – baseados nos princípios da evolução natural, operam sobre populações de soluções utilizando operadores como seleção, cruzamento e mutação;
- **Variable Neighborhood Search (VNS)** – alterna sistematicamente diferentes estruturas de vizinhança para intensificar e diversificar a busca;
- **Greedy Randomized Adaptive Search Procedure (GRASP)** – combina construção gulosa aleatória com fases de busca local, refinando progressivamente as soluções obtidas.

Essas abordagens diferem quanto à estratégia de exploração do espaço de busca, podendo operar de forma determinística (TS, VNS) ou probabilística (SA, AG), além de utilizarem memória adaptativa ou populações simultâneas de soluções. Tais diferenças estruturais impactam diretamente a capacidade de escapar de ótimos locais e a velocidade de convergência.

Esses métodos empregam mecanismos como históricos de movimentos, adaptação dinâmica da vizinhança e operadores probabilísticos para evitar o aprisionamento em ótimos locais. A literatura demonstra resultados competitivos para o PSUMAA, mesmo em cenários com múltiplas restrições (SIMÕES; PEREIRA, 2016; FIDELIS; OLIVEIRA, 2017; ROSA, 2019).

2.2.4.4 Algoritmo Genético

Os Algoritmos Genéticos (AG) são meta-heurísticas inspiradas no processo evolutivo da natureza, empregando operadores como seleção, recombinação e mutação para gerar soluções a partir de uma população inicial. Cada indivíduo representa uma solução candidata, avaliada por uma função de aptidão que orienta o processo evolutivo.

De acordo com [Ferreira et al. \(2018\)](#), os **AG** destacam-se pela capacidade de explorar diferentes regiões do espaço de busca simultaneamente, reduzindo o risco de convergência prematura. No contexto da produção, sua aplicação é frequente em problemas que envolvem múltiplas restrições, como o Job Shop e o sequenciamento industrial.

Para o **PSUMAA**, os **AG** mostram bom desempenho, desde que parâmetros como tamanho da população, taxa de cruzamento e intensidade da mutação sejam corretamente ajustados. Modelos estendidos, como cataclismos e epidemias, também podem ser aplicados para restaurar diversidade quando o algoritmo apresenta sinais de estagnação ([GUANG, 2012](#)).

A Tabela 2 apresenta, de forma organizada, os principais operadores utilizados no processo evolutivo dos Algoritmos Genéticos, destacando suas funções, objetivos e contribuições para a construção e aprimoramento das soluções ao longo das gerações.

Tabela 2 – Principais operadores dos Algoritmos Genéticos

Operador	Descrição
Seleção	Define os indivíduos que gerarão descendentes, priorizando aqueles com maior aptidão (PEREIRA, 2013).
Recombinação (Crossover)	Combina características de diferentes indivíduos, ampliando a exploração do espaço de busca (PEREIRA, 2013).
Mutação	Introduz pequenas alterações na solução, mantendo diversidade populacional e evitando convergência prematura (PEREIRA, 2013).
Cataclismo / Epidemia	Mecanismos opcionais que reinicializam parte da população quando o processo evolutivo perde diversidade (GUANG, 2012).
Função de aptidão	Avalia a qualidade da solução, considerando custos, restrições e penalidades do PSUMAA .
Parâmetros de controle	Definem configuração do algoritmo, como tamanho da população e taxa de cruzamento, influenciando a eficiência e a convergência.

Fonte: Elaborada pela autora.

Apesar da ampla utilização de **AG** no sequenciamento em máquina única,

observa-se na literatura que a eficiência do método depende fortemente da configuração de seus operadores e da incorporação de estratégias de intensificação e controle de diversidade. Dessa forma, investiga-se neste trabalho a implementação de mecanismos adicionais, como elitismo, mutação adaptativa e busca local aplicada seletivamente, visando melhorar a qualidade das soluções e reduzir estagnações durante o processo evolutivo.

2.2.5 Combinação de Técnicas Heurísticas

Diante das limitações de métodos isolados, uma abordagem amplamente explorada na literatura consiste na combinação de heurísticas e meta-heurísticas, aproveitando as forças de cada técnica para melhorar o equilíbrio entre exploração global e refinamento local (MORAIS; ROCHA; CARVALHO, 2010).

Uma estratégia comum consiste em utilizar heurísticas construtivas para gerar soluções iniciais de boa qualidade, posteriormente refinadas por busca local ou meta-heurísticas populacionais, como AG ou VNS (SIMÕES; PEREIRA, 2016). Esse procedimento reduz o tempo necessário para que o algoritmo alcance regiões promissoras, evitando gasto de esforço em soluções pouco competitivas.

Outra abordagem envolve o acoplamento direto entre métodos globais e locais. Em técnicas como GRASP, Busca Tabu e AG, cada solução gerada passa automaticamente por uma etapa de intensificação via busca local, garantindo ganhos progressivos de qualidade a cada iteração (PENNA et al., 2012).

Além disso, modelos híbridos podem incorporar adaptações dinâmicas de parâmetros, como intensidades de mutação, profundidade de vizinhança ou probabilidades de movimentação. Esses mecanismos permitem ao algoritmo ajustar-se às características da instância e evitar estagnação (ROSA, 2019).

Assim, métodos híbridos consolidam-se como alternativa robusta para o PSUMAA, proporcionando soluções de alta qualidade em tempos compatíveis com aplicações reais, mesmo em cenários com múltiplas restrições.

2.3 Considerações Finais

A fundamentação teórica apresentada contextualizou o [PSUMAA](#) no âmbito dos problemas de sequenciamento da produção, destacando sua relevância em ambientes industriais sujeitos a prazos rígidos, múltiplas restrições e tempos de setup dependentes da sequência. Evidenciou-se que o problema possui crescimento combinatorial de ordem fatorial ($n!$), caracterizando-se como NP-difícil e limitando a aplicação de métodos exatos em instâncias de maior porte.

Nesse contexto, heurísticas, métodos de busca local e meta-heurísticas consolidam-se como alternativas eficientes, capazes de produzir soluções de alta qualidade com esforço computacional compatível com aplicações reais. Técnicas como [EDD](#), [GRASP](#), [SA](#), [AG](#) e [VNS](#) demonstram desempenho consistente ao equilibrar intensificação e diversificação no espaço de busca.

A literatura também evidencia que abordagens híbridas tendem a apresentar resultados superiores aos obtidos por técnicas isoladas, ao combinar geração inicial estruturada, mecanismos de refinamento local e estratégias de controle de diversidade. Esse equilíbrio reduz o risco de estagnação em ótimos locais e melhora a robustez do processo de otimização.

Diante desse panorama, justifica-se a investigação de estratégias baseadas em Algoritmos Genéticos com mecanismos adicionais de intensificação e adaptação, buscando aprimorar o desempenho na resolução do [PSUMAA](#). O próximo capítulo apresenta a modelagem adotada e os procedimentos metodológicos implementados, detalhando as escolhas estruturais do algoritmo proposto.

3 Desenvolvimento

O presente capítulo tem como objetivo descrever o processo de desenvolvimento da solução computacional proposta para o **PSUMAA**. São detalhadas as escolhas metodológicas adotadas, os algoritmos implementados, as estruturas de dados utilizadas e os experimentos computacionais realizados.

A [Seção 3.1](#) apresenta as tecnologias e os recursos computacionais utilizados no desenvolvimento da solução. Em seguida, a [Seção 3.2](#) descreve a modelagem do problema, incluindo a representação formal do PSUMAA, sua formulação matemática, a definição da função de avaliação, a representação das soluções e a definição da vizinhança.

Na [Seção 3.3](#) são apresentados os métodos de resolução aplicados, abrangendo o procedimento de construção da solução inicial, as estratégias de busca local (*Best Improvement* e *First Improvement*), o algoritmo genético e a abordagem híbrida que combina algoritmo genético e busca local.

Por fim, a [Seção 3.4](#) apresenta as considerações finais do capítulo.

3.1 Tecnologias e Recursos Computacionais Utilizados

As tecnologias e ferramentas empregadas neste projeto foram selecionadas considerando a familiaridade prévia da autora, a robustez das soluções disponíveis e a adequação às necessidades específicas de implementação da meta-heurística e realização dos experimentos computacionais.

A linguagem de programação utilizada foi o **Python 3.12.4**, escolhida por sua versatilidade, legibilidade e ampla disponibilidade de bibliotecas científicas e numéricas, características que a tornam particularmente adequada para o desenvolvimento de algoritmos de otimização e experimentação computacional¹.

Para o desenvolvimento e execução dos códigos, utilizou-se o ambiente

¹ Documentação oficial do Python: <<https://www.python.org>>

Visual Studio Code, selecionado devido à sua integração eficiente com ambientes Python, suporte a extensões e recursos que facilitam a organização e depuração do código-fonte².

No que se refere às bibliotecas empregadas, destacam-se:

- **NumPy 1.26.4** – utilizada para manipulação eficiente de vetores e matrizes, além de suporte ao processamento numérico de alto desempenho³;
- **Math** – empregada para funções matemáticas necessárias ao cálculo da função objetivo e das penalidades;
- **Random** – utilizada na geração de números aleatórios, essenciais para os operadores evolutivos e mecanismos estocásticos da meta-heurística.

Os experimentos computacionais foram realizados em um computador equipado com processador Intel(R) Core(TM) i5-11300H (11^a geração), 16 GB de memória RAM e sistema operacional Windows 11 Home de 64 bits. A especificação do ambiente computacional é apresentada para garantir a reprodutibilidade dos experimentos e contextualizar o desempenho observado.

O código-fonte desenvolvido neste trabalho está disponível publicamente, permitindo a reprodução dos experimentos e a verificação dos resultados obtidos. A disponibilização do repositório contribui para a transparência e a reprodutibilidade científica, podendo ser acessado por meio do link⁴.

3.2 Modelagem

Nesta seção, apresenta-se a formalização do **PSUMAA**, estabelecendo os elementos necessários para a implementação das estratégias de resolução propostas. São descritos os componentes fundamentais do problema, sua formulação matemática, a função de avaliação utilizada para mensuração da qualidade das soluções e a definição das vizinhanças exploradas pelos métodos heurísticos e meta-heurísticos.

² Visual Studio Code: <<https://code.visualstudio.com>>

³ NumPy: <<https://numpy.org>>

⁴ <<https://github.com/marianavieiras/TCC>>

Essa estrutura estabelece a conexão entre a fundamentação teórica e o desenvolvimento computacional, permitindo a aplicação consistente da Busca Local, do [AG](#) e da abordagem híbrida na resolução do problema.

3.2.1 Representação Formal do Problema

O [PSUMAA](#) modela o processo de alocação de um conjunto de tarefas em uma única máquina, considerando restrições temporais, custos associados ao término das operações e tempos de preparação dependentes da sequência. Todas as tarefas estão disponíveis desde o tempo zero e devem ser processadas sem interrupção, sendo permitido inserir períodos de ociosidade quando isso contribui para a redução do custo total da solução.

Cada tarefa possui um tempo de processamento registrado e uma janela de entrega definida por um intervalo mínimo e máximo para o término desejado. Finalizações antes ou depois desses limites resultam em custos proporcionais, enquanto conclusões dentro da janela não geram penalidades adicionais. Além disso, quando duas tarefas consecutivas pertencem a categorias distintas, um tempo de setup pode ser necessário, alterando o instante real de início da tarefa subsequente.

Dessa forma, o [PSUMAA](#) busca determinar uma sequência de execução que minimize a soma dos custos totais de antecipação e atraso, levando em conta todos os fatores relacionados ao processamento, aos setups e às janelas de entrega. A [Tabela 1](#) já apresentou anteriormente os elementos que compõem essa formulação, que servem de base para o desenvolvimento da modelagem matemática, da função de avaliação e das estratégias heurísticas utilizadas neste trabalho.

3.2.1.1 Formulação Matemática do PSUMAA

A formulação matemática do [PSUMAA](#) estabelece como os tempos de término e os custos associados são determinados a partir de uma sequência de tarefas. Conforme apresentado por [Bustamante \(2007\)](#) e seguindo a estrutura clássica dos problemas de máquina única [Pinedo \(2012\)](#), o tempo de conclusão de cada tarefa depende de seu tempo de processamento e dos tempos de preparação entre tarefas consecutivas.

Considerando uma sequência π , o tempo de conclusão das tarefas é determinado recursivamente a partir dos tempos de processamento e dos tempos de preparação dependentes da sequência. O modelo de Programação Linear Inteira Mista (PLIM) proposto por Júnior (2007) formaliza essa dinâmica por meio da seguinte estrutura:

$$\text{Minimizar } Z = \sum_{i=1}^n (\alpha_i e_i + \beta_i t_i) \quad (3.1)$$

Sujeito a:

$$\sum_{j=1, j \neq i}^{n+1} y_{ij} = 1 \quad \forall i \in \{0, \dots, n\} \quad (3.2)$$

$$\sum_{i=0, i \neq j}^n y_{ij} = 1 \quad \forall j \in \{1, \dots, n+1\} \quad (3.3)$$

$$s_j - s_i - y_{ij}(M + S_{ij}) \geq P_i - M \quad \forall i, j, i \neq j \quad (3.4)$$

$$s_i + P_i + e_i \geq E_i \quad \forall i \in \{1, \dots, n\} \quad (3.5)$$

$$s_i + P_i - t_i \leq T_i \quad \forall i \in \{1, \dots, n\} \quad (3.6)$$

$$s_i, e_i, t_i \geq 0, \quad y_{ij} \in \{0, 1\} \quad \forall i, j \quad (3.7)$$

A partir dos valores de conclusão, as penalidades de antecipação (e_i) e atraso (t_i) são calculadas conforme as Equações (3.5) e (3.6), compondo a função objetivo definida na Equação (3.1). O objetivo consiste em minimizar o custo total associado à sequência, combinando penalidades por antecipação e atraso em relação às janelas de entrega de cada tarefa.

Esse custo é calculado a partir da soma ponderada dos adiantamentos e atrasos, conforme definido na função objetivo do modelo apresentado nesta seção, baseada na formulação proposta por Rodrigues e Pizzolato (2014). O problema considera ainda as restrições de processamento em uma única máquina, ausência de preempção das tarefas e tempos de setup dependentes da sequência, garantidos pela restrição (3.4).

Essa formulação constitui a base matemática utilizada para representar o problema e avaliar a qualidade de qualquer sequência candidata. O modelo considera as hipóteses clássicas do [PSUMAA](#), incluindo o processamento de um conjunto de tarefas em uma única máquina, sem preempção, com tempos de setup dependentes da sequência, bem como penalidades associadas à antecipação e ao atraso em relação às janelas de entrega definidas para cada tarefa.

3.2.2 Função de Avaliação

A função de avaliação implementada neste trabalho utiliza a formulação do [PSUMAA](#) como mecanismo para atribuir um custo a cada sequência gerada pelas heurísticas e meta-heurísticas. Seu papel é estritamente computacional, servindo como critério para comparar, selecionar e orientar a melhoria das soluções.

Dada uma ordem de processamento, calculam-se os tempos de término considerando os tempos de processamento e os setups dependentes da sequência. Em seguida, determinam-se os valores de antecipação e atraso conforme as janelas estabelecidas [Bustamante \(2007\)](#):

$$e_i = \max(0, E_i - C_i), \quad t_i = \max(0, C_i - T_i).$$

O custo total associado à solução é dado por:

$$f(\pi) = \sum_{i=1}^n (ce_i \cdot e_i + ct_i \cdot t_i),$$

correspondendo ao valor utilizado pelos métodos heurísticos para comparar o desempenho entre diferentes sequências [Rodrigues e Pizzolato \(2014\)](#).

Essa função é crucial nas etapas de Busca Local e [AG](#), pois orienta a seleção de soluções promissoras e possibilita o refinamento progressivo ao longo do processo de otimização.

3.2.3 Representação da Solução

A solução adotada para o [PSUMAA](#) neste trabalho é representada por uma permutação das tarefas. Cada solução candidata é expressa como uma sequência

$$\pi = (\pi(1), \pi(2), \dots, \pi(n)),$$

em que $\pi(k)$ indica a tarefa processada na posição k . Essa representação é simples, direta e adequada para a aplicação dos operadores utilizados na Busca Local e no [AG](#).

A partir da sequência, calcula-se o tempo de término de cada tarefa considerando os tempos de processamento e preparação. Posteriormente, determinam-se os valores de antecipação e atraso, que compõem o custo total da solução.

Considere um conjunto de cinco tarefas $\{1, 2, 3, 4, 5\}$. Uma possível solução para o [PSUMAA](#) pode ser representada pela permutação:

$$\pi = (3, 1, 5, 2, 4).$$

A partir dessa ordem, o cálculo do tempo de término segue o modelo:

$$C_{\pi(k)} = C_{\pi(k-1)} + S_{\pi(k-1)\pi(k)} + P_{\pi(k)}.$$

Essa estrutura permite que os métodos de otimização realizem operações como troca, inserção e inversão para gerar novas soluções viáveis.

3.2.4 Definição da Vizinhança

A busca implementada neste trabalho opera sobre soluções representadas por permutações. Dessa forma, a definição da vizinhança deve preservar essa estrutura, garantindo a viabilidade das soluções geradas.

Adotou-se o operador de troca (*swap*), no qual duas posições distintas da sequência têm suas tarefas permutadas, produzindo uma nova solução candidata. Esse movimento é simples, computacionalmente eficiente e amplamente empregado em problemas de máquina única, pois mantém a integridade da permutação sem introduzir duplicações ou omissões de tarefas.

A vizinhança definida por esse operador é explorada diretamente pelos métodos de melhoria descritos na subseção seguinte.

3.3 Métodos de Resolução Aplicados

Esta seção apresenta os métodos de resolução empregados para abordar o PSUMAA. São descritas as estratégias adotadas para construção, avaliação e melhoria das soluções, incluindo a Busca Local, o AG e a abordagem híbrida que integra ambas as técnicas.

Cada método é detalhado em termos de funcionamento, operadores utilizados, critérios de aceitação e papel desempenhado no processo de otimização. O objetivo é demonstrar, de forma estruturada, como as soluções são geradas, refinadas e selecionadas ao longo do processo, evidenciando a contribuição de cada abordagem para a obtenção de sequências de menor custo no contexto do PSUMAA.

3.3.1 Construção da Solução Inicial

A construção da solução inicial tem como objetivo gerar uma primeira sequência viável de tarefas, que servirá como ponto de partida para os métodos de otimização aplicados neste trabalho, tais como a Busca Local, o AG e a abordagem híbrida AG + Busca Local.

Diferentemente de uma simples permutação aleatória, neste trabalho é utilizada uma heurística construtiva baseada na regra *Shortest Processing Time* (SPT). Essa regra consiste em ordenar as tarefas em ordem crescente de seus tempos de processamento P_i .

A adoção da heurística SPT justifica-se por sua ampla utilização na literatura de problemas de sequenciamento, sendo reconhecida por gerar soluções iniciais de boa qualidade e por reduzir o tempo médio de conclusão em diversos cenários clássicos. Embora não seja necessariamente ótima para o problema específico tratado neste trabalho, sua utilização fornece uma solução estruturada inicial capaz de direcionar os métodos de melhoria para regiões promissoras do espaço de busca.

Computacionalmente, a solução inicial é obtida ordenando-se as tarefas de acordo com seus respectivos tempos de processamento, garantindo a geração de uma permutação válida, na qual cada tarefa aparece exatamente uma vez na

sequência resultante.

O Algoritmo 1 apresenta o procedimento adotado.

Algoritmo 1 Geração da Solução Inicial pela Regra SPT

- 1: **Entrada:** conjunto de tarefas $\{1, 2, \dots, n\}$ com tempos de processamento P_i
 - 2: **Saída:** solução inicial π
 - 3: Ordenar as tarefas em ordem crescente de P_i
 - 4: $\pi \leftarrow$ sequência ordenada resultante
 - 5: **return** π
-

A Tabela 3 ilustra um exemplo de instância utilizada nos experimentos, contendo os tempos de processamento, janelas de entrega e custos de antecipação e atraso das tarefas. Em seguida, apresenta-se a matriz de *setup*, representando os tempos de preparação dependentes da sequência.

Tabela 3 – Exemplo de dados de entrada para uma instância com 8 tarefas.

Job	P_i	E_i	T_i	α_i	β_i
1	82	482	532	69	96
2	47	562	568	4	98
3	40	442	492	1	32
4	47	299	407	32	98
5	48	304	386	12	88
6	32	286	338	19	29
7	51	495	591	13	83
8	90	403	405	20	22

Fonte: Elaborada pela autora.

Tabela 4 – Matriz de tempos de setup S_{ij} .

	1	2	3	4	5	6	7	8
1	0	10	47	39	45	1	29	12
2	10	0	28	5	41	1	29	2
3	47	28	0	32	10	22	4	17
4	39	5	32	0	27	4	27	31
5	45	41	10	27	0	30	48	25
6	1	1	22	4	30	0	46	30
7	29	29	4	27	48	46	0	10
8	12	2	17	31	25	30	10	0

Fonte: Elaborada pela autora.

A solução inicial gerada será a sequência resultante da ordenação crescente dos tempos de processamento. Considerando os dados da Tabela 3, a ordenação segundo a regra SPT produz, por exemplo:

$$\pi = (6, 3, 2, 4, 5, 7, 1, 8)$$

Essa sequência será então utilizada como ponto de partida para a Busca Local e como indivíduo inicial na população do AG.

3.3.2 Busca Local

Com o objetivo de analisar diferentes estratégias de exploração da vizinhança, foram implementadas duas variações do método de Busca Local baseadas no operador *swap*: a estratégia de melhor melhoria (*Best Improvement*) e a estratégia de primeira melhoria (*First Improvement*).

Ambas as abordagens partem de uma solução inicial e exploram a vizinhança definida por todas as trocas possíveis entre pares de tarefas. Considerando uma instância com n tarefas, a vizinhança baseada em *swap* é composta por $\frac{n(n-1)}{2}$ soluções vizinhas possíveis.

A principal diferença entre os métodos está no critério de aceitação do movimento e na forma como a vizinhança é percorrida. Enquanto a estratégia de melhor melhoria avalia completamente a vizinhança antes de atualizar a solução

corrente, a estratégia de primeira melhoria aceita imediatamente a primeira solução que apresente redução de custo.

A utilização de ambas as estratégias permite avaliar o impacto da forma de exploração da vizinhança no desempenho do algoritmo, tanto em termos de qualidade da solução quanto de esforço computacional.

3.3.2.1 Busca Local – Best Improvement

A estratégia de melhor melhoria, também conhecida como *Steepest Descent*, examina toda a vizinhança da solução corrente antes de realizar qualquer movimentação. Durante cada iteração, todas as trocas possíveis entre pares de tarefas são avaliadas, e o melhor vizinho encontrado é selecionado para substituir a solução atual, caso apresente custo inferior.

Essa abordagem tende a produzir reduções mais significativas no valor da função objetivo a cada iteração, porém exige maior esforço computacional devido à avaliação completa da vizinhança.

Algoritmo 2 Busca Local – Best Improvement

```

1: Entrada: solução inicial  $\pi$ , número de tarefas  $n$ 
2: Saída: solução melhorada  $\pi^*$ 
3:  $atual \leftarrow \pi$ 
4:  $custoAtual \leftarrow custo(atual)$ 
5:  $melhorou \leftarrow \text{verdadeiro}$ 
6: while  $melhorou$  do
7:    $melhorou \leftarrow \text{falso}$ 
8:    $melhorVizinho \leftarrow atual$ 
9:    $melhorCusto \leftarrow custoAtual$ 
10:  for  $i = 1$  até  $n - 1$  do
11:    for  $j = i + 1$  até  $n$  do
12:      Gerar vizinho trocando posições  $i$  e  $j$ 
13:       $c \leftarrow custo(vizinho)$ 
14:      if  $c < melhorCusto$  then
15:         $melhorVizinho \leftarrow vizinho$ 
16:         $melhorCusto \leftarrow c$ 
17:      end if
18:    end for
19:  end for
20:  if  $melhorCusto < custoAtual$  then
21:     $atual \leftarrow melhorVizinho$ 
22:     $custoAtual \leftarrow melhorCusto$ 
23:     $melhorou \leftarrow \text{verdadeiro}$ 
24:  end if
25: end while
26: return  $atual$ 

```

3.3.2.2 Busca Local – *First Improvement*

A estratégia de primeira melhoria, também denominada *First Descent*, percorre a vizinhança de forma sequencial e aceita imediatamente a primeira solução que apresente redução no custo da função objetivo.

Assim que uma melhoria é identificada, a solução corrente é atualizada e a exploração da vizinhança é reiniciada a partir da nova configuração. Essa estratégia reduz o número de avaliações realizadas por iteração, podendo resultar em menor tempo computacional quando comparada à abordagem de melhor melhoria.

Algoritmo 3 Busca Local – *First Improvement*

```

1: Entrada: solução inicial  $\pi$ , número de tarefas  $n$ 
2: Saída: solução melhorada  $\pi^*$ 
3:  $atual \leftarrow \pi$ 
4:  $custoAtual \leftarrow custo(atual)$ 
5:  $melhorou \leftarrow \text{verdadeiro}$ 
6: while  $melhorou$  do
7:    $melhorou \leftarrow \text{falso}$ 
8:   for  $i = 1$  até  $n - 1$  do
9:     for  $j = i + 1$  até  $n$  do
10:      Gerar vizinho trocando posições  $i$  e  $j$ 
11:       $c \leftarrow custo(vizinho)$ 
12:      if  $c < custoAtual$  then
13:         $atual \leftarrow vizinho$ 
14:         $custoAtual \leftarrow c$ 
15:         $melhorou \leftarrow \text{verdadeiro}$ 
16:        interromper ambos os laços
17:      end if
18:    end for
19:  end for
20: end while
21: return  $atual$ 

```

Em ambas as estratégias, o processo é repetido até que nenhuma solução vizinha apresente melhoria, caracterizando a convergência para um ótimo local em relação à vizinhança definida pelo operador *swap*.

3.3.3 Algoritmo Genético AG

O AG desenvolvido neste trabalho segue o modelo evolucionário baseado em populações, utilizando representação por permutação adequada ao contexto do PSUMAA. O método combina mecanismos de exploração global e intensificação controlada, incorporando estratégias adicionais como inicialização parcialmente heurística, elitismo explícito, mutação adaptativa e critério de parada por estagnação.

A representação por permutação é amplamente empregada em algoritmos genéticos aplicados a problemas de sequenciamento e escalonamento. Conforme

descrito por Guzek, Pióro e Żotkiewicz (2014), a organização dos genes no cromossomo pode refletir uma permutação que estabelece uma ordem de prioridade entre os elementos, sendo especialmente adequada para problemas cuja decisão consiste na definição de uma sequência de execução, como no PSUMAA.

Assim, cada indivíduo é modelado como uma permutação das tarefas $\{1, 2, \dots, n\}$, representando diretamente a ordem de processamento na máquina única e assegurando a viabilidade das soluções ao longo do processo evolutivo.

Componentes do Algoritmo

- **Inicialização parcialmente heurística:** A população inicial é composta por duas partes: 30% dos indivíduos são gerados por uma heurística construtiva baseada na regra SPT (Shortest Processing Time), enquanto 70% são gerados aleatoriamente. Essa estratégia equilibra qualidade inicial e diversidade populacional.
- **Tamanho da população:** 40 indivíduos.
- **Função de avaliação:** O custo de cada solução é calculado pela função objetivo do PSUMAA, considerando tempos de processamento, tempos de preparação dependentes da sequência e penalidades por antecipação e atraso.
- **Seleção – Tournament Selection:** Três indivíduos são selecionados aleatoriamente e o de menor custo é escolhido como pai.
- **Cruzamento – Order Crossover (OX):** Operador apropriado para representações por permutação, preservando a ordem relativa das tarefas e evitando duplicações.
- **Mutação – Swap Mutation com taxa adaptativa:** A mutação consiste na troca de duas posições da permutação. A taxa de mutação é adaptativa, sendo maior nas primeiras gerações e reduzida progressivamente ao longo do processo evolutivo.
- **Elitismo explícito:** Os 5% melhores indivíduos da população são preservados a cada geração.

- **Critério de parada:** O algoritmo encerra quando atinge o número máximo de gerações (100, 1.000 ou 10.000) ou quando não há melhoria da melhor solução por 30 gerações consecutivas.

Pseudocódigo do AG

O pseudocódigo do AG é apresentado no Algoritmo 4.

Algoritmo 4 AG com Elitismo e Mutação Adaptativa

```

1: Inicializar população (30% heurística SPT + 70% aleatória)
2: Avaliar população
3: (bestSol, bestCost) ← melhor indivíduo
4: stagnant ← 0
5: for gen = 1 até generations do
6:   Ordenar população
7:   Preservar elite (5%)
8:   Calcular taxa de mutação adaptativa
9:   newPopulation ← elite
10:  while |newPopulation| < pop_size do
11:    Selecionar dois pais via torneio
12:    Gerar filho via OX
13:    if probabilidade de mutação satisfeita then
14:      Aplicar mutação swap
15:    end if
16:    Avaliar filho
17:    Inserir em newPopulation
18:  end while
19:  population ← newPopulation
20:  Atualizar melhor solução global
21:  if não houve melhoria then
22:    stagnant ← stagnant + 1
23:  else
24:    stagnant ← 0
25:  end if
26:  if stagnant ≥ 30 then
27:    break
28:  end if
29: end for
30: return melhor solução encontrada

```

3.3.4 Algoritmo Híbrido AG + Busca Local

O algoritmo híbrido combina o AG com procedimentos de Busca Local, caracterizando uma abordagem do tipo memética. O AG é responsável pela exploração global do espaço de busca, enquanto a Busca Local promove intensificação controlada em regiões promissoras.

Após a geração da nova população, aplica-se a Busca Local apenas nos 10% melhores indivíduos da geração corrente, reduzindo o custo computacional e mantendo eficiência.

Foram consideradas duas variantes de Busca Local:

- Busca Local *First Improvement*;
- Busca Local *Best Improvement*.

Ambas utilizam o operador *swap*, explorando a vizinhança por trocas de duas posições na permutação.

Essa integração proporciona:

- Intensificação seletiva;
- Melhoria da qualidade média da população;
- Redução do risco de convergência prematura;
- Melhor equilíbrio entre exploração global e refinamento local.

Pseudocódigo do Algoritmo Híbrido

O pseudocódigo do Algoritmo Híbrido proposto é apresentado no Algoritmo 5.

Algoritmo 5 Algoritmo Híbrido AG + Busca Local

```
1: Inicializar população (30% heurística + 70% aleatória)
2: Avaliar população
3: stagnant ← 0
4: for gen = 1 até generations do
5:   Ordenar população
6:   Preservar elite (5%)
7:   Gerar nova população via seleção, OX e mutação adaptativa
8:   Ordenar nova população
9:   Aplicar Busca Local nos 10% melhores indivíduos
10:  Atualizar melhor solução global
11:  if não houver melhoria por 30 gerações then
12:    break
13:  end if
14: end for
15: return melhor solução encontrada
```

3.4 Considerações Finais do Desenvolvimento

O desenvolvimento deste trabalho envolveu a implementação progressiva de diferentes estratégias de otimização aplicadas ao [PSUMAA](#), permitindo a construção de uma estrutura algorítmica modular e extensível.

Inicialmente, foram implementados separadamente o [AG](#) e os procedimentos de Busca Local, o que possibilitou compreender o comportamento individual de cada abordagem. A organização do código em módulos independentes favoreceu testes controlados, validação de operadores e posterior integração entre os métodos.

Ao longo da implementação, decisões importantes de projeto foram adotadas, como a utilização de representação por permutação, operadores específicos para problemas de sequenciamento (*Order Crossover* e mutação por *swap*), elitismo explícito e critério de parada por estagnação. Além disso, optou-se por uma inicialização parcialmente heurística da população, combinando soluções baseadas na regra SPT com indivíduos gerados aleatoriamente, buscando equilíbrio entre qualidade inicial e diversidade.

A estrutura genérica do núcleo evolutivo permitiu a criação de diferentes va-

riantes do algoritmo, incluindo versões híbridas com Busca Local *First Improvement* e *Best Improvement*. Essa flexibilidade tornou o sistema mais adaptável e facilitou a comparação entre estratégias sem necessidade de reestruturação significativa do código.

O processo de desenvolvimento também evidenciou a importância da clareza na definição da vizinhança, da consistência na avaliação das soluções e da organização dos critérios de parada para garantir estabilidade e reprodutibilidade dos experimentos.

De modo geral, o trabalho resultou em uma base computacional sólida, modular e preparada para futuras extensões e aprimoramentos. O algoritmo proposto difere dos modelos clássicos principalmente pela utilização de inicialização heurística (SPT), elitismo, mutação adaptativa e, sobretudo, pela hibridização com Busca Local. Enquanto o AG tradicional foca na exploração global do espaço de busca, a abordagem híbrida permite também o refinamento local das soluções. Como resultado, observa-se um melhor desempenho das versões híbridas em relação ao AG clássico.

4 Resultados

Este capítulo apresenta os resultados obtidos com os métodos implementados para a resolução do problema **PSUMAA**. O objetivo é avaliar o desempenho do **AG** clássico, das estratégias de Busca Local aplicadas isoladamente e das versões híbridas **AG** + Busca Local.

A comparação entre os métodos considera qualidade de solução, estabilidade estatística e custo computacional, utilizando como referência valores ótimos ou melhores valores conhecidos disponíveis na literatura.

4.1 Conjunto de Instâncias utilizadas

Foram utilizadas instâncias clássicas da literatura propostas por [Rosa et al. \(2017\)](#), amplamente empregadas em estudos anteriores sobre o **PSUMAA**. Essas instâncias são disponibilizadas juntamente com valores ótimos ou melhores soluções conhecidas, permitindo o cálculo do *gap* percentual.

As instâncias abrangem diferentes níveis de complexidade, variando de 7 a 50 tarefas. Essa diversidade possibilita avaliar o comportamento dos algoritmos tanto em problemas de pequeno porte quanto em cenários de maior dimensão e dificuldade combinatória.

Cada instância é composta por um arquivo `.dat`, contendo os dados do problema, e um arquivo `.sol`, que apresenta a solução de referência utilizada para o cálculo do *gap*. O *gap* percentual é calculado conforme a Equação 4.1, medindo a diferença relativa entre o valor obtido pelo método heurístico e o melhor valor conhecido, conforme prática adotada na literatura de programação da produção (por exemplo, ([JúNIOR, 2007](#))).

$$\text{gap}(\%) = \frac{C_{\text{sol}} - C_{\text{best}}}{C_{\text{best}}} \times 100 \quad (4.1)$$

Sendo que: C_{sol} representa o custo da solução obtida pelo método heurístico

e C_{best} corresponde ao melhor valor conhecido.

A Tabela 5 apresenta as instâncias utilizadas nos experimentos e o respectivo número de tarefas.

Tabela 5 – Instâncias utilizadas nos experimentos.

Instância	Número de tarefas
INST0703	7
INST0801	8
INST1001	10
INST1010	10
INST2001	20
INST2002	20
INST2003	20
INST2004	20
INST2005	20
INST2006	20
INST2007	20
INST2008	20
INST2009	20
INST2010	20
INST2011	20
INST2012	20
INST2501	25
INST2507	25
INST3001	30
INST4001	40
INST5001	50

Fonte: Elaborada pela autora.

Contudo, os parâmetros do Algoritmo Genético foram definidos com base em práticas consolidadas na literatura e em testes preliminares realizados neste trabalho. Foram conduzidos experimentos exploratórios variando o tamanho da população, a taxa de cruzamento e a taxa de mutação, buscando um equilíbrio entre qualidade da solução e tempo computacional.

4.2 Configuração do Ambiente Experimental

Os experimentos computacionais foram realizados em ambiente com as seguintes configurações:

- Processador: Intel(R) Core(TM) i5-11300H (11^a geração)
- Memória RAM: 16 GB
- Sistema Operacional: Windows 11 Home (64 bits)
- Linguagem de programação: Python 3.12

Devido à natureza estocástica dos algoritmos evolutivos, cada método foi executado 30 vezes para cada instância, permitindo análise estatística dos resultados e redução do efeito de variações aleatórias.

As seguintes métricas foram coletadas:

- Melhor custo obtido
- Custo médio
- Desvio padrão
- *Gap* médio (%)
- Tempo médio de execução
- Tempo mínimo e máximo de execução

4.3 Análise dos Resultados Computacionais

Nesta seção são apresentados e discutidos os resultados obtidos a partir da aplicação dos métodos avaliados. A análise considera tanto a qualidade das soluções geradas, medida pelo valor da função objetivo e pelo *gap* percentual, quanto o tempo computacional necessário para a obtenção dessas soluções em cada abordagem. Dessa forma, busca-se estabelecer uma comparação abrangente entre desempenho, eficiência e escalabilidade.

4.4 Desempenho das Estratégias de Busca Local

As estratégias de busca local foram inicialmente avaliadas de forma isolada, com o objetivo de analisar seu desempenho sem a influência de métodos populacionais ou abordagens híbridas. Foram consideradas duas variações: **BL-FI** e **BL-BI**. Cada método foi executado 30 vezes por instância, permitindo avaliar a qualidade média das soluções obtidas, a variabilidade dos resultados e o tempo computacional requerido.

A Tabela 6 apresenta os resultados médios obtidos para um conjunto representativo de instâncias, considerando o valor médio da função objetivo, o *gap* médio em relação à solução reportada na literatura por Rosa et al. (2017), bem como o tempo médio de execução. O *gap* foi calculado conforme a Equação 4.1, seguindo a mesma metodologia de comparação adotada pelos autores.

Tabela 6 – Resultados obtidos pelas estratégias de **BL-FI** e **BL-BI**.

Jobs	Literatura	Gerações	BL-FI			BL-BI		
			Resultado	Gap (%)	Tempo (s)	Resultado	Gap (%)	Tempo (s)
7	2.117	100	3.747	77.00	0.0010	3.747	77.00	0.0009
	2.117	1.000	3.747	77.00	0.0009	3.747	77.00	0.0011
	2.117	10.000	3.747	77.00	0.0015	3.747	77.00	0.0014
8	4.928	100	5.388	9.33	0.0017	5.388	9.33	0.0021
	4.928	1.000	5.388	9.33	0.0014	5.388	9.33	0.0017
	4.928	10.000	5.388	9.33	0.0062	5.388	9.33	0.0034
10	14.411	100	14.618	1.44	0.0056	14.618	1.44	0.0035
	14.411	1.000	15.622	8.40	0.0058	14.618	1.44	0.0033
	14.411	10.000	14.618	1.44	0.0062	14.618	1.44	0.0034
20	55.783	1.000	58.247	4.42	0.1361	56.212	0.77	0.0688
	55.783	1.000	55.783	0.00	0.1389	57.329	2.77	0.0715
	55.783	10.000	55.000	-1.40	0.1489	60.632	8.69	0.0850
30	43.673	100	57.021	30.56	0.5129	59.773	36.86	0.3356
	43.673	1.000	48.658	11.41	0.5266	54.806	25.49	0.3624
	43.673	10.000	54.188	24.08	0.5476	62.179	24.65	0.3896
50	118.220	100	140.648	18.97	10.2162	139.700	18.17	3.0189
	118.220	1.000	143.139	21.08	10.3109	144.839	22.52	3.0662
	118.220	10.000	138.952	17.54	10.0824	148.512	25.62	3.0368

Fonte: Elaborada pela autora.

Observa-se que ambas as estratégias apresentam tempos computacionais extremamente reduzidos nas instâncias de pequeno porte (7, 8 e 10 tarefas), com valores da ordem de milissegundos. À medida que o número de tarefas aumenta, o tempo cresce de forma consistente, sendo mais pronunciado na **BL-FI**, especialmente

para as instâncias de 30 e 50 tarefas.

Em relação à qualidade das soluções, para as instâncias com 7 tarefas, ambas as estratégias apresentaram desempenho idêntico em todos os cenários, com *gap* elevado (77%), indicando forte limitação da busca local isolada nesse porte. Para 8 tarefas, novamente não houve diferença entre as abordagens, mantendo-se um *gap* de 9,33%.

Na instância de 10 tarefas, a **BL-BI** mostrou maior estabilidade, mantendo *gap* de 1,44% em todos os cenários, enquanto a **BL-FI** apresentou piora significativa no caso de 1.000 gerações, alcançando *gap* de 8,40%.

Para 20 tarefas, observa-se maior variabilidade nos resultados. Em um dos cenários, a **BL-FI** atingiu *gap* igual a 0,00% e, em outro, apresentou valor negativo (-1,40%), indicando solução superior à referência adotada. Já a **BL-BI** apresentou melhor desempenho em um dos casos (*gap* de 0,77%), mas também resultados inferiores em outros cenários.

Nas instâncias de 30 tarefas, ambas as estratégias apresentaram *gaps* elevados, variando entre 11,41% e 36,86%. De modo geral, a **BL-FI** obteve resultados melhores que a **BL-BI** nesse porte, embora com maior tempo computacional.

Para 50 tarefas, os *gaps* permaneceram elevados para ambas as abordagens, variando entre 17,54% e 25,62%. A **BL-BI** apresentou tempos significativamente menores que a **BL-FI**, enquanto a qualidade das soluções foi semelhante ou levemente inferior em alguns cenários.

De forma geral, os resultados indicam que ambas as estratégias de busca local são computacionalmente eficientes, porém apresentam limitações na qualidade das soluções quando aplicadas de forma isolada, especialmente em instâncias de maior porte. Além disso, não há predominância absoluta entre **BL-FI** e **BL-BI**, sendo o desempenho dependente da instância analisada.

4.4.1 Comparação entre **AG** Clássico e **AG** Híbrido

A comparação entre o **AG** clássico (**AG**) e suas versões híbridas permite avaliar os impactos da incorporação de estratégias de busca local no desempenho do

método evolutivo, tomando como base os melhores valores conhecidos reportados na literatura por Rosa et al. (2017). Enquanto o AG clássico prioriza a exploração global do espaço de busca, os algoritmos híbridos combinam essa característica com mecanismos de intensificação, promovendo o refinamento das soluções ao longo do processo evolutivo.

As Tabelas 7 e 8 apresentam os resultados obtidos para o AG e o AG+BL-FI, considerando respectivamente a versão inicial e a versão final do algoritmo. De forma análoga, as Tabelas 9 e 10 apresentam a comparação entre o AG e o AG+BL-BI.

O termo “versão final” refere-se à configuração do algoritmo após ajustes realizados nos parâmetros e operadores ao longo de experimentos preliminares. Essa versão foi utilizada para a geração dos resultados apresentados, representando a melhor configuração obtida em termos de desempenho médio.

Tabela 7 – Comparação entre o AG e o AG+BL-FI.

Jobs	Literatura	Gerações	AG			AG+BL-FI		
			Resultado	Gap (%)	Tempo (s)	Resultado	Gap (%)	Tempo (s)
7	2.117	100	3.747	77.00	0.0468	3.747	77.00	0.0324
	2.117	1.000	3.747	77.00	0.0327	3.747	77.00	0.0321
	2.117	10.000	3.747	77.00	0.0269	3.747	77.00	0.0333
8	4.928	100	5.388	9.33	0.0380	5.388	9.33	0.0477
	4.928	1.000	5.388	9.33	0.0366	5.388	9.33	0.0496
	4.928	10.000	5.388	9.33	0.0350	5.388	9.33	0.0415
10	14.411	100	18.109	25.66	0.0440	14.618	1.44	0.0878
	14.411	1.000	18.780	30.32	0.0459	14.618	1.44	0.0912
	14.411	10.000	16.304	13.14	0.0483	14.618	1.44	0.0895
20	55.783	100	123.219	120.89	0.0817	54.538	-2.23	3.1598
	55.783	1.000	110.317	97.76	0.0974	54.538	-2.23	3.0876
	55.783	10.000	126.402	126.60	0.0717	54.538	-2.23	2.0322
30	43.673	100	229.861	426.32	0.1188	44.056	0.88	12.0849
	43.673	1.000	233.521	434.70	0.1076	44.647	2.23	30.4274
	43.673	10.000	251.134	475.03	0.1110	45.675	4.58	11.1333
50	118.220	100	865.105	631.78	0.1998	126.646	7.13	209.5897
	118.220	1.000	937.280	692.83	0.1881	127.794	8.10	1467.4425
	118.220	10.000	951.251	704.64	0.1835	127.557	7.90	195.2240

Fonte: Elaborada pela autora.

Tabela 8 – Comparação entre AG (AG) e o (AG+BL-FI), Versão Final.

Jobs	Literatura	Gerações	AG			AG+BL-FI		
			Resultado	Gap (%)	Tempo (s)	Resultado	Gap (%)	Tempo (s)
7	2.117	100	3.747	77.00	0.0261	3.747	77.00	0.0440
	2.117	1.000	3.747	77.00	0.0337	3.747	77.00	0.0495
	2.117	10.000	3.747	77.00	0.0478	3.747	77.00	0.0506
8	4.928	100	5.388	9.33	0.0323	5.388	9.33	0.0440
	4.928	1.000	5.388	9.33	0.0458	5.388	9.33	0.0594
	4.928	10.000	5.388	9.33	0.0629	5.388	9.33	0.0746
10	14.411	100	14.618	1.44	0.0472	14.618	1.44	0.1347
	14.411	1.000	14.618	1.44	0.0515	14.618	1.44	0.1434
	14.411	10.000	14.618	1.44	0.0480	14.618	1.44	0.1539
20	55.783	100	82.154	47.27	0.1084	54.538	-2.23	4.1327
	55.783	1.000	78.716	41.11	0.1092	54.538	-2.23	4.2141
	55.783	10.000	93.501	67.62	0.1205	54.538	-2.23	4.4511
30	43.673	100	173.721	297.78	0.1508	43.760	0.20	14.3453
	43.673	1.000	153.376	251.19	0.1760	43.760	0.20	15.5374
	43.673	10.000	154.965	254.83	0.2493	43.760	0.20	18.0925
50	118.220	100	682.605	477.40	0.2555	122.961	4.01	270.1283
	118.220	1.000	690.643	484.20	0.3081	123.084	4.11	265.1865
	118.220	10.000	629.110	432.15	0.3669	124.764	5.54	275.7434

Fonte: Elaborada pela autora.

Para as instâncias de pequeno porte (7 e 8 tarefas), observa-se que a incorporação da busca local do tipo *First Improvement* não produz alterações na qualidade das soluções, uma vez que tanto o AG quanto o AG+BL-FI apresentam resultados idênticos. Nessas situações, as diferenças limitam-se ao tempo computacional, que tende a ser ligeiramente superior nas versões híbridas devido ao custo adicional da intensificação.

Para a instância de 10 tarefas, na versão inicial do algoritmo, o AG clássico apresenta *gaps* elevados (acima de 13% e atingindo valores superiores a 30%), enquanto o AG+BL-FI reduz esse valor para aproximadamente 1,44% em todos os cenários, indicando ganho expressivo de qualidade. Já na versão final, o próprio AG passa a alcançar *gap* de 1,44%, reduzindo a diferença entre as abordagens.

Nas instâncias de 20 tarefas, os resultados sugerem superioridade da versão híbrida. Enquanto isso, o AG clássico apresenta *gaps* que variam entre aproximadamente 41% e 126% (versão final), o AG+BL-FI obtém *gap* negativo (-2,23%), indicando soluções superiores ao valor de referência adotado.

Para 30 tarefas, a diferença é ainda mais significativa. O AG clássico apresenta *gaps* extremamente elevados, ultrapassando 250% na versão final, enquanto o

AG+BL-FI mantém valores próximos de 0% a 4,5%, sugerindo forte capacidade de intensificação.

Em instâncias de 50 tarefas, o comportamento se mantém: o **AG** clássico apresenta *gaps* superiores a 400% na versão final, enquanto o **AG+BL-FI** reduz esse valor para a faixa entre aproximadamente 4% e 8%. Esse ganho ocorre à custa de aumento substancial no tempo computacional, que passa de frações de segundo para dezenas ou centenas de segundos.

Tabela 9 – Comparação entre o **AG** (**AG**) e o **AG** com Busca Local Best Improvement (**AG+BL-BI**).

Jobs	Literatura	Gerações	AG			AG+BL-BI		
			Resultado	Gap (%)	Tempo (s)	Resultado	Gap (%)	Tempo (s)
7	2.117	100	3.747	77.00	0.0468	3.747	77.00	0.0354
	2.117	1.000	3.747	77.00	0.0382	3.747	77.00	0.0332
	2.117	10.000	3.747	77.00	0.0320	3.747	77.00	0.0333
8	4.928	100	5.388	9.33	0.0380	5.388	9.33	0.0422
	4.928	1.000	5.388	9.33	0.0345	5.388	9.33	0.0435
	4.928	10.000	5.388	9.33	0.0399	5.388	9.33	0.0415
10	14.411	100	18.109	1.44	0.0440	14.618	1.44	0.0844
	14.411	1.000	18.780	1.44	0.0459	14.618	1.44	0.0912
	14.411	10.000	16.304	1.44	0.0483	14.618	1.44	0.0895
20	55.783	100	123.219	120.89	0.0817	54.538	-2.23	2.1234
	55.783	1.000	110.317	97.76	0.0974	54.538	-2.23	2.2974
	55.783	10.000	126.402	126.60	0.0717	54.538	-2.23	2.0322
30	43.673	100	229.861	426.32	0.1188	45.512	4.21	7.6743
	43.673	1.000	233.521	434.70	0.1076	44.056	0.88	9.0176
	43.673	10.000	251.134	475.03	0.1110	43.760	0.20	9.3158
50	118.220	100	865.105	631.78	0.1998	137.125	15.99	74.9596
	118.220	1.000	937.280	692.83	0.1881	133.485	12.91	86.9981
	118.220	10.000	951.251	704.64	0.1835	133.504	12.93	85.9095

Fonte: Elaborada pela autora.

Tabela 10 – Comparação entre o AG e o AG+BL-BI Versão final.

Jobs	Literatura	Gerações	AG			AG+BL-BI		
			Resultado	Gap (%)	Tempo (s)	Resultado	Gap (%)	Tempo (s)
7	2.117	100	3.747	77.00	0.0261	3.747	77.00	0.0437
	2.117	1.000	3.747	77.00	0.0337	3.747	77.00	0.0435
	2.117	10.000	3.747	77.00	0.0478	3.747	77.00	0.0498
8	4.928	100	5.388	9.33	0.0323	5.388	9.33	0.0585
	4.928	1.000	5.388	9.33	0.0458	5.388	9.33	0.0594
	4.928	10.000	5.388	9.33	0.0629	5.388	9.33	0.0576
10	14.411	100	14.618	1.44	0.0472	14.618	1.44	0.1272
	14.411	1.000	14.618	1.44	0.0515	14.618	1.44	0.1608
	14.411	10.000	14.618	1.44	0.0480	14.618	1.44	0.1556
20	55.783	100	82.154	47.27	0.1084	54.538	-2.23	3.0003
	55.783	1.000	78.716	41.11	0.1092	54.538	-2.23	3.0822
	55.783	10.000	93.501	67.62	0.1205	54.538	-2.23	3.3246
30	43.673	100	173.721	297.78	0.1508	43.760	0.20	13.8905
	43.673	1.000	153.376	251.19	0.1760	43.760	0.20	17.0415
	43.673	10.000	154.965	254.83	0.2493	43.760	0.20	17.4818
50	118.220	100	682.605	477.40	0.2555	125.928	6.52	181.3119
	118.220	1.000	690.643	484.20	0.3081	125.398	6.07	174.9630
	118.220	10.000	629.110	432.15	0.3669	124.117	4.99	166.0377

Fonte: Elaborada pela autora.

Resultados semelhantes são observados na comparação entre o AG e o AG+BL-BI.

Para instâncias pequenas (7 e 8 tarefas), não há diferença na qualidade das soluções, mantendo-se *gaps* idênticos entre as abordagens.

Na instância de 10 tarefas, a versão inicial mostra redução significativa do *gap* quando a busca local é incorporada. Na versão final, o AG já apresenta desempenho equivalente ao híbrido, ambos com *gap* de 1,44%.

Para 20 tarefas, o AG clássico apresenta *gaps* elevados (acima de 40%), enquanto o AG+BL-BI obtém *gap* negativo (-2,23%), sugerindo ganho qualitativo.

Nas instâncias de 30 tarefas, o AG clássico apresenta *gaps* superiores a 250% na versão final, enquanto o AG+BL-BI mantém valores próximos de 0,20%, sugerindo eficácia na redução da função objetivo.

Para 50 tarefas, observa-se novamente grande diferença entre as abordagens. O AG clássico apresenta *gaps* superiores a 400%, ao passo que o AG+BL-BI reduz esse valor para aproximadamente 5% a 7%. O tempo computacional aumenta de forma significativa, porém o ganho de qualidade é substancial.

Tabela 11 – Comparação entre o AG+BL-FI e AG+BL-BI.

Jobs	Literatura	Gerações	AG+BL-FI			AG+BL-BI		
			Resultado	Gap (%)	Tempo (s)	Resultado	Gap (%)	Tempo (s)
7	2.117	100	3.747	77.00	0.0324	3.747	77.00	0.0354
	2.117	1.000	3.747	77.00	0.0321	3.747	77.00	0.0332
	2.117	10.000	3.747	77.00	0.0300	3.747	77.00	0.0333
8	2.117	100	5.388	9.33	0.0477	5.388	9.33	0.0422
	2.117	1.000	5.388	9.33	0.0496	5.388	9.33	0.0435
	2.117	10.000	5.388	9.33	0.0363	5.388	9.33	0.0415
10	14.411	100	14.618	1.44	0.0878	14.618	1.44	0.0844
	14.411	1.000	14.618	1.44	0.0928	14.618	1.44	0.0912
	14.411	10.000	14.618	1.44	0.0888	14.618	1.44	0.0895
20	55.783	100	54.538	-2.23	3.1598	54.538	-2.23	2.1234
	55.783	1.000	54.538	-2.23	3.0876	54.538	-2.23	2.2974
	55.783	10.000	54.538	-2.23	3.2382	54.538	-2.23	2.0322
40	77.005	100	62.246	-19.62	307.8288	64.977	-16.10	30.4177
	77.005	1.000	60.295	-21.70	130.9362	62.308	-19.09	30.7728
	77.005	10.000	59.703	-22.47	103.9526	64.530	-16.20	31.7972
50	118.220	100	126.646	7.13	209.5897	137.125	15.99	74.9596
	118.220	1.000	127.794	8.10	1467.4425	133.485	12.91	86.9981
	118.220	10.000	127.557	7.90	195.2240	133.504	12.93	85.9095

Fonte: Elaborada pela autora.

Tabela 12 – Comparação entre o AG+BL-FI e AG+BL-BI Versão Final.

Jobs	Literatura	Gerações	AG+BL-FI			AG+BL-BI		
			Resultado	Gap (%)	Tempo (s)	Resultado	Gap (%)	Tempo (s)
0703	2.117	100	3.747	77.00	0.0382	3.747	77.00	0.0385
	2.117	1.000	3.747	77.00	0.0495	3.747	77.00	0.0435
	2.117	10.000	3.747	77.00	0.0506	3.747	77.00	0.0498
0801	2.117	100	5.388	9.33	0.0602	5.388	9.33	0.0585
	2.117	1.000	5.388	9.33	0.0594	5.388	9.33	0.0638
	2.117	10.000	5.388	9.33	0.0746	5.388	9.33	0.0689
1001	14.411	100	14.618	1.44	0.1347	14.618	1.44	0.1272
	14.411	1.000	14.618	1.44	0.1434	14.618	1.44	0.1608
	14.411	10.000	14.618	1.44	0.1539	14.618	1.44	0.1556
2011	55.783	100	54.538	-2.23	4.1327	54.538	-2.23	3.0003
	55.783	1.000	54.538	-2.23	4.2141	54.538	-2.23	3.0822
	55.783	10.000	54.538	-2.23	4.4511	54.538	-2.23	3.3246
4001	77.005	100	58.462	-24.08	97.3240	58.795	-23.65	68.9795
	77.005	1.000	58.462	-24.08	102.6942	58.372	-24.20	73.7969
	77.005	10.000	58.462	-24.08	111.4410	58.567	-23.94	74.5760
5001	118.220	100	122.961	4.01	270.1283	125.928	6.52	181.3119
	118.220	1.000	123.084	4.11	265.1865	125.398	6.07	174.9630
	118.220	10.000	124.764	5.54	275.7434	124.117	4.99	166.0377

Fonte: Elaborada pela autora.

De forma geral, os resultados indicam que a incorporação de busca local ao AG é determinante para a obtenção de soluções competitivas em instâncias de médio e grande porte. Enquanto o AG clássico apresenta forte limitação de

intensificação, resultando em *gaps* elevados à medida que o tamanho do problema cresce, as versões híbridas conseguem reduzir drasticamente essa lacuna.

Observa-se ainda que, na versão final do algoritmo, parte da melhoria também é atribuída às alterações estruturais implementadas, como elitismo, inicialização parcialmente heurística e mutação adaptativa, que aumentaram significativamente a eficiência do **AG** mesmo antes da aplicação da busca local.

4.4.2 Análise Estatística do Tempo de Execução

Esta seção apresenta a análise estatística do tempo de execução dos métodos avaliados para diferentes números de gerações. O objetivo é examinar o custo computacional associado a cada abordagem, bem como sua variabilidade, permitindo uma comparação mais detalhada entre o **AG** clássico, suas versões híbridas e as estratégias de busca local aplicadas isoladamente.

Foram consideradas as métricas de média, desvio padrão, valores mínimo e máximo do tempo de execução. A Tabela 13 apresenta o resumo estatístico obtido a partir das execuções experimentais.

Tabela 13 – Estatísticas do tempo de execução (em segundos) para os métodos avaliados.

Método	Gerações	Média	Desvio Padrão	Mínimo	Máximo
AG	100	0.1083	0.0517	0.0261	0.2555
AG	1.000	0.1310	0.0672	0.0337	0.3081
AG	10.000	0.1465	0.0741	0.0478	0.3669
AG+BL-BI	100	14.8508	40.8579	0.0437	181.3119
AG+BL-BI	1.000	14.9559	39.9061	0.0435	174.9630
AG+BL-BI	10.000	14.9147	38.1017	0.0498	166.0377
AG+BL-FI	100	20.7773	60.7558	0.0440	270.1283
AG+BL-FI	1.000	20.8778	60.0839	0.0495	265.1865
AG+BL-FI	10.000	22.1824	62.7516	0.0506	275.7434
BL-BI	100	0.2727	0.6809	0.0009	3.0189
BL-BI	1.000	0.2752	0.6886	0.0014	3.0662
BL-BI	10.000	0.2780	0.6822	0.0013	3.0368
BL-FI	100	0.7281	2.2532	0.0007	10.2162
BL-FI	1.000	0.7330	2.2716	0.0010	10.3109
BL-FI	10.000	0.7289	2.2239	0.0008	10.0824

Fonte: Elaborada pela autora.

Observa-se que o AG (AG) clássico apresenta tempos médios reduzidos, variando de 0,1083 s a 0,1465 s conforme o número de gerações aumenta de 100 para 10.000. O crescimento é discreto e aproximadamente linear, indicando boa escalabilidade. Além disso, os baixos valores de desvio padrão sugerem pequena variabilidade, caracterizando comportamento computacional estável e previsível. Contudo, conforme discutido anteriormente, essa eficiência ocorre em detrimento da qualidade das soluções obtidas.

As versões híbridas do AG apresentam aumento expressivo no custo computacional, decorrente da incorporação da busca local a cada geração. O AG+BL-BI (Best Improvement) apresenta tempos médios próximos de 15 segundos, com elevada variabilidade, refletida nos altos valores de desvio padrão e nos máximos observados. Já o AG+BL-FI (First Improvement) apresenta os maiores tempos médios, variando entre 20,77 s e 22,18 s, além da maior dispersão dos dados. Esse comportamento sugere maior esforço computacional associado à exploração mais intensiva da vizinhança na estratégia First Improvement.

Quando analisadas isoladamente, as estratégias de busca local (BL-BI e BL-FI) apresentam tempos médios consideravelmente inferiores aos métodos híbridos. Seus valores permanecem relativamente estáveis mesmo com o aumento do número de gerações, indicando que, de forma isolada, essas abordagens não sofrem impacto significativo da variação no número de gerações considerada neste experimento. No entanto, observa-se maior variabilidade relativa nos tempos máximos, especialmente para a estratégia BL-FI.

A Figura 8 complementa a análise estatística ao ilustrar a evolução do tempo médio de execução em função do número de gerações. A análise visual confirma que o AG clássico apresenta crescimento discreto e aproximadamente linear, enquanto os métodos híbridos mantêm patamares significativamente superiores de tempo de execução. Destaca-se que o AG+BL-FI é a abordagem mais custosa computacionalmente, ao passo que o AG+BL-BI configura-se como alternativa intermediária em termos de custo.

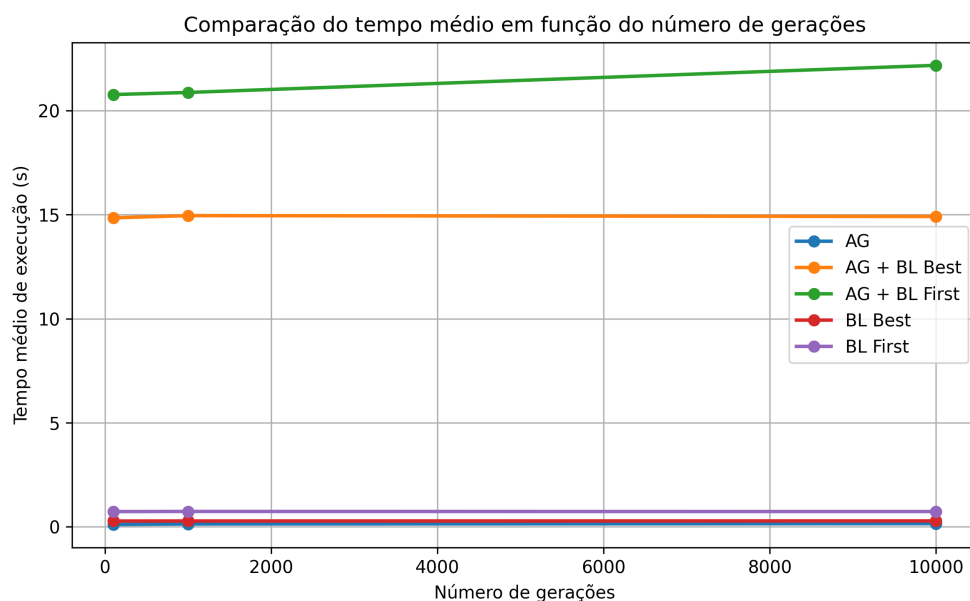


Figura 8 – Comparação do tempo médio de execução em função do número de gerações para os métodos avaliados.

Fonte: Elaborada pela autora.

De forma geral, os resultados sugerem relação entre qualidade da solução

e custo computacional. Métodos híbridos tendem a produzir soluções de melhor qualidade, conforme apresentado nas análises anteriores, porém exigem maior tempo de processamento. Em contrapartida, abordagens mais simples apresentam maior eficiência computacional e previsibilidade, embora com desempenho inferior em termos de qualidade das soluções.

4.5 Análise Gráfica e Estatística dos Resultados

Esta seção apresenta a análise comparativa dos métodos avaliados, considerando o comportamento de convergência, a qualidade das soluções obtidas — medida pelo *gap médio* (%) — e o tempo médio de execução. Foram analisados cinco métodos: **AG** (**AG**), Busca Local com estratégia Best Improvement (**BL-BI**), Busca Local com estratégia First Improvement (**BL-FI**) e as abordagens híbridas **AG+BL-BI** e **AG+BL-FI**.

4.5.1 Convergência em função do número de gerações

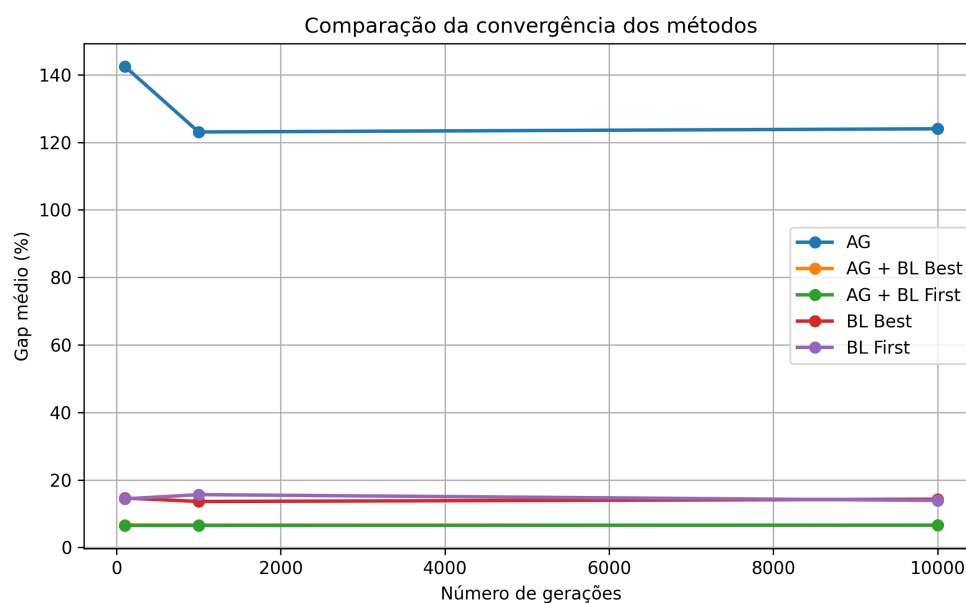


Figura 9 – Convergência dos métodos em função do número de gerações (Gap Médio \times Gerações).

Fonte: Elaborada pela autora.

A Figura 9 apresenta a evolução do *gap médio* em função do número de gerações para os métodos analisados. Observa-se que o **AG** (**AG**) apresenta os maiores valores de *gap* em todas as configurações testadas. Embora haja redução expressiva do *gap* ao passar de 100 para 1.000 gerações, o desempenho permanece significativamente inferior aos demais métodos, mantendo valores acima de 120%.

As abordagens híbridas apresentam desempenho substancialmente superior. O método **AG+BL-FI** mantém os menores valores de *gap*, próximos de 6–7%, sugerindo capacidade de refinamento das soluções. Já o **AG+BL-BI** apresenta desempenho intermediário, com *gap* em torno de 14–15%, mantendo estabilidade ao longo das gerações.

Quando analisadas isoladamente, as estratégias de busca local (**BL-BI** e **BL-FI**) apresentam desempenho semelhante entre si, com *gap* médio próximo de 14–16%. Observa-se pequena variação com o aumento do número de gerações,

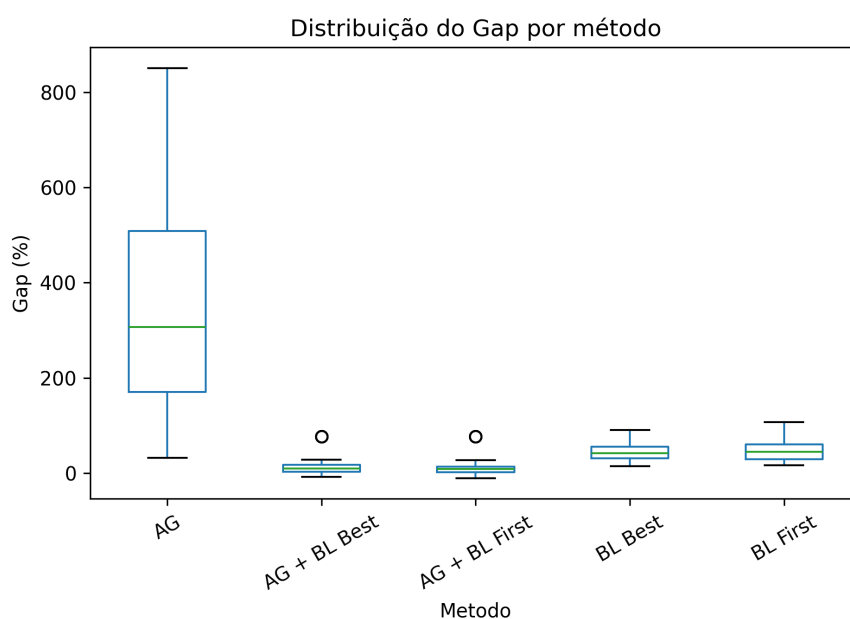
indicando que, nesse contexto experimental, o número de gerações não exerce influência relevante quando a busca local é aplicada de forma independente.

De maneira geral, os resultados sugerem que a hibridização do **AG** com estratégias de busca local proporciona ganhos expressivos de qualidade em relação ao **AG** puro. Destaca-se especialmente a combinação **AG+BL-FI**, que apresentou a melhor capacidade de convergência entre os métodos avaliados.

4.5.2 Distribuição do gap por método

A Figura 10 apresenta a distribuição do *gap* por método. O **AG** puro apresenta elevada variabilidade e presença de valores extremos, indicando instabilidade na qualidade das soluções. Em contrapartida, os métodos híbridos apresentam baixa dispersão e medianas menores, sugerindo robustez e consistência.

Figura 10 – Distribuição do gap (%) por método.

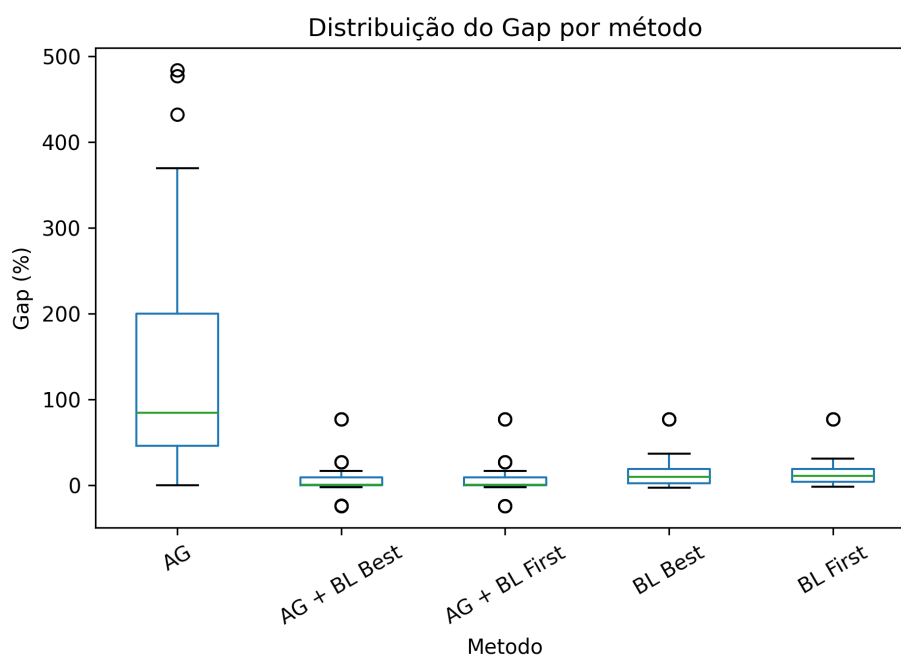


Fonte: Elaborada pela autora.

4.5.3 Comparação do gap médio agregado

A Figura 11 apresenta a distribuição do *gap* considerando todos os experimentos realizados, independentemente do número de gerações. O uso do boxplot permite analisar não apenas o valor médio, mas também a mediana, a dispersão dos dados e a presença de valores discrepantes (*outliers*).

Figura 11 – Distribuição do gap (%) por método considerando todas as gerações.



Fonte: Elaborada pela autora.

Observa-se que o AG (AG) apresenta a maior mediana de *gap* e a maior variabilidade entre todos os métodos. A amplitude interquartil é elevada e há presença de diversos *outliers* com valores bastante superiores aos demais métodos, sugerindo instabilidade e baixa qualidade das soluções encontradas.

As abordagens híbridas apresentam comportamento mais consistente. O método AG+BL-FI apresenta a menor mediana de *gap*, além de baixa dispersão relativa, indicando maior capacidade de convergência e estabilidade dos resultados. O AG+BL-BI também apresenta desempenho superior ao AG puro, com distribuição

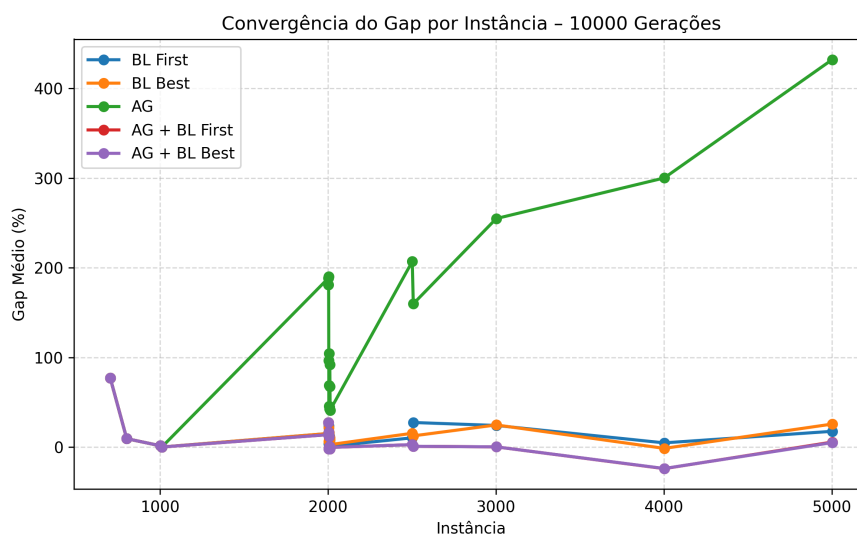
concentrada em valores reduzidos de *gap*, embora com variabilidade ligeiramente maior que a versão com First Improvement.

As buscas locais aplicadas isoladamente (BL-BI e BL-FI) apresentam desempenho intermediário. Embora apresentem menor variabilidade em comparação ao AG puro, seus valores medianos de *gap* são superiores aos obtidos pelas versões híbridas, indicando que a combinação entre exploração global (AG) e refinamento local é determinante para alcançar melhores resultados.

De forma geral, a análise agregada confirma que a hibridização do AG com estratégias de busca local proporciona ganhos consistentes tanto em qualidade quanto em estabilidade das soluções, destacando-se especialmente a combinação AG+BL-FI como a mais eficiente entre as abordagens avaliadas.

4.5.4 Impacto do Número de Gerações na Convergência por Instância

Figura 12 – Convergência do gap médio (%) por instância considerando 100 gerações.



Fonte: Elaborada pela autora.

A análise do impacto do número de gerações na convergência foi realizada considerando configurações com 100, 1000 e 10000 gerações. As Figuras 12, 13 e 14

apresentam o comportamento do *gap médio (%)* por instância para cada abordagem avaliada.

De forma geral, observa-se que os métodos híbridos apresentam convergência mais rápida e soluções de melhor qualidade em comparação aos métodos isolados. Esse comportamento evidencia o impacto positivo da intensificação proporcionada pela Busca Local, que permite explorar de forma mais eficiente regiões promissoras do espaço de busca, reduzindo o risco de estagnação em ótimos locais.

Na configuração com 100 gerações (Figura 12), observa-se que o AG isolado apresenta os maiores valores de *gap*, especialmente nas instâncias de maior porte, caracterizadas por um número mais elevado de tarefas (3000, 4000 e 5000). Esse resultado indica que, com um número reduzido de gerações, o algoritmo não dispõe de tempo suficiente para explorar adequadamente o espaço de busca, resultando em soluções de menor qualidade.

Por outro lado, as abordagens híbridas (AG + BL-FI e AG + BL-BI) apresentam desempenho superior ao AG puro, sugerindo que a intensificação promovida pela Busca Local contribui significativamente para a melhoria das soluções, mesmo em cenários com limitação no número de gerações.

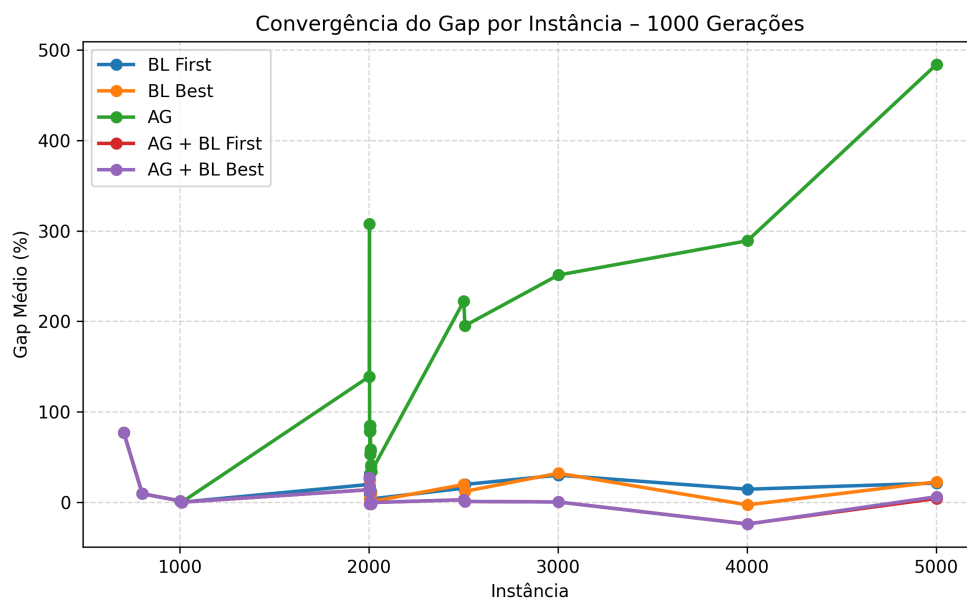


Figura 13 – Convergência do gap médio (%) por instância considerando 1000 gerações.

Fonte: Elaborada pela autora.

Ao aumentar para 1000 gerações (Figura 13), percebe-se uma leve melhoria no desempenho das abordagens híbridas, principalmente nas instâncias intermediárias e maiores. O AG isolado mantém comportamento semelhante ao observado com 100 gerações, apresentando *gaps* elevados à medida que o tamanho da instância cresce. As estratégias puramente baseadas em Busca Local mantêm estabilidade, uma vez que não dependem diretamente do processo evolutivo.

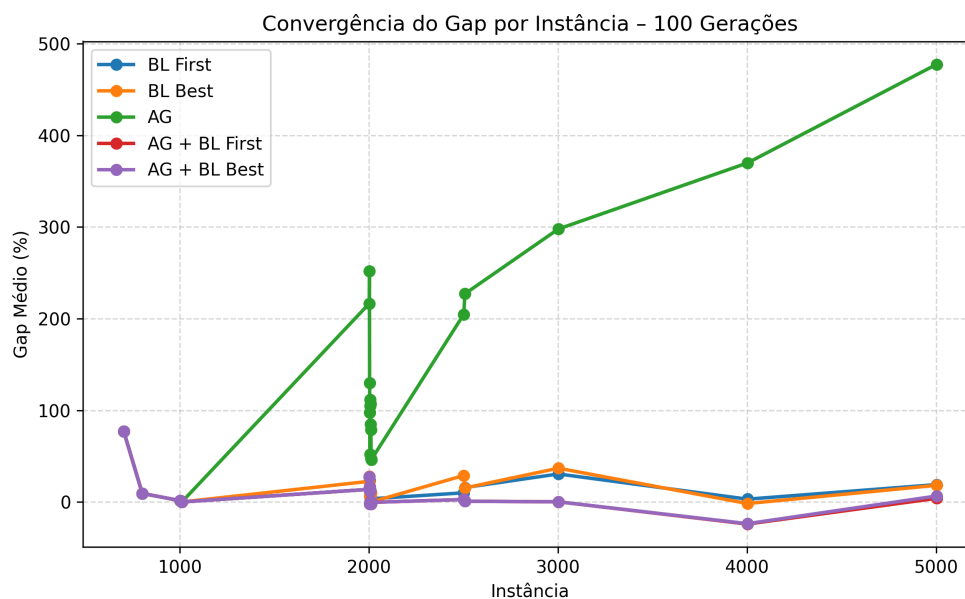


Figura 14 – Convergência do gap médio (%) por instância considerando 10000 gerações.

Fonte: Elaborada pela autora.

Por fim, na configuração com 10000 gerações (Figura 14), observa-se maior estabilidade e leve redução do *gap* nas abordagens híbridas, especialmente no **AG+ BL-BI**, que apresenta os melhores resultados nesse contexto experimental. Entretanto, o **AG** isolado continua exibindo crescimento acentuado do *gap* nas maiores instâncias, sugerindo que o aumento do número de gerações, por si só, não é suficiente para torná-lo competitivo.

De modo geral, considerando o contexto experimental estabelecido, os resultados sugerem que o impacto do aumento do número de gerações é mais expressivo nas abordagens híbridas, sobretudo em instâncias de maior porte. Nas instâncias menores (700 e 1000), as diferenças entre 100, 1000 e 10000 gerações são pouco expressivas, sugerindo que poucas gerações já são suficientes para alcançar soluções próximas do ótimo. Em contrapartida, para instâncias maiores, o maior esforço evolutivo contribui para melhor convergência quando associado à Busca Local.

4.5.5 Resumo comparativo dos resultados

Tabela 14 – Resumo comparativo dos métodos avaliados

Método	GAP Médio (%)	Tempo Médio (s)
AG+ BL-FI	6.54	21.28
AG+ BL-BI	6.60	14.91
BL-BI	14.17	0.28
BL-FI	14.64	0.73
AG	129.86	0.13

A Tabela 14 apresenta o desempenho médio dos métodos avaliados em termos de GAP percentual e tempo médio de execução. Observa-se que as versões híbridas do AG apresentaram os menores GAPs médios, com valores de 6,54% para o AG+ BL-FI e 6,60% para o AG+ BL-BI, em relação às abordagens baseadas exclusivamente em busca local.

Os métodos BL-BI e BL-FI apresentaram GAPs médios próximos de 14%, com desempenho inferior em termos de qualidade da solução quando comparados às versões híbridas. Entretanto, tais métodos mantiveram tempos computacionais substancialmente menores, indicando sua eficiência quando o critério prioritário é o custo computacional.

Por sua vez, o AG isolado apresentou o maior GAP médio (129,86%), indicando baixa capacidade de refinamento das soluções na ausência de mecanismos de intensificação. Além disso, verificou-se maior variabilidade nos resultados e pior escalabilidade nas instâncias médias e grandes, reforçando a limitação do modelo puramente evolutivo para o problema considerado.

Constata-se, portanto, um claro trade-off entre qualidade da solução e custo computacional. Enquanto os métodos híbridos reduzem significativamente o GAP médio, promovendo soluções mais próximas do melhor valor conhecido, tal ganho ocorre à custa de maior tempo de execução.

Dessa forma, considerando o contexto experimental estabelecido, conclui-se que a hibridização do AG com estratégias de busca local constitui a abordagem mais eficaz em termos de qualidade de solução. Por outro lado, as buscas locais

isoladas configuram alternativas adequadas em cenários nos quais há restrições severas de tempo computacional.

4.6 Considerações sobre os Resultados

Considerando o contexto experimental estabelecido, os resultados obtidos permitem uma análise comparativa consistente entre o AG clássico, as estratégias de Busca Local isoladas e as abordagens híbridas aplicadas ao problema PSUMAA. A avaliação considerou qualidade das soluções (gap médio), estabilidade dos resultados e tempo computacional.

O AG clássico apresentou tempos médios de execução reduzidos e comportamento estável entre as execuções. Entretanto, os valores de *gap* permaneceram superiores aos obtidos pelas abordagens híbridas, sobretudo em instâncias de maior porte, sugerindo limitações no processo de intensificação da busca quando utilizado isoladamente.

As estratégias de Busca Local, tanto na variante *First Improvement* quanto *Best Improvement*, sugerindo eficiência computacional e rápida convergência. Contudo, observou-se maior sensibilidade à escolha da solução inicial e tendência à estagnação em ótimos locais, impactando negativamente a qualidade final em instâncias mais complexas.

As abordagens híbridas apresentaram os melhores desempenhos em termos de qualidade média das soluções. A integração do AG com mecanismos de Busca Local permitiu reduzir consistentemente os valores de *gap*, além de apresentar menor variabilidade entre execuções. Esse comportamento confirma a eficácia da combinação entre exploração global (via operadores evolutivos) e intensificação local (via busca local), promovendo melhor equilíbrio entre diversidade populacional e refinamento das soluções.

Em termos de custo computacional, verificou-se aumento no tempo médio de execução nas estratégias híbridas, especialmente quando aplicada a intensificação mais aprofundada. Ainda assim, o ganho em qualidade das soluções mostrou-se proporcional ao esforço adicional, principalmente nas instâncias de médio e grande

porte.

De maneira geral, os experimentos sugerem um *trade-off* entre qualidade da solução e custo computacional. Métodos isolados oferecem maior rapidez e simplicidade, enquanto as abordagens híbridas indicam superioridade na obtenção de soluções próximas às melhores conhecidas, sendo mais indicadas para cenários em que a qualidade da solução é fator prioritário.

Os resultados detalhados por instância, incluindo valores médios, melhores soluções obtidas, tempo de execução e *gap*, encontram-se apresentados nos Apêndices, permitindo a rastreabilidade e a reprodutibilidade dos experimentos conduzidos.

5 Conclusão

Este trabalho apresentou o estudo, a implementação e a análise comparativa de meta-heurísticas aplicadas ao Problema de Sequenciamento de Produção em uma única máquina com penalidades por antecipação e atraso. Foram avaliadas estratégias de Busca Local (*First Improvement* e *Best Improvement*), um Algoritmo Genético clássico e versões híbridas do tipo memético, combinando AG com busca local.

Os resultados, considerando o contexto experimental estabelecido, sugerem diferenças entre as abordagens analisadas. A Busca Local isolada apresentou tempos de execução reduzidos e comportamento computacional estável, porém com limitação na qualidade das soluções em instâncias de maior porte, em função da convergência prematura para ótimos locais.

O Algoritmo Genético clássico indica maior capacidade de exploração do espaço de busca quando comparado à Busca Local isolada, produzindo soluções de melhor qualidade média. Entretanto, sua intensificação limitada impactou o desempenho em instâncias mais complexas, refletindo-se em valores de *gap* ainda elevados quando executado de forma isolada.

As abordagens híbridas (AG + BL First e AG + BL Best) apresentaram os melhores desempenhos globais, combinando a exploração populacional do AG com a intensificação promovida pela busca local. Observou-se redução consistente do *gap médio* e menor variabilidade entre execuções, principalmente nas instâncias de maior dimensão, onde o espaço de busca é significativamente mais amplo.

Em contrapartida, essa melhoria na qualidade das soluções implicou aumento no custo computacional, caracterizando um *trade-off* entre desempenho e tempo de processamento. De modo geral, a estratégia AG + BL Best apresentou equilíbrio entre qualidade e esforço computacional, enquanto a versão com *First Improvement* apresentou maior custo de execução.

Portanto, os resultados sugerem que a hibridização constitui uma estratégia

eficaz para problemas combinatórios complexos, especialmente quando se busca maior proximidade às melhores soluções conhecidas, ainda que ao custo de maior tempo de processamento.

5.1 Resultados e Contribuições

Nesta seção são consolidados os principais resultados obtidos a partir dos experimentos computacionais, bem como as contribuições teóricas e metodológicas decorrentes da implementação das meta-heurísticas propostas para o problema de sequenciamento em máquina única com penalidades de antecipação e atraso.

5.1.1 Análise dos Resultados Experimentais

A análise comparativa entre os métodos BL First, BL Best, AG, AG + BL First e AG + BL Best evidencia comportamentos distintos conforme a dimensão das instâncias avaliadas.

- **Desempenho da Busca Local:** As estratégias de Busca Local apresentaram excelente desempenho em termos de tempo de execução, principalmente nas instâncias de menor porte (até aproximadamente 1000 tarefas). Contudo, conforme o tamanho do problema aumenta, observa-se uma tendência clara de estagnação em ótimos locais. Essa limitação é refletida no aumento do Gap percentual em instâncias maiores, indicando baixa capacidade de diversificação do espaço de busca.
- **Eficácia do Algoritmo Genético:** O Algoritmo Genético isolado demonstrou maior capacidade exploratória quando comparado à Busca Local pura, produzindo soluções de melhor qualidade média nas instâncias intermediárias e grandes. Entretanto, a análise do desvio padrão evidencia variabilidade significativa entre execuções, sugerindo que o método, embora exploratório, apresenta dificuldade em intensificar a busca nas regiões mais promissoras do espaço de soluções.

- **Superioridade da Hibridização (Algoritmos Meméticos):** As versões híbridas (AG + BL First e AG + BL Best) apresentaram desempenho consistentemente superior. A incorporação da Busca Local como mecanismo de intensificação dentro do ciclo evolutivo permitiu reduzir substancialmente o Gap e, em diversas instâncias, alcançar o valor ótimo ou a melhor solução conhecida. Observou-se ainda redução significativa do desvio padrão, indicando maior estabilidade e robustez do método híbrido. Esse comportamento confirma que a combinação entre diversificação global (AG) e intensificação local (BL) constitui uma estratégia eficaz para problemas de alta complexidade combinatória.

5.1.2 Principais Contribuições

As principais contribuições deste Trabalho de Conclusão de Curso podem ser sintetizadas da seguinte forma:

1. **Proposição de Arquitetura Híbrida Modular:** Foi desenvolvida uma estrutura flexível que permite a integração de diferentes operadores de busca local ao Algoritmo Genético, possibilitando análises comparativas sistemáticas entre estratégias de *First Improvement* e *Best Improvement*.
2. **Avaliação Experimental em Instâncias de Grande Porte:** O estudo contemplou instâncias variando de 703 a 5001 tarefas, permitindo analisar o comportamento dos algoritmos em diferentes escalas. Os resultados evidenciam que a hibridização mantém elevada qualidade de solução mesmo em cenários de maior complexidade, embora com aumento proporcional do custo computacional.
3. **Análise Quantitativa do Trade-off Qualidade vs. Tempo:** O trabalho demonstrou empiricamente o equilíbrio entre qualidade de solução (Gap) e tempo de processamento, fornecendo evidências concretas sobre os limites práticos da intensificação em instâncias acima de 4000 tarefas.
4. **Aplicabilidade em Contextos Industriais:** A metodologia desenvolvida pode ser adaptada a sistemas reais de planejamento e controle da produção,

especialmente em ambientes *Just-in-Time*, nos quais a minimização simultânea de penalidades por antecipação e atraso é fundamental para a eficiência operacional.

5.2 Limitações do trabalho

Apesar dos resultados satisfatórios, algumas limitações foram identificadas:

- **Custo Computacional:** A aplicação da busca local em cada geração do AG elevou significativamente o tempo de execução nas instâncias de maior porte, podendo limitar sua aplicação em contextos que exigem respostas em tempo real.
- **Calibração de Parâmetros:** O desempenho do Algoritmo Genético mostrou-se sensível à escolha de parâmetros como tamanho da população, número de gerações e taxas de mutação e cruzamento. Uma calibração mais aprofundada poderia potencialmente melhorar os resultados.
- **Escopo do Modelo:** O estudo concentrou-se no ambiente de máquina única, não contemplando extensões mais complexas, como ambientes *flow-shop*, *job-shop* ou múltiplas máquinas paralelas.

5.3 Trabalhos Futuros

Como continuidade desta pesquisa, sugerem-se as seguintes direções:

- **Paralelização do Algoritmo Genético:** Implementar versões paralelas para reduzir o tempo de processamento das instâncias de grande porte.
- **Avaliação de Outras Meta-heurísticas:** Comparar os resultados com técnicas como *Simulated Annealing (SA)* e *Variable Neighborhood Search (VNS)*, ampliando a análise comparativa.

- **Exploração de Operadores Genéticos:** Investigar operadores de cruzamento e mutação específicos para problemas de permutação, visando melhorar a manutenção da diversidade populacional.
- **Extensão para Modelos Multiobjetivo:** Considerar simultaneamente diferentes critérios de desempenho, como minimização do número de tarefas atrasadas e minimização do custo total de penalidades.

Referências

ABUALIGAH, L. et al. Revolutionizing sustainable supply chain management: A review of metaheuristics. *Engineering Applications of Artificial Intelligence*, v. 126, p. 106839, 2023. ISSN 0952-1976. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0952197623010230>>. Citado na página 13.

BAKER, K. R. *Introduction to Sequencing and Scheduling*. New York, NY: John Wiley & Sons, 1974. Citado na página 20.

BUSTAMANTE, L. de M. *Minimização do custo de antecipação e atraso para o problema de sequenciamento de uma máquina com tempo de preparação dependente da sequência: aplicação em uma usina siderúrgica*. Dissertação (Dissertação de Mestrado) — Universidade Federal de Minas Gerais, Belo Horizonte, MG, fev. 2007. Orientador: Prof. Dr. Carlos Roberto Venâncio de Carvalho. Citado 2 vezes nas páginas 37 e 39.

DU, D.-Z.; LEUNG, J. Y.-T. Minimizing total tardiness on one machine is np-hard. *Mathematics of Operations Research*, INFORMS, v. 15, n. 3, p. 483–495, 1990. Citado na página 14.

FELDMANN, M.; BISKUP, D. Single-machine scheduling for minimizing earliness and tardiness penalties by meta-heuristic approaches. *Computers & Industrial Engineering*, v. 44, n. 2, p. 307–323, 2003. ISSN 0360-8352. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S036083520200181X>>. Citado na página 12.

FERREIRA, A. da S. et al. Programação da produção e meta-heurística: Uma revisão de mapeamento sistemático. *Revista Gestão Industrial*, v. 14, n. 3, 2018. Disponível em: <<https://periodicos.utfpr.edu.br/revistagi/article/view/7140>>. Citado 2 vezes nas páginas 29 e 32.

FIDELIS, M.; OLIVEIRA, D. Abordagens heurísticas para problemas de sequenciamento com múltiplas restrições operacionais. *Revista Engenharia de Produção*, Brasil, v. 25, n. 4, 2017. Citado 2 vezes nas páginas 29 e 31.

FILHO, M. F. R. *GRASP e Busca Tabu aplicados a problemas de programação de tarefas em máquinas paralelas*. Tese (Tese (Doutorado em Engenharia de Sistemas)) — Universidade Estadual de Campinas, Campinas, 2007. Citado 2 vezes nas páginas 12 e 14.

GUANG, Y. Application of catastrophic adaptive genetic algorithm to reactive power optimization in power system with wind farm. In: *International Conference on Automatic Control and Artificial Intelligence (ACAI 2012)*. [S.l.: s.n.], 2012. p. 534–537. Citado na página 32.

GUZEK, A.; PIÓRO, M.; ŻOTKIEWICZ, M. Permutation-based genetic algorithms for scheduling problems. *Computers & Operations Research*, Elsevier, v. 51, p. 1–12, 2014. Citado na página 47.

JÚNIOR, A. de C. G. *Problema de Seqüenciamento em uma Máquina com Penalidades por Antecipação e Atraso: Modelagem e Resolução*. Dissertação (Dissertação de Mestrado) — Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil, março 2007. Disponível em: <https://repositorio.ufmg.br/bitstream/1843/NVEA-7B7M9F/1/aloisio_de_castro_gomes_dissertacao_.pdf>. Citado 8 vezes nas páginas 13, 20, 21, 22, 23, 25, 38 e 52.

MORAIS, M. de F.; ROCHA, R. P. da; CARVALHO, L. de. Problemas de programação da produção: Uma visão geral dos métodos de solução. In: *Anais do IX Encontro de Engenharia de Produção Agroindustrial – EEPA*. Campo Mourão, Brasil: UNESPAR, 2010. Acesso em: 20 nov. 2025. Disponível em: <https://www.fecilcam.br/anais/ix_eepa/data/uploads/3-pesquisa-operacional/3-01.pdf>. Citado 3 vezes nas páginas 28, 30 e 33.

OLIVEIRA, B. A. S.; SIMÃO, D. R. A. Problema de sequenciamento de máquinas: Abordagem de data de entrega comum. In: *VIII Seminário de Iniciação Científica do IFMG*. Ribeirão das Neves, MG, Brasil: [s.n.], 2019. Disponível em: <<https://www.ifmg.edu.br/sic/edicoes-anteriores/resumos-2019/problema-de-sequenciamento-de-maquinas-abordagem-de-data-de-entrega-comum.pdf>>. Citado na página 13.

PENNA, P. H. V. et al. Uma heurística híbrida para minimizar custos com antecipação e atraso do sequenciamento da produção em uma máquina. *Produção*, v. 22, n. 4, p. 766–777, setembro/dezembro 2012. Disponível em: <<https://www.scielo.br/j/prod/a/bMK8YchLbRbd3DyMX9phvck/?format=pdf&lang=pt>>. Citado 3 vezes nas páginas 12, 26 e 33.

PENNONE, M. D.; MENDONÇA, R. G. R. *Abordagem em scheduling para o problema de sequenciamento de obras de infraestrutura no setor público*. Tese (Trabalho de Conclusão de Curso (Graduação em Engenharia de Produção)) — Universidade Estadual de Campinas, Faculdade de Ciências Aplicadas, Limeira, Brasil, 2019. Orientadora: Profa. Dra. Priscila Cristina Berbert Rampazzo. Citado na página 26.

PEREIRA, F. J. B. Algoritmo genético. In: *Manual de Computação Evolutiva e Metaheurísticas*. Belo Horizonte: Editora UFMG, 2013. Citado na página 32.

PINEDO, M. L. *Scheduling: Theory, Algorithms, and Systems*. 4. ed. New York: Springer, 2012. Citado 3 vezes nas páginas 19, 20 e 37.

PINTO, A. R. F.; MARTARELLI, N. J.; NAGANO, M. S. Algoritmo genético: Principais gaps, trade-offs e perspectivas para futuras pesquisas. *Trends in Computational and Applied Mathematics*, Sociedade Brasileira de Matemática Aplicada e Computacional, v. 23, n. 3, p. 413–438, 2022. Disponível em: <<https://www.scielo.br/j/tcam/a/DcLsnWLxJSQBwQrfm3sh6Cp/?format=pdf&lang=pt>>. Citado na página 18.

RIBEIRO, F. F.; SOUZA, M. J. F.; SOUZA, S. R. de. Uma metodologia adaptativa baseada em algoritmos genéticos para a resolução do problema de sequenciamento em uma máquina com penalização por antecipação e atraso da produção. In: *XXIX Encontro Nacional de Engenharia de Produção (ENEGEP)*. Salvador, BA, Brasil: [s.n.], 2009. Disponível em: <https://www.abepro.org.br/biblioteca/enegep2009_TN_STO_091_616_13973.pdf>. Citado 2 vezes nas páginas 17 e 24.

RODRIGUES, G. S.; PIZZOLATO, N. D. Sequenciamento de um lote de peças em uma única máquina com datas de entrega e penalidades: um estudo de caso. In: *Anais do Simpósio de Pesquisa Operacional da Marinha (SPOLM)*. Blucher, 2014. Disponível em: <<https://pdf.blucher.com.br/marineengineeringproceedings/spolm2014/126523.pdf>>. Citado 3 vezes nas páginas 23, 38 e 39.

ROSA, B. *Algoritmos e metaheurísticas para problemas de sequenciamento com penalidades de antecipação e atraso*. Tese (Tese de Doutorado) — Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), Belo Horizonte, Brasil, 2019. Citado 4 vezes nas páginas 26, 29, 31 e 33.

ROSA, B. F. et al. Algorithms for job scheduling problems with distinct time windows and general earliness/tardiness penalties. *Computers & Operations Research*, v. 81, p. 203–215, 2017. ISSN 0305-0548. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S030505481630332X>>. Citado 3 vezes nas páginas 52, 55 e 57.

RUSSELL, S.; NORVIG, P. *Figura ilustrativa do algoritmo de busca descendente preso em um mínimo local*. 2007. Disponível em: <https://www.researchgate.net/figure/Figura-32-O-algoritmo-de-busca-descendente-fica-pres-a-um-vale-com-um-minimo-local_fig1_237297408>. Acesso em: 20 nov. 2025. Citado na página 30.

- SANTOS, L. R. dos et al. Aplicando algoritmos genéticos e algoritmos culturais no sequenciamento de máquina com penalidades por antecipação e atraso em uma indústria metalúrgica. In: *Simpósio Brasileiro de Pesquisa Operacional (SBPO)*. Gramado, RS, Brasil: [s.n.], 2010. Disponível em: <<https://www.din.uem.br/~ademir/sbpo/sbpo2010/pdf/72524.pdf>>. Citado na página 18.
- SILVA, J. R.; OLIVEIRA, L. H.; MORAIS, R. Meta-heurísticas para o problema de sequenciamento de lotes de tarefas em máquinas paralelas. In: *Anais do Simpósio Brasileiro de Pesquisa Operacional*. São Paulo: SOBRAPO, 2007. p. 1–12. Citado na página 28.
- SIMÕES, R.; PEREIRA, M. Meta-heurísticas híbridas aplicadas ao sequenciamento de tarefas com tempos de setup dependentes da sequência. *Gestão & Produção*, Brasil, v. 23, n. 1, p. 99–114, 2016. Citado 3 vezes nas páginas 29, 31 e 33.
- SOUSA, J. J. R. de. *Um algoritmo genético híbrido para otimização do escalonamento de tarefas independentes em máquinas heterogêneas*. Dissertação (Dissertação de Mestrado) — Universidade Federal de Uberlândia, Uberlândia, MG, Brasil, setembro 2022. Disponível em: <<https://repositorio.ufu.br/bitstream/123456789/36613/1/AlgoritmoGeneticoHibrido.pdf>>. Citado na página 18.
- SPRICIGO, M. C. *Aplicação de Algoritmos Genéticos para o Problema de Escalonamento de Tarefas em Sistemas de Manufatura com Controle Supervisório e Autômatos com Parâmetros*. Dissertação (Dissertação de Mestrado) — Universidade Estadual do Oeste do Paraná (Unioeste), Foz do Iguaçu, PR, Brasil, dezembro 2018. Disponível em: <https://tede.unioeste.br/bitstream/tede/4209/5/Mailla_Cristine_Spricigo_2018.pdf>. Citado na página 18.

6 Materiais elaborados pelo autor

Tabela 15 – Resultados Experimentais Detalhados - 100 Gerações

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
0703	30	BL First	3747.0	4432.47	1336.07	77.00	0.0007	0.0000	0.0019
0703	30	BL Best	3747.0	4119.03	876.87	77.00	0.0009	0.0000	0.0019
0703	30	AG	3747.0	3787.90	86.42	77.00	0.0261	0.0161	0.0394
0703	30	AG + BL First	3747.0	3747.00	0.00	77.00	0.0440	0.0382	0.0689
0703	30	AG + BL Best	3747.0	3747.00	0.00	77.00	0.0437	0.0385	0.0909
0801	30	BL First	5388.0	5962.07	1659.57	9.33	0.0014	0.0000	0.0023
0801	30	BL Best	5388.0	6235.67	1872.32	9.33	0.0015	0.0000	0.0030
0801	30	AG	5388.0	6475.83	1076.40	9.33	0.0323	0.0216	0.0579
0801	30	AG + BL First	5388.0	5388.00	0.00	9.33	0.0602	0.0520	0.0710
0801	30	AG + BL Best	5388.0	5388.00	0.00	9.33	0.0585	0.0537	0.0633
1001	30	BL First	14 618.0	17 771.27	595.56	1.44	0.0056	0.0030	0.0091
1001	30	BL Best	14 618.0	16 610.50	2087.78	1.44	0.0033	0.0017	0.0050
1001	30	AG	14 618.0	19 436.00	2687.40	1.44	0.0472	0.0287	0.0722
1001	30	AG + BL First	14 618.0	14 618.00	0.00	1.44	0.1347	0.1090	0.2204
1001	30	AG + BL Best	14 618.0	14 618.00	0.00	1.44	0.1272	0.1133	0.1514
1010	30	BL First	22 273.0	27 051.90	5665.99	0.00	0.0049	0.0020	0.0072
1010	30	BL Best	22 273.0	27 591.33	5761.31	0.00	0.0036	0.0020	0.0060
1010	30	AG	22 273.0	29 195.57	5441.73	0.00	0.0492	0.0271	0.0738
1010	30	AG + BL First	22 273.0	22 273.00	0.00	0.00	0.1341	0.1186	0.1680
1010	30	AG + BL Best	22 273.0	22 273.00	0.00	0.00	0.1341	0.1112	0.1886
2001	30	BL First	18 542.0	24 386.40	3141.27	13.80	0.0964	0.0576	0.1270
2001	30	BL Best	19 963.0	26 300.37	5968.27	22.52	0.0644	0.0434	0.0881
2001	30	AG	51 568.0	71 187.47	12 652.48	216.48	0.1077	0.0510	0.1196
2001	30	AG + BL First	18 542.0	18 724.73	318.38	13.80	2.5702	1.4581	4.6117
2001	30	AG + BL Best	18 542.0	18 676.03	409.39	13.80	2.4314	1.4633	3.8845
2002	30	BL First	25 657.0	39 556.63	8597.41	27.24	0.1406	0.0740	0.1930
2002	30	BL Best	25 737.0	39 875.53	10 352.57	27.64	0.0640	0.0459	0.0922
2002	30	AG	70 976.0	98 548.73	16 069.51	251.99	0.1086	0.0452	0.1221

Continua na próxima página...

Tabela 15 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
2002	30	AG + BL First	25 657.0	25 657.00	0.00	27.24	3.3525	2.4692	4.6265
2002	30	AG + BL Best	25 657.0	25 657.00	0.00	27.24	2.4949	1.7968	4.5097
2003	30	BL First	25 251.0	33 891.20	5793.11	23.40	0.1092	0.0621	0.1451
2003	30	BL Best	23 879.0	32 161.67	5698.31	16.70	0.0626	0.0372	0.0883
2003	30	AG	47 055.0	71 975.13	13 719.24	129.96	0.1003	0.0546	0.1227
2003	30	AG + BL First	23 879.0	24 370.17	669.27	16.70	3.2293	1.7924	5.6718
2003	30	AG + BL Best	23 879.0	24 065.77	537.29	16.70	2.4414	1.4624	3.8290
2004	30	BL First	34 462.0	45 574.70	7458.53	11.92	0.1053	0.0636	0.1494
2004	30	BL Best	32 813.0	40 202.47	5676.69	6.56	0.0666	0.0429	0.0904
2004	30	AG	60 764.0	87 385.73	17 682.30	97.34	0.1046	0.0519	0.1216
2004	30	AG + BL First	30 176.0	30 744.63	847.52	-2.00	3.3774	1.9778	6.1939
2004	30	AG + BL Best	30 176.0	30 911.57	944.31	-2.00	3.0590	1.6278	5.3141
2005	30	BL First	62 052.0	76 597.23	8345.17	4.89	0.0869	0.0490	0.1334
2005	30	BL Best	59 769.0	79 724.43	12 672.52	1.03	0.0617	0.0353	0.0846
2005	30	AG	90 025.0	125 847.50	15 152.38	52.18	0.1094	0.0489	0.1223
2005	30	AG + BL First	59 158.0	60 018.37	932.52	0.00	3.1019	1.9037	5.8471
2005	30	AG + BL Best	59 158.0	60 411.90	1581.94	0.00	2.6102	1.8036	4.8068
2006	30	BL First	35 204.0	41 839.80	5653.31	13.37	0.1315	0.0538	0.1800
2006	30	BL Best	35 204.0	47 800.10	8810.64	13.37	0.0660	0.0419	0.0943
2006	30	AG	65 787.0	104 499.80	19 630.30	111.85	0.1016	0.0529	0.1207
2006	30	AG + BL First	34 636.0	35 239.70	465.31	11.54	3.6129	2.4555	6.3198
2006	30	AG + BL Best	34 636.0	35 301.33	880.77	11.54	2.7267	1.5315	5.1306
2007	30	BL First	44 655.0	60 851.73	9194.49	10.02	0.0809	0.0277	0.1299
2007	30	BL Best	44 894.0	59 159.57	8204.97	10.61	0.0687	0.0355	0.1025
2007	30	AG	83 041.0	106 540.13	12 059.38	104.59	0.1068	0.0640	0.1314
2007	30	AG + BL First	40 438.0	42 388.10	1786.08	-0.37	3.3743	1.7342	5.0302
2007	30	AG + BL Best	40 438.0	43 277.50	1999.06	-0.37	2.6077	1.4801	4.9186
2008	30	BL First	29 622.0	38 499.33	6874.90	0.95	0.1263	0.0745	0.2468
2008	30	BL Best	30 674.0	38 141.80	6197.72	4.53	0.0651	0.0529	0.0885
2008	30	AG	60 741.0	85 853.73	17 054.38	107.00	0.0992	0.0497	0.1212
2008	30	AG + BL First	29 622.0	29 688.47	218.64	0.95	2.9061	1.7894	4.9624
2008	30	AG + BL Best	29 622.0	29 702.60	112.79	0.95	2.8973	1.6096	4.8321

Continua na próxima página...

Tabela 15 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
2009	30	BL First	75 952.0	85 509.03	6807.34	11.98	0.0997	0.0459	0.1392
2009	30	BL Best	73 625.0	97 103.20	11 468.63	8.55	0.0657	0.0398	0.0906
2009	30	AG	125 227.0	160 144.47	18 522.16	84.63	0.1033	0.0542	0.1271
2009	30	AG + BL First	67 827.0	69 793.13	2372.39	0.00	3.0117	1.4653	4.8212
2009	30	AG + BL Best	67 827.0	71 093.33	4043.95	0.00	2.6226	1.4249	4.6160
2010	30	BL First	37 756.0	47 346.37	6108.50	10.13	0.0723	0.0342	0.1214
2010	30	BL Best	37 612.0	48 932.17	6358.03	9.71	0.0726	0.0490	0.1012
2010	30	AG	61 268.0	86 608.13	16 428.33	78.71	0.1021	0.0529	0.1217
2010	30	AG + BL First	34 283.0	35 018.03	1037.35	0.00	3.3626	1.3418	5.3166
2010	30	AG + BL Best	34 283.0	35 143.07	1458.95	0.00	3.1011	1.6489	5.1654
2011	30	BL First	58 247.0	65 788.23	3147.81	4.42	0.1361	0.0721	0.2092
2011	30	BL Best	56 212.0	70 078.10	8039.94	0.77	0.0688	0.0425	0.1018
2011	30	AG	82 154.0	127 464.03	19 317.62	47.27	0.1084	0.0434	0.1226
2011	30	AG + BL First	54 538.0	54 600.00	250.03	-2.23	4.1327	2.4577	6.9940
2011	30	AG + BL Best	54 538.0	55 859.33	1508.63	-2.23	3.0003	1.8652	5.6096
2012	30	BL First	82 725.0	99 723.77	12 227.49	3.50	0.0904	0.0331	0.1827
2012	30	BL Best	79 931.0	100 978.67	11 993.02	0.00	0.0642	0.0430	0.1039
2012	30	AG	116 871.0	144 396.43	18 303.80	46.21	0.1070	0.0479	0.1250
2012	30	AG + BL First	79 599.0	81 048.67	1540.93	-0.42	2.8997	1.7496	4.5857
2012	30	AG + BL Best	79 599.0	80 814.87	1848.29	-0.42	3.0419	1.8840	4.6414
2501	30	BL First	25 916.0	39 493.13	9445.01	9.99	0.2455	0.1295	0.4104
2501	30	BL Best	30 372.0	41 489.33	8844.27	28.90	0.1643	0.1251	0.2289
2501	30	AG	71 802.0	109 888.17	20 162.89	204.72	0.1309	0.0854	0.1453
2501	30	AG + BL First	24 197.0	24 930.70	662.38	2.69	7.7903	4.6314	13.2154
2501	30	AG + BL Best	24 197.0	24 988.43	698.26	2.69	7.0504	4.1691	10.5382
2507	30	BL First	43 187.0	59 257.43	10 058.55	15.74	0.2380	0.1016	0.3458
2507	30	BL Best	43 015.0	57 869.00	9036.70	15.28	0.1854	0.1125	0.3067
2507	30	AG	122 049.0	160 694.43	27 094.65	227.10	0.1173	0.0482	0.1458
2507	30	AG + BL First	37 615.0	38 708.97	819.06	0.81	7.4307	3.8938	12.2923
2507	30	AG + BL Best	37 615.0	39 207.00	1572.91	0.81	7.2371	4.6865	12.1242
3001	30	BL First	57 021.0	80 339.30	14 682.89	30.56	0.5129	0.2986	0.7298
3001	30	BL Best	59 773.0	78 327.87	12 989.41	36.86	0.3356	0.2420	0.4596
3001	30	AG	173 721.0	230 619.07	29 788.12	297.78	0.1508	0.0633	0.1728

Continua na próxima página...

Tabela 15 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
3001	30	AG + BL First	43 760.0	47 298.23	3272.91	0.20	14.3453	8.0048	22.8680
3001	30	AG + BL Best	43 760.0	47 770.43	3470.60	0.20	13.8905	7.5201	23.0478
4001	30	BL First	79 375.0	129 952.10	27 632.55	3.08	2.7900	2.1034	3.4139
4001	30	BL Best	75 695.0	93 272.83	12 706.46	-1.70	1.2229	0.9530	1.4812
4001	30	AG	361 797.0	473 447.37	46 394.66	369.84	0.2054	0.1100	0.2293
4001	30	AG + BL First	58 462.0	62 840.30	3827.35	-24.08	97.3240	54.3429	141.4078
4001	30	AG + BL Best	58 795.0	62 219.63	2455.66	-23.65	68.9795	44.5905	83.5840
5001	30	BL First	140 648.0	171 364.50	19 472.34	18.97	10.2162	7.3458	12.6469
5001	30	BL Best	139 700.0	169 616.80	17 687.10	18.17	3.0189	2.4917	4.3863
5001	30	AG	682 605.0	866 743.07	91 876.77	477.40	0.2555	0.1293	0.2839
5001	30	AG + BL First	122 961.0	128 483.50	3239.20	4.01	270.1283	121.4883	412.5757
5001	30	AG + BL Best	125 928.0	132 886.80	3225.96	6.52	181.3119	132.7319	223.4417

Tabela 16 – Resultados Experimentais Detalhados - 1000 Gerações

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
0703	30	BL First	3747.0	4477.70	1231.55	77.00	0.0010	0.0000	0.0030
0703	30	BL Best	3747.0	3958.27	529.33	77.00	0.0014	0.0000	0.0030
0703	30	AG	3747.0	3796.03	84.70	77.00	0.0337	0.0197	0.0759
0703	30	AG + BL First	3747.0	3747.00	0.00	77.00	0.0495	0.0387	0.0912
0703	30	AG + BL Best	3747.0	3747.00	0.00	77.00	0.0435	0.0383	0.0575
0801	30	BL First	5388.0	5756.30	1266.48	9.33	0.0017	0.0010	0.0030
0801	30	BL Best	5388.0	6238.70	1404.11	9.33	0.0016	0.0005	0.0025
0801	30	AG	5388.0	6017.00	819.62	9.33	0.0458	0.0222	0.0968
0801	30	AG + BL First	5388.0	5388.00	0.00	9.33	0.0594	0.0529	0.0892
0801	30	AG + BL Best	5388.0	5388.00	0.00	9.33	0.0638	0.0566	0.0872
1001	30	BL First	14 618.0	17 808.00	1846.73	1.44	0.0054	0.0016	0.0100
1001	30	BL Best	14 618.0	17 322.30	2242.86	1.44	0.0033	0.0020	0.0052
1001	30	AG	14 618.0	20 006.37	2857.95	1.44	0.0515	0.0252	0.0812
1001	30	AG + BL First	14 618.0	14 618.00	0.00	1.44	0.1434	0.1117	0.2097
1001	30	AG + BL Best	14 618.0	14 618.00	0.00	1.44	0.1608	0.1158	0.2701
1010	30	BL First	22 273.0	27 013.37	5394.29	0.00	0.0050	0.0000	0.0076

Continua na próxima página...

Tabela 16 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
1010	30	BL Best	22 273.0	27 919.57	6268.11	0.00	0.0040	0.0023	0.0065
1010	30	AG	22 273.0	29 709.97	5651.33	0.00	0.0509	0.0233	0.1007
1010	30	AG + BL First	22 273.0	22 273.00	0.00	0.00	0.1407	0.1229	0.1649
1010	30	AG + BL Best	22 273.0	22 273.00	0.00	0.00	0.1422	0.1222	0.1737
2001	30	BL First	19 490.0	24 603.27	2126.68	19.61	0.1035	0.0615	0.1530
2001	30	BL Best	18 542.0	23 733.73	4103.08	13.80	0.0704	0.0492	0.1059
2001	30	AG	38 927.0	65 841.87	12 737.16	138.90	0.1329	0.0682	0.2174
2001	30	AG + BL First	18 542.0	18 681.13	308.59	13.80	2.9141	1.4729	5.3756
2001	30	AG + BL Best	18 542.0	18 677.63	364.24	13.80	2.4076	1.4841	4.1090
2002	30	BL First	25 657.0	36 351.70	7354.43	27.24	0.1427	0.0606	0.2001
2002	30	BL Best	25 737.0	37 499.60	8826.82	27.64	0.0719	0.0453	0.1017
2002	30	AG	82 168.0	108 237.13	18 235.92	307.50	0.1211	0.0529	0.2666
2002	30	AG + BL First	25 657.0	25 657.00	0.00	27.24	3.5113	2.6051	4.3215
2002	30	AG + BL Best	25 657.0	25 657.00	0.00	27.24	2.3258	1.7424	3.2017
2003	30	BL First	26 834.0	33 440.97	6146.18	31.14	0.1071	0.0518	0.1727
2003	30	BL Best	25 423.0	33 854.97	5319.24	24.24	0.0634	0.0448	0.0841
2003	30	AG	37 671.0	75 509.17	17 254.59	84.10	0.1255	0.0570	0.2144
2003	30	AG + BL First	23 879.0	24 420.00	681.95	16.70	3.3056	1.6070	5.0417
2003	30	AG + BL Best	23 879.0	24 396.60	778.45	16.70	2.5459	1.6590	3.8764
2004	30	BL First	33 374.0	46 403.07	7723.18	8.39	0.1026	0.0551	0.1445
2004	30	BL Best	31 966.0	42 874.50	6320.13	3.81	0.0663	0.0488	0.0996
2004	30	AG	54 969.0	86 748.07	19 255.31	78.52	0.1235	0.0553	0.1927
2004	30	AG + BL First	30 176.0	30 545.07	336.32	-2.00	3.0478	1.7206	4.8439
2004	30	AG + BL Best	30 176.0	30 755.90	926.73	-2.00	2.7871	1.6883	4.4817
2005	30	BL First	62 052.0	77 083.17	9685.91	4.89	0.0803	0.0449	0.1104
2005	30	BL Best	59 158.0	77 823.10	10 707.92	0.00	0.0642	0.0383	0.0882
2005	30	AG	90 453.0	122 224.93	16 668.99	52.90	0.1179	0.0603	0.2333
2005	30	AG + BL First	59 158.0	59 773.97	796.77	0.00	3.2297	1.9640	5.0152
2005	30	AG + BL Best	59 158.0	61 064.00	1693.70	0.00	2.8248	1.4681	5.7618
2006	30	BL First	35 204.0	42 847.03	5736.81	13.37	0.1332	0.0736	0.1844
2006	30	BL Best	34 636.0	51 450.07	9274.99	11.54	0.0619	0.0350	0.1011
2006	30	AG	55 489.0	93 745.03	21 325.84	78.69	0.1331	0.0544	0.2820
2006	30	AG + BL First	34 636.0	35 044.53	398.65	11.54	3.8943	2.5191	7.1499

Continua na próxima página...

Tabela 16 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
2006	30	AG + BL Best	34 636.0	35 428.17	769.43	11.54	2.8876	1.6165	4.9484
2007	30	BL First	46 101.0	59 701.10	10 313.71	13.58	0.0797	0.0255	0.1332
2007	30	BL Best	45 777.0	60 076.60	11 133.94	12.78	0.0635	0.0353	0.0888
2007	30	AG	74 927.0	108 810.93	16 883.85	84.60	0.1137	0.0526	0.2656
2007	30	AG + BL First	40 438.0	42 213.57	1799.94	-0.37	3.2143	1.8131	6.2134
2007	30	AG + BL Best	40 438.0	43 246.47	2813.64	-0.37	2.9976	1.5889	6.4636
2008	30	BL First	29 622.0	38 666.33	7722.81	0.95	0.1233	0.0841	0.1793
2008	30	BL Best	29 622.0	36 374.13	6094.73	0.95	0.0674	0.0373	0.0894
2008	30	AG	46 430.0	82 087.70	18 120.06	58.23	0.1207	0.0548	0.2008
2008	30	AG + BL First	29 622.0	29 861.97	442.70	0.95	3.3917	1.8530	6.9757
2008	30	AG + BL Best	29 622.0	29 729.73	223.18	0.95	2.7490	1.8551	5.2836
2009	30	BL First	72 071.0	87 214.77	8100.82	6.26	0.1067	0.0677	0.1689
2009	30	BL Best	72 504.0	97 197.90	14 224.84	6.90	0.0659	0.0469	0.1027
2009	30	AG	106 269.0	151 878.57	22 457.79	56.68	0.1192	0.0563	0.2770
2009	30	AG + BL First	67 827.0	69 716.93	2245.09	0.00	3.0429	1.8255	5.6198
2009	30	AG + BL Best	67 827.0	69 731.73	2724.36	0.00	2.8736	1.4288	5.9729
2010	30	BL First	38 125.0	48 288.60	8422.56	11.21	0.0724	0.0200	0.1023
2010	30	BL Best	37 612.0	48 697.90	4976.69	9.71	0.0716	0.0383	0.1018
2010	30	AG	47 886.0	80 354.83	21 933.90	39.68	0.1251	0.0475	0.2367
2010	30	AG + BL First	34 283.0	35 071.97	1061.06	0.00	2.9777	1.8335	4.3570
2010	30	AG + BL Best	34 283.0	35 154.00	1301.66	0.00	3.0994	1.6455	5.5838
2011	30	BL First	55 783.0	65 337.60	4860.97	0.00	0.1389	0.0491	0.2004
2011	30	BL Best	57 329.0	71 530.03	9385.36	2.77	0.0715	0.0459	0.0918
2011	30	AG	78 716.0	129 925.10	23 209.12	41.11	0.1092	0.0438	0.2290
2011	30	AG + BL First	54 538.0	54 677.90	384.01	-2.23	4.2141	2.5666	6.5193
2011	30	AG + BL Best	54 538.0	55 689.80	1277.81	-2.23	3.0822	1.8268	6.1542
2012	30	BL First	82 831.0	98 778.77	8353.66	3.63	0.0989	0.0539	0.1486
2012	30	BL Best	80 493.0	103 001.87	12 766.93	0.70	0.0637	0.0353	0.1010
2012	30	AG	106 995.0	143 388.07	17 342.43	33.86	0.1355	0.0528	0.2389
2012	30	AG + BL First	79 599.0	81 511.83	2275.62	-0.42	3.0628	1.7043	5.5338
2012	30	AG + BL Best	79 599.0	80 943.27	2015.60	-0.42	2.7753	1.5500	5.1549
2501	30	BL First	27 208.0	38 323.47	5862.26	15.47	0.2668	0.1981	0.3720

Continua na próxima página...

Tabela 16 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
2501	30	BL Best	28 165.0	43 003.93	8784.91	19.53	0.1642	0.1060	0.2425
2501	30	AG	75 935.0	103 030.37	18 859.21	222.26	0.1513	0.0777	0.3108
2501	30	AG + BL First	24 197.0	24 792.63	604.17	2.69	7.6062	3.9867	14.8219
2501	30	AG + BL Best	24 197.0	24 934.03	661.72	2.69	7.1230	4.2624	11.0841
2507	30	BL First	44 632.0	61 227.80	10 453.51	19.62	0.2242	0.1525	0.3293
2507	30	BL Best	41 788.0	58 411.13	10 813.08	11.99	0.1684	0.1184	0.2470
2507	30	AG	110 134.0	148 456.03	22 727.72	195.16	0.1715	0.0463	0.3426
2507	30	AG + BL First	37 615.0	39 410.33	1650.83	0.81	7.2105	4.3065	10.3767
2507	30	AG + BL Best	37 615.0	39 455.60	1859.84	0.81	7.3831	3.7548	12.9136
3001	30	BL First	56 701.0	77 267.77	14 300.22	29.83	0.5271	0.3549	0.6502
3001	30	BL Best	57 651.0	72 967.27	12 455.41	32.01	0.3764	0.2414	0.5800
3001	30	AG	153 376.0	221 936.73	35 958.90	251.19	0.1760	0.0621	0.4070
3001	30	AG + BL First	43 760.0	47 483.77	3562.63	0.20	15.5374	7.1152	29.7276
3001	30	AG + BL Best	43 760.0	46 907.27	3393.77	0.20	17.0415	9.1196	28.4401
4001	30	BL First	88 049.0	122 679.77	20 443.72	14.34	2.7610	1.9031	3.4238
4001	30	BL Best	74 693.0	97 359.07	14 099.41	-3.00	1.1925	0.9240	1.4280
4001	30	AG	299 634.0	432 394.90	71 581.73	289.11	0.2858	0.1175	0.6693
4001	30	AG + BL First	58 462.0	62 539.90	2934.84	-24.08	102.6942	50.1340	178.3932
4001	30	AG + BL Best	58 372.0	62 306.23	2349.42	-24.20	73.7969	30.1332	116.0484
5001	30	BL First	143 139.0	175 267.03	17 998.29	21.08	10.3109	8.1179	13.2907
5001	30	BL Best	144 839.0	167 745.47	11 769.19	22.52	3.0662	2.3010	3.8311
5001	30	AG	690 643.0	847 738.87	94 318.92	484.20	0.3081	0.1074	0.6021
5001	30	AG + BL First	123 084.0	129 957.83	3157.92	4.11	265.1865	136.3851	512.5937
5001	30	AG + BL Best	125 398.0	133 068.27	4361.35	6.07	174.9630	71.5142	354.3909

Tabela 17 – Resultados Experimentais Detalhados - 10000 Gerações

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
0703	30	BL First	3747.0	4447.63	1398.79	77.00	0.0008	0.0000	0.0029
0703	30	BL Best	3747.0	3797.47	54.50	77.00	0.0013	0.0000	0.0029
0703	30	AG	3747.0	3766.40	34.40	77.00	0.0478	0.0226	0.1032
0703	30	AG + BL First	3747.0	3747.00	0.00	77.00	0.0506	0.0411	0.0837

Continua na próxima página...

Tabela 17 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
0703	30	AG + BL Best	3747.0	3747.00	0.00	77.00	0.0498	0.0420	0.0664
0801	30	BL First	5388.0	5824.00	1496.90	9.33	0.0015	0.0000	0.0028
0801	30	BL Best	5388.0	6058.73	1555.62	9.33	0.0015	0.0010	0.0023
0801	30	AG	5388.0	5829.40	761.29	9.33	0.0629	0.0281	0.1122
0801	30	AG + BL First	5388.0	5388.00	0.00	9.33	0.0746	0.0532	0.1556
0801	30	AG + BL Best	5388.0	5388.00	0.00	9.33	0.0689	0.0576	0.0963
1001	30	BL First	14 618.0	18 447.73	1792.68	1.44	0.0067	0.0039	0.0107
1001	30	BL Best	14 618.0	16 677.90	2093.12	1.44	0.0038	0.0019	0.0061
1001	30	AG	14 618.0	21 228.10	3376.54	1.44	0.0480	0.0284	0.0866
1001	30	AG + BL First	14 618.0	14 618.00	0.00	1.44	0.1539	0.1164	0.2105
1001	30	AG + BL Best	14 618.0	14 618.00	0.00	1.44	0.1556	0.1129	0.2022
1010	30	BL First	22 273.0	28 417.70	5990.61	0.00	0.0060	0.0020	0.0094
1010	30	BL Best	22 273.0	27 592.30	5373.89	0.00	0.0048	0.0021	0.0088
1010	30	AG	22 273.0	30 252.57	6913.03	0.00	0.0656	0.0276	0.1571
1010	30	AG + BL First	22 273.0	22 273.00	0.00	0.00	0.1613	0.1266	0.2045
1010	30	AG + BL Best	22 273.0	22 273.00	0.00	0.00	0.1696	0.1421	0.3008
2001	30	BL First	18 754.0	23 999.83	3392.42	15.10	0.1078	0.0490	0.1745
2001	30	BL Best	18 754.0	24 858.93	5189.07	15.10	0.0934	0.0505	0.2049
2001	30	AG	46 970.0	63 599.33	11 991.29	188.27	0.1716	0.0829	0.4045
2001	30	AG + BL First	18 542.0	18 666.47	237.42	13.80	2.6710	1.5999	6.2051
2001	30	AG + BL Best	18 542.0	18 651.63	355.80	13.80	3.0788	1.7830	5.0017
2002	30	BL First	25 657.0	37 933.73	7723.08	27.24	0.1604	0.0928	0.2220
2002	30	BL Best	25 657.0	38 636.33	8923.72	27.24	0.0763	0.0545	0.1076
2002	30	AG	56 624.0	98 569.57	20 530.11	180.82	0.1490	0.0589	0.3172
2002	30	AG + BL First	25 657.0	25 657.00	0.00	27.24	3.9453	2.9803	5.0395
2002	30	AG + BL Best	25 657.0	25 657.00	0.00	27.24	2.4613	1.8386	4.1600
2003	30	BL First	25 251.0	32 507.93	3594.35	23.40	0.1136	0.0548	0.1756
2003	30	BL Best	24 869.0	33 272.40	5360.38	21.54	0.0648	0.0448	0.0938
2003	30	AG	59 329.0	81 398.13	16 484.38	189.95	0.1219	0.0593	0.2463
2003	30	AG + BL First	23 879.0	24 651.17	694.89	16.70	3.4785	1.9474	5.6465
2003	30	AG + BL Best	23 879.0	24 564.53	1004.91	16.70	2.5571	1.5532	4.0179
2004	30	BL First	31 878.0	43 729.33	6866.83	3.53	0.1091	0.0674	0.1877

Continua na próxima página...

Tabela 17 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
2004	30	BL Best	32 658.0	43 460.37	8167.86	6.06	0.0695	0.0438	0.0997
2004	30	AG	60 542.0	84 577.37	14 094.17	96.62	0.1354	0.0598	0.1973
2004	30	AG + BL First	30 176.0	30 595.37	707.39	-2.00	3.7440	2.2546	5.5599
2004	30	AG + BL Best	30 176.0	30 751.53	705.17	-2.00	3.4672	1.8941	7.8407
2005	30	BL First	59 986.0	75 050.90	8536.34	1.40	0.0944	0.0478	0.1322
2005	30	BL Best	64 191.0	82 638.27	10 239.33	8.51	0.0754	0.0351	0.2138
2005	30	AG	86 084.0	119 220.50	15 635.73	45.52	0.1581	0.0600	0.3018
2005	30	AG + BL First	59 158.0	60 068.83	1085.86	0.00	3.3162	1.9799	7.2387
2005	30	AG + BL Best	59 158.0	60 688.30	1441.26	0.00	2.8121	1.8913	5.3161
2006	30	BL First	35 204.0	43 361.93	5512.91	13.37	0.1309	0.0820	0.1799
2006	30	BL Best	36 087.0	49 582.47	8261.05	16.21	0.0629	0.0429	0.0947
2006	30	AG	63 501.0	97 165.53	21 048.29	104.49	0.1247	0.0585	0.2131
2006	30	AG + BL First	34 636.0	35 171.17	436.92	11.54	3.7716	2.5488	6.4609
2006	30	AG + BL Best	34 636.0	35 404.03	818.19	11.54	3.1496	1.8726	6.0067
2007	30	BL First	45 596.0	61 762.23	10 465.96	12.34	0.0810	0.0354	0.1436
2007	30	BL Best	44 372.0	61 134.40	9247.48	9.32	0.0649	0.0405	0.0907
2007	30	AG	68 524.0	107 291.77	18 720.03	68.83	0.1097	0.0459	0.2170
2007	30	AG + BL First	40 588.0	43 213.47	1870.74	0.00	3.1805	1.6628	5.3864
2007	30	AG + BL Best	40 438.0	42 876.27	2352.62	-0.37	3.2703	1.8401	5.8938
2008	30	BL First	30 788.0	37 543.93	5800.94	4.92	0.1312	0.0640	0.2392
2008	30	BL Best	30 098.0	37 679.13	5886.64	2.57	0.0688	0.0500	0.1088
2008	30	AG	41 830.0	79 289.97	15 896.16	42.55	0.1302	0.0651	0.2156
2008	30	AG + BL First	29 622.0	29 658.00	69.88	0.95	3.8710	2.0315	6.7737
2008	30	AG + BL Best	29 622.0	29 732.03	216.14	0.95	3.2730	1.8260	5.7580
2009	30	BL First	75 952.0	87 951.47	6924.48	11.98	0.1137	0.0661	0.1802
2009	30	BL Best	72 922.0	93 558.37	13 339.78	7.51	0.0738	0.0533	0.0994
2009	30	AG	98 246.0	156 509.70	25 641.62	44.85	0.1357	0.0556	0.2632
2009	30	AG + BL First	67 827.0	70 470.47	3544.19	0.00	3.1914	1.8066	5.3959
2009	30	AG + BL Best	67 827.0	69 777.07	2829.49	0.00	2.5852	1.6424	4.3584
2010	30	BL First	36 923.0	48 382.23	7203.33	7.70	0.0731	0.0421	0.1205
2010	30	BL Best	37 756.0	50 842.60	9811.66	10.13	0.0674	0.0390	0.1002
2010	30	AG	65 749.0	89 522.23	14 434.36	91.78	0.1306	0.0523	0.2598
2010	30	AG + BL First	34 283.0	35 076.53	1181.16	0.00	3.3130	1.6201	5.4863

Continua na próxima página...

Tabela 17 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
2010	30	AG + BL Best	34283.0	34842.53	966.73	0.00	3.7026	2.1002	6.4875
2011	30	BL First	55000.0	65258.50	6146.02	-1.40	0.1489	0.0760	0.2197
2011	30	BL Best	60632.0	73532.60	11319.42	8.69	0.0850	0.0414	0.1141
2011	30	AG	93501.0	131833.93	21879.01	67.62	0.1205	0.0528	0.2177
2011	30	AG + BL First	54538.0	54825.63	695.11	-2.23	4.4511	2.4436	6.8491
2011	30	AG + BL Best	54538.0	55669.60	1324.35	-2.23	3.3246	1.8250	5.8462
2012	30	BL First	80663.0	96080.73	8579.26	0.92	0.0932	0.0508	0.1610
2012	30	BL Best	82123.0	104371.40	12369.78	2.74	0.0671	0.0366	0.0997
2012	30	AG	112811.0	142297.30	15184.79	41.14	0.1490	0.0524	0.2585
2012	30	AG + BL First	79599.0	80427.50	968.55	-0.42	3.6398	2.0388	5.8790
2012	30	AG + BL Best	79599.0	81403.33	2255.62	-0.42	3.4666	1.4787	6.2336
2501	30	BL First	25983.0	40813.47	9699.58	10.27	0.2675	0.1750	0.4026
2501	30	BL Best	27182.0	41299.77	7989.02	15.36	0.1609	0.0896	0.2382
2501	30	AG	72332.0	108137.77	22100.11	206.97	0.1768	0.0894	0.3721
2501	30	AG + BL First	24197.0	24616.20	408.49	2.69	8.2750	5.1719	13.8377
2501	30	AG + BL Best	24197.0	24977.07	732.37	2.69	8.5256	4.2567	15.2343
2507	30	BL First	47525.0	62077.53	9620.64	27.37	0.2396	0.1696	0.3616
2507	30	BL Best	41966.0	57926.30	7987.43	12.47	0.1700	0.1060	0.2362
2507	30	AG	96915.0	141108.63	26308.39	159.74	0.1694	0.0627	0.3005
2507	30	AG + BL First	37615.0	39049.50	1425.09	0.81	9.2640	4.6194	16.2105
2507	30	AG + BL Best	37615.0	38859.07	1671.59	0.81	8.9960	5.1466	14.9406
3001	30	BL First	54188.0	82477.90	15179.73	24.08	0.5476	0.3367	0.7867
3001	30	BL Best	54439.0	78482.83	12131.21	24.65	0.3896	0.2556	0.5555
3001	30	AG	154965.0	209201.87	30685.69	254.83	0.2493	0.0878	0.7376
3001	30	AG + BL First	43760.0	46850.30	3322.39	0.20	18.0925	8.3279	35.1908
3001	30	AG + BL Best	43760.0	47818.80	3653.88	0.20	17.4818	8.8356	43.0765
4001	30	BL First	80586.0	118384.37	23406.26	4.65	2.7968	2.1138	3.7290
4001	30	BL Best	75860.0	98599.93	14416.64	-1.49	1.1994	0.8460	1.5822
4001	30	AG	308135.0	455026.27	68756.82	300.15	0.2537	0.0891	0.6496
4001	30	AG + BL First	58462.0	62372.30	2625.17	-24.08	111.4410	44.2957	193.4783
4001	30	AG + BL Best	58567.0	62802.37	2851.61	-23.94	74.5760	42.4629	166.0551
5001	30	BL First	138952.0	167353.73	20600.88	17.54	10.0824	5.3454	13.3385

Continua na próxima página...

Tabela 17 – Continuação

Inst.	Ex.	Método	Melhor	Média	Desvio	Gap (%)	T. Méd	T. Min	T. Max
5001	30	BL Best	148 512.0	171 004.30	14 675.43	25.62	3.0368	2.3480	4.0467
5001	30	AG	629 110.0	821 122.30	80 843.22	432.15	0.3669	0.1331	0.5494
5001	30	AG + BL First	124 764.0	129 286.13	3582.87	5.54	275.7434	123.9765	519.0022
5001	30	AG + BL Best	124 117.0	132 808.90	4019.02	4.99	166.0377	82.3376	288.6358