

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Um Algoritmo Genético de Chave Aleatória
Tendencioso para o Problema de
Agendamento de Tarefas em Máquina Única
com Tempos de Configuração Dependentes
de Sequência e Atrasos de Precedência**

Igor Corrêa Rodrigues

JUIZ DE FORA
JANEIRO, 2026

Um Algoritmo Genético de Chave Aleatória Tendencioso para o Problema de Agendamento de Tarefas em Máquina Única com Tempos de Configuração Dependentes de Sequência e Atrasos de Precedência

IGOR CORRÊA RODRIGUES

Universidade Federal de Juiz de Fora

Instituto de Ciências Exatas

Departamento de Ciência da Computação

Bacharelado em Ciência da Computação

Orientador: Luciana Brugiolo Gonçalves

Coorientador: Lorenza Leão Oliveira Moreno

JUIZ DE FORA

JANEIRO, 2026

UM ALGORITMO GENÉTICO DE CHAVE ALEATÓRIA
TENDENCIOSO PARA O PROBLEMA DE AGENDAMENTO DE
TAREFAS EM MÁQUINA ÚNICA COM TEMPOS DE
CONFIGURAÇÃO DEPENDENTES DE SEQUÊNCIA E
ATRASOS DE PRECEDÊNCIA

Igor Corrêa Rodrigues

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS
EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTE-
GRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE
BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Luciana Brugiolo Gonçalves
D.Sc. em Ciência da Computação

Lorenza Leão Oliveira Moreno
D.Sc. em Informática

Stênio Sã Rosário Furtado Soares
D.Sc. em Ciência da Computação

Marcos de Mendonça Passini
D.Sc. em Engenharia de Sistemas e Computação

JUIZ DE FORA

20 DE JANEIRO, 2026

Resumo

Este trabalho aborda o problema de agendamento em máquina única com tempos de configuração dependentes de sequência e atrasos de precedência, cujo objetivo é a minimização do *makespan*. Uma abordagem baseada no Algoritmo Genético de Chave Aleatória Tendencioso (*Biased Random Key Genetic Algorithm* – BRKGA) integrado a decodificadores especificamente desenvolvidos para o problema é investigada. São considerados dois decodificadores: um método simples de inserção ao final da sequência e um decodificador mais elaborado baseado em uma adaptação da heurística construtiva NEH, controlada por um parâmetro que define o número de posições avaliadas durante a inserção. Experimentos computacionais realizados em instâncias de diferentes portes demonstram que a abordagem proposta é competitiva com o algoritmo estado da arte da literatura, apresentando desempenho superior em instâncias médias e grandes. Os resultados evidenciam o papel central do decodificador na eficiência do BRKGA e indicam que a combinação entre busca populacional e heurísticas construtivas adaptadas constitui uma estratégia eficaz para o problema considerado.

Palavras-chave: SMSP, SDST, DP, BRKGA, NEH, Agendamento, Otimização, Heurísticas, Metaheurísticas, Algoritmos Genéticos, Algoritmos Construtivos.

Abstract

This work addresses the single-machine scheduling problem with sequence-dependent setup times and precedence delays, aiming to minimize the makespan. An approach based on the Biased Random Key Genetic Algorithm (BRKGA), integrated with decoders specifically developed for the problem, is investigated. Two decoders are considered: a simple method that inserts jobs at the end of the sequence, and a more elaborate decoder based on an adaptation of the constructive NEH heuristic, controlled by a parameter that defines the number of positions evaluated during insertion. Computational experiments conducted on instances of different sizes demonstrate that the proposed approach is competitive with the state-of-the-art algorithm in the literature, showing superior performance on medium- and large-sized instances. The results highlight the central role of the decoder in the efficiency of the BRKGA and indicate that the combination of population-based search and adapted constructive heuristics constitutes an effective strategy for the problem considered.

Keywords: SMSP, SDST, DP, BRKGA, NEH, Scheduling, Optimization, Heuristics, Metaheuristics, Genetic Algorithms, Constructive Algorithms.

Agradecimentos

Gostaria de expressar meus mais sinceros agradecimentos a todos que contribuíram para a realização deste trabalho. Primeiramente, agradeço aos meus pais pelo apoio, amor e confiança que sempre me ofereceram.

Às minhas orientadoras, expresso minha profunda gratidão pela orientação e paciência ao longo do desenvolvimento deste projeto.

Também agradeço aos professores da Universidade Federal de Juiz de Fora por todo o conhecimento e experiência compartilhados ao longo da minha trajetória acadêmica.

*“Aquilo que se faz por amor está sempre
além do bem e do mal”.*

Friedrich Nietzsche

Conteúdo

Lista de Figuras	8
Lista de Tabelas	9
Lista de Abreviações	10
1 Introdução	11
2 Problema de Agendamento de Tarefas em Máquina Única com Tempos de Configuração Dependentes de Sequência e Atrasos de Precedência	14
2.1 Problemas de agendamento em máquina única	15
2.2 Tempos de configuração dependentes da sequência	15
2.3 Atrasos de precedência	16
2.4 Problema de agendamento em máquina única com SDST e DP	16
2.4.1 Descrição formal do problema	18
2.5 Trabalhos relacionados	20
3 Fundamentação Teórica	23
3.1 Metaheurísticas	23
3.2 Algoritmos Genéticos	23
3.3 Algoritmo Genético de Chave Aleatória	24
3.4 Algoritmos construtivos	24
3.5 Algoritmo NEH	25
4 Abordagem Proposta	26
4.1 Processo evolutivo do BRKGA	26
4.1.1 Critério de parada	26
4.1.2 Avaliação da qualidade da população	27
4.1.3 Seleção e cruzamento	27
4.1.4 Mutação	27
4.1.5 Formação da nova população	28
4.2 Decodificadores	28
4.2.1 Decodificador simples	28
4.2.2 Decodificador NEH	29
4.3 Cálculo do <i>makespan</i>	31
5 Experimentos e resultados	34
5.1 Instâncias	34
5.2 Ambiente de testes	34
5.3 Configuração dos parâmetros	35
5.4 Resultados	36
5.4.1 Comparação entre decodificadores	36
5.4.2 Comparação com os resultados da literatura	39
6 Conclusão	43

7	Apêndice	46
7.1	Apêndice A: Resultados por instâncias	46
	Bibliografia	54

Lista de Figuras

4.1	Estrutura básica do BRKGA. Fonte: Imagem adaptada de (GONÇALVES; RESENDE, 2011).	29
5.1	Desempenho do decodificador para diferentes valores de k	37
5.2	Tempo $\times$ <i>makespan</i>	38
5.3	Gerações $\times$ Tempo	38

Lista de Tabelas

5.1	Parâmetros para o BRKGA	35
5.2	Resultados por combinação de parâmetros	40
7.1	Resultados para instâncias com 10 <i>jobs</i>	47
7.2	Resultados para instâncias com 15 <i>jobs</i>	48
7.3	Resultados para instâncias com 20 <i>jobs</i>	49
7.4	Resultados para instâncias com 25 <i>jobs</i>	50
7.5	Resultados para instâncias com 50 <i>jobs</i>	51
7.6	Resultados para instâncias com 75 <i>jobs</i>	52
7.7	Resultados para instâncias com 100 <i>jobs</i>	53

Lista de Abreviações

SMSP	<i>Single Machine Scheduling Problem</i>
SDST	<i>Sequence Dependent Setup Times</i>
DP	<i>Delayed Precedence</i>
BRKGA	<i>Biased Random Key Genetic Algorithm</i>
LIG	<i>Lean Iterated Greedy</i>
DPR	Desvio Percentual Relativo

1 Introdução

Problemas de agendamento desempenham um papel central em diversos contextos industriais e logísticos, estando diretamente associados à utilização eficiente de recursos, à redução de custos operacionais e ao cumprimento de prazos. Em ambientes produtivos reais, decisões relacionadas à ordem de processamento de tarefas influenciam significativamente o desempenho global do sistema, tornando o estudo de modelos de agendamento um tema recorrente na literatura de Pesquisa Operacional e Otimização Combinatória.

Neste contexto, o problema de agendamento em máquina única (*Single Machine Scheduling Problem* – SMSP) constitui uma classe fundamental, frequentemente utilizada como base para o estudo de variantes mais complexas. Apesar de sua aparente simplicidade, a introdução de características adicionais, como tempos de configuração dependentes da sequência (*Sequence Dependent Setup Times* – SDST) e restrições de atrasos de precedência (*Delayed Precedence* – DP), aumenta a complexidade do problema. A restrição de SDST ocorre quando o tempo necessário para preparar a máquina antes do processamento de uma tarefa depende diretamente da tarefa processada imediatamente antes. A consideração explícita dessa característica em decisões de planejamento da produção é fundamental em diversos ambientes industriais, pois pode levar à reduções significativas nos custos de preparação, eliminação de desperdícios e aumento da produtividade. Exemplos típicos incluem indústrias têxtil, gráfica, química, farmacêutica e metalúrgica, nas quais trocas de ferramentas, ajustes de parâmetros operacionais ou procedimentos de limpeza variam conforme a sequência de tarefas executadas (CHOOBINEH; MOHEBBI; KHOO, 2006; ALLAHVERDI, 2015).

As restrições de atraso de precedência impõem que determinadas tarefas só possam iniciar após um intervalo mínimo de tempo decorrido desde a conclusão de suas predecessoras. Esse atraso ocorre em situações nas quais um item precisa aguardar um período antes de ser processado novamente, como no resfriamento de peças, na secagem de tintas ou adesivos, ou em processos que exigem etapas separadas por restrições técnicas.

A presença simultânea de tempos de configuração dependentes da sequência e

restrições de atraso de precedência elimina diversas propriedades estruturais do problema clássico de máquina única, tornando a minimização do tempo total de execução na máquina (*makespan*) fortemente dependente da ordem de processamento das tarefas e das interações entre elas. Um dos primeiros trabalhos a abordar explicitamente essa combinação de restrições a partir de um problema prático é apresentado por (KUO; CHEN; YEH, 2020), que motiva o estudo do problema com base em um caso real observado em uma empresa de montagem de amplificadores em Taiwan. Nesse ambiente, o processo produtivo é dividido em várias etapas executadas em uma única máquina, exigindo tempos de configuração distintos conforme o tipo de produto ou componente a ser montado. Além disso, determinados componentes necessitam de um período de espera após a aplicação de adesivos especiais para garantir propriedades de impermeabilização, caracterizando atrasos de precedência entre etapas.

Outro trabalho de referência fundamental nesse contexto é o estudo de (LIN; YING, 2022), que investiga de forma sistemática o problema de agendamento em máquina única com SDST e DP. Os autores apresentam a formulação formal do problema, analisam suas propriedades estruturais, discutem sua complexidade computacional e propõem uma abordagem de referência (*Lean Iterated Greedy* – LIG), baseada no algoritmo *Iterated Greedy*, para a minimização do *makespan*. Além disso, o trabalho disponibiliza um conjunto de instâncias de teste, estabelecendo uma base comparativa para novas abordagens. O presente trabalho adota diretamente a modelagem do problema, as instâncias e os resultados de referência propostos por (LIN; YING, 2022), utilizando-os como principal fundamento para o desenvolvimento e a avaliação da abordagem do presente trabalho.

Embora a literatura relacionada explore diferentes estratégias heurísticas e de busca local para esse problema, observa-se a ausência de abordagens baseadas em algoritmos populacionais, em particular algoritmos genéticos. Nesse cenário, o Algoritmo Genético de Chave Aleatória Tendencioso (*Biased Random-Key Genetic Algorithm* – BRKGA) surge como uma alternativa promissora, uma vez que permite desacoplar o processo evolutivo da construção da solução por meio do uso de decodificadores específicos do problema. A representação por chaves aleatórias simplifica a manutenção da viabilidade das soluções ao longo das gerações e facilita a incorporação de conhecimento heurístico

no processo de decodificação.

Motivado por essas características, este trabalho propõe a aplicação de um BRKGA ao problema de agendamento em máquina única com tempos de configuração dependentes da sequência e restrições de precedência com atraso. Para isso, são desenvolvidos dois decodificadores distintos. O primeiro consiste em um decodificador simples, que ordena as tarefas (*jobs*) de acordo com suas chaves aleatórias e os insere sequencialmente ao final da solução parcial, servindo como abordagem de referência. O segundo decodificador é baseado em uma adaptação da heurística construtiva NEH, na qual cada *job* selecionado pode ser inserido em diferentes posições da solução parcial, respeitando as restrições do problema. Essa inserção é controlada por um parâmetro K , que define o número máximo de posições a serem avaliadas, permitindo analisar o impacto desse parâmetro na qualidade das soluções obtidas e no custo computacional do método.

Dessa forma, o objetivo é investigar o uso de algoritmos genéticos de chave aleatória como uma alternativa viável para o problema estudado, avaliando o desempenho de diferentes estratégias de decodificação e comparando os resultados obtidos com aqueles reportados na literatura. Os experimentos computacionais realizados permitem analisar o comportamento da abordagem proposta frente a diferentes configurações do parâmetro K e posicionar os resultados em relação às soluções de referência existentes.

Este trabalho está organizado da seguinte forma. No Capítulo 2 é apresentada a definição formal do problema de agendamento em máquina única com tempos de configuração dependentes da sequência e atrasos de precedência, incluindo a descrição das restrições e da função objetivo, bem como uma revisão dos trabalhos relacionados. O Capítulo 3 apresenta a fundamentação teórica que sustenta o desenvolvimento do trabalho. O Capítulo 4 descreve a metodologia proposta, detalhando o Algoritmo Genético de Chave Aleatória Tendencioso (BRKGA), os mecanismos evolutivos empregados e os decodificadores desenvolvidos, com ênfase na adaptação da heurística construtiva NEH. No Capítulo 5 são apresentados os experimentos computacionais, o ambiente de testes, a calibração dos parâmetros e a análise comparativa dos resultados obtidos em relação ao algoritmo de estado da arte da literatura. Por fim, o Capítulo 6 apresenta as conclusões do trabalho e discute possibilidades de extensões e trabalhos futuros.

2 Problema de Agendamento de Tarefas em Máquina Única com Tempos de Configuração Dependentes de Sequência e Atrasos de Precedência

Problemas de agendamento são amplamente estudados na área de pesquisa operacional devido à sua relevância prática em ambientes produtivos, logísticos e computacionais (PINEDO, 2016; BRUCKER, 1999). Um problema de agendamento consiste em determinar a ordem e os instantes de execução de um conjunto de tarefas, denominadas *jobs*, em um ou mais recursos limitados, como máquinas, de modo a otimizar um ou mais critérios de desempenho.

A importância desses problemas decorre do fato de que decisões de sequenciamento impactam diretamente indicadores como tempo total de processamento, nível de utilização dos recursos, cumprimento de prazos e custos operacionais. Como consequência, problemas de agendamento surgem em diversos contextos, incluindo sistemas de manufatura, processamento de dados, transporte e serviços.

Uma contribuição central para a sistematização da literatura é a notação proposta por (GRAHAM et al., 1979), conhecida como notação $\alpha|\beta|\gamma$, na qual α descreve o ambiente de máquinas, β representa as restrições e características do problema, e γ indica o critério de otimização. Essa notação permite classificar de forma concisa uma grande variedade de problemas e facilita a comparação entre diferentes variantes.

Do ponto de vista da complexidade computacional, muitos problemas de agendamento são NP-difíceis, especialmente quando incorporam restrições adicionais ou objetivos mais elaborados. Mesmo variantes aparentemente simples podem tornar-se intratáveis quando adicionam-se restrições como tempos de preparação, precedências ou múltiplos critérios de otimização, motivando o desenvolvimento de métodos heurísticos e metaheurísticos.

2.1 Problemas de agendamento em máquina única

O ambiente de máquina única ocupa um papel central na literatura de agendamento, sendo frequentemente utilizado como ponto de partida para o estudo de variantes mais complexas. Nesse ambiente, todos os *jobs* devem ser processados por uma única máquina, não sendo permitida sobreposição no processamento (PINEDO, 2016).

Apesar de sua simplicidade aparente, o problema de máquina única apresenta uma grande diversidade de variantes. Em sua forma mais básica, a minimização do *makespan* em uma configuração sem restrições pode ser resolvida trivialmente, uma vez que qualquer sequência produz o mesmo valor de *makespan*. No entanto, a introdução de objetivos alternativos, como a minimização do tempo médio de conclusão, ou de restrições adicionais, altera significativamente a estrutura do problema.

A relevância do ambiente de máquina única também se deve ao fato de que ele aparece como subproblema em contextos mais gerais, como *flow shops*, *job shops* e sistemas com múltiplas máquinas paralelas. Assim, avanços obtidos nesse ambiente frequentemente podem ser estendidos ou adaptados a configurações mais complexas.

2.2 Tempos de configuração dependentes da sequência

Tempos de configuração dependentes da sequência surgem quando o tempo necessário para preparar a máquina para processar um *job* depende do *job* processado imediatamente antes. Essa característica é comum em ambientes industriais nos quais há troca de ferramentas, ajustes finos ou limpeza de equipamento entre operações, como nas indústrias têxtil, química, farmacêutica e metalúrgica (ALLAHVERDI, 2015; CHOOBINEH; MOHEBBI; KHOO, 2006).

A presença de tempos de configuração dependentes da sequência elimina propriedades estruturais importantes do problema clássico de máquina única. Em particular, o custo associado a um *job* passa a depender de seu predecessor imediato, tornando o problema fortemente dependente da ordem de processamento. Essa característica aproxima o problema de agendamento de problemas clássicos de roteamento, como o Problema do Caixeiro Viajante assimétrico.

Diversos estudos demonstram que a introdução de tempos de configuração dependentes da sequência torna o problema NP-difícil, mesmo para critérios de otimização simples. Revisões abrangentes sobre esse tema podem ser encontradas em trabalhos como o de (ALLAHVERDI, 2015), que destacam tanto a relevância prática quanto os desafios computacionais associados a essa classe de problemas.

2.3 Atrasos de precedência

Restrições de atraso de precedência, também conhecidas como *precedence delays* ou *time-lags*, impõem que o início do processamento de um *job* ocorra apenas após um intervalo mínimo de tempo em relação à conclusão de outro *job*. Formalmente, para um par (i, j) sujeito a atraso de precedência, deve-se respeitar a condição $b_j \geq c_i + d_{ij}$.

Esse tipo de restrição é particularmente relevante em aplicações nas quais operações intermediárias exigem períodos de espera, como processos de resfriamento, secagem, cura de materiais ou estabilização química. Diferentemente das precedências clássicas, que apenas impõem uma ordem relativa entre tarefas, os atrasos de precedência introduzem uma dimensão temporal explícita no problema.

A literatura mostra que a inclusão de atrasos de precedência aumenta substancialmente a complexidade do problema, restringindo o conjunto de soluções viáveis e dificultando tanto a modelagem quanto o desenvolvimento de algoritmos eficientes. Trabalhos clássicos analisam subclasses polinomiais, enquanto estudos mais recentes propõem heurísticas e metaheurísticas para lidar com instâncias de maior porte.

2.4 Problema de Agendamento de Tarefas em Máquina Única com Tempos de Configuração Dependentes de Sequência e Atrasos de Precedência

O problema abordado neste trabalho consiste no agendamento de um conjunto finito de tarefas, ou *jobs*, em uma única máquina, com o objetivo de minimizar o tempo total de conclusão das tarefas, denominado *makespan*. Assume-se que todos os *jobs* estão

disponíveis para processamento no instante inicial e que a máquina pode processar, a qualquer momento, no máximo um *job*. Não é permitida a interrupção do processamento de um *job* uma vez iniciado, caracterizando um ambiente sem preempção.

Cada *job* possui um tempo de processamento conhecido e determinístico. Além disso, entre o processamento de dois *jobs* consecutivos, pode ser necessário um tempo de configuração (*setup*), cujo valor depende diretamente da ordem em que os *jobs* são executados. Esses tempos de configuração dependentes da sequência refletem a necessidade de ajustes, trocas de ferramentas ou preparações específicas da máquina, sendo aplicados sempre que um *job* sucede outro na sequência de processamento.

O problema também incorpora restrições de precedência com atraso, nas quais determinados pares de *jobs* estão relacionados por uma dependência temporal. Nessas situações, o início do processamento de um *job* sucessor só é permitido após a conclusão de seu predecessor e o cumprimento de um intervalo mínimo de tempo adicional. Esse atraso representa uma espera obrigatória entre a finalização de um *job* e o início de outro, decorrente de requisitos técnicos ou operacionais, e deve ser respeitado independentemente da posição relativa dos *jobs* na sequência de processamento.

A interação entre tempos de configuração dependentes da sequência e restrições de precedência com atraso torna a determinação do instante de início de cada *job* dependente de múltiplos fatores. Para cada *job*, o início do processamento é condicionado tanto pela conclusão do *job* processado imediatamente antes, considerando o respectivo tempo de configuração, quanto pelas restrições de precedência que possam existir, considerando os atrasos associados..

Uma solução viável para o problema é definida por uma permutação dos *jobs* que respeite todas as restrições de precedência e permita o cálculo consistente dos tempos de início e conclusão de cada tarefa. O *makespan* da solução corresponde ao instante de conclusão do último *job* na sequência, sendo este o critério adotado para avaliar a qualidade da solução. O objetivo do problema consiste, portanto, em determinar uma sequência viável que minimize o *makespan*, considerando simultaneamente os tempos de processamento, os tempos de configuração dependentes da sequência e as restrições de precedência com atraso.

A próxima seção apresenta a descrição formal do problema, incluindo a notação adotada, a função objetivo e as restrições que caracterizam matematicamente o modelo considerado neste trabalho.

2.4.1 Descrição formal do problema

Esta seção apresenta a formulação matemática do problema de agendamento em máquina única com tempos de configuração dependentes da sequência e restrições de atraso de precedência, conforme proposta por (LIN; YING, 2022). O problema é modelado por meio de programação linear inteira mista.

Na notação clássica de (GRAHAM et al., 1979) para problemas de agendamento, o problema é representado por

$$1 \mid s_{ij}, \text{ prec}(d_{ij}) \mid C_{\max},$$

onde 1 indica o ambiente de máquina única, s_{ij} representa os tempos de configuração dependentes da sequência e $\text{prec}(d_{ij})$ denota a presença de restrições de atraso de precedência. O critério de otimização adotado é a minimização do *makespan* ($C_{\max}$).

A seguir, são apresentados os índices, parâmetros, variáveis de decisão e a formulação matemática do problema.

Índices

- i, j : índices dos *jobs*, com $i \in \{0, 1, 2, \dots, n\}$ e $j \in \{0, 1, 2, \dots, n\}$, onde o índice 0 representa um *job* fictício (ou tarefa sentinela);
- A : conjunto de pares ordenados (i, j) , que representa o conjunto das restrições de atraso de precedência, ou seja, se $(i, j) \in A$, então o *job* j só pode iniciar após a conclusão do *job* i , respeitando um atraso mínimo d_{ij} .

Parâmetros

- n : número total de *jobs*;
- p_j : tempo de processamento do *job* j ;

- s_{ij} : tempo de configuração necessário para o processamento do *job* j imediatamente após o *job* i ;
- d_{ij} : tempo de atraso mínimo necessário para que o *job* j possa ser iniciado após a conclusão do *job* i ;
- M : constante positiva suficientemente grande.

Variáveis de decisão

- x_{ij} : variável binária que assume valor 1 se o *job* j é processado imediatamente após o *job* i , e 0 caso contrário;
- b_j : instante de início do processamento do *job* j ;
- c_i, c_j : instantes de conclusão do processamento dos *jobs* i e j ;
- $C_{\max}$: tempo total de execução na máquina (*makespan*).

Função objetivo

$$\min C_{\max} \quad (2.1)$$

Restrições

$$C_{\max} \geq c_j; \quad j = 1, 2, \dots, n, \quad (2.2)$$

$$\sum_{j=0}^n x_{ij} = 1; \quad i = 0, 1, \dots, n \ (i \neq j), \quad (2.3)$$

$$\sum_{i=0}^n x_{ij} = 1; \quad j = 0, 1, 2, \dots, n \ (i \neq j), \quad (2.4)$$

$$b_j \geq c_i + s_{ij} - M(1 - x_{ij}); \quad i = 1, 2, \dots, n; \ j = 1, 2, \dots, n \ (i \neq j), \quad (2.5)$$

$$b_j \geq c_i + d_{ij}; \quad \forall (i, j) \in A, \quad (2.6)$$

$$c_j = b_j + p_j; \quad j = 1, 2, \dots, n, \quad (2.7)$$

$$x_{ij} \in \{0, 1\}; \quad i = 0, 1, \dots, n; \ j = 1, 2, \dots, n \ (i \neq j). \quad (2.8)$$

A função objetivo 2.1 busca minimizar o *makespan* do cronograma, definido como o maior tempo de conclusão entre todos os *jobs*. A restrição 2.2 define o *makespan* como um limitante superior para os tempos de conclusão de todos os *jobs*. As restrições 2.3 e 2.4 garantem que cada *job* possua exatamente um sucessor e um predecessor na sequência de processamento, assegurando a construção de uma permutação válida dos *jobs* na máquina única. A restrição 2.5 estabelece a relação entre o instante de início de um *job* e o tempo de conclusão de seu predecessor imediato na sequência, incorporando explicitamente os tempos de configuração dependentes da sequência. A restrição 2.6 modela os atrasos de precedência, impondo que, para cada par (i, j) pertencente ao conjunto A , o início do processamento do *job* j ocorra somente após a conclusão do *job* i , acrescida de um atraso mínimo d_{ij} . A restrição 2.7 define o tempo de conclusão de cada *job* como a soma de seu tempo de início e de seu tempo de processamento. Por fim, a restrição 2.8 estabelece o domínio das variáveis de decisão x_{ij} , restringindo-as a valores binários.

2.5 Trabalhos relacionados

A literatura sobre problemas de agendamento em máquina única envolvendo tempos de configuração dependentes da sequência e restrições de precedência com atraso é relativamente restrita quando comparada a variantes clássicas do problema de agendamento. Ainda assim, diferentes linhas de pesquisa contribuíram de forma complementar para a compreensão da complexidade e das propriedades estruturais associadas a cada uma dessas restrições, preparando o terreno para estudos que tratam sua combinação de forma integrada.

No que se refere às restrições de precedência com atraso, diversos trabalhos investigaram o impacto dessas dependências temporais no problema de máquina única. (WIKUM; LLEWELLYN; NEMHAUSER, 1994) analisaram variantes do problema, identificando subclasses resolvíveis em tempo polinomial e demonstrando a dificuldade computacional do caso geral. Posteriormente, (BALAS; LENSTRA; VAZACOPOULOS, 1995) propuseram um algoritmo de *branch-and-bound* que explora limites inferiores derivados da estrutura do grafo de precedência, evidenciando o aumento significativo da complexidade quando atrasos são incorporados às restrições de precedência. Em uma linha mais

restrita, (FINTA; LIU, 1996) estudaram o caso em que os atrasos de precedência possuem comprimento unitário, apresentando um algoritmo ótimo com complexidade $O(n^2)$, o que reforça a importância da magnitude dos atrasos na dificuldade do problema.

De forma paralela, outro corpo de literatura concentrou-se na análise de tempos de configuração dependentes da sequência no contexto de máquina única. Esses trabalhos demonstram que a introdução de *setups* dependentes da ordem de processamento elimina propriedades estruturais importantes do problema clássico, tornando a avaliação do *makespan* fortemente dependente da sequência das tarefas. Estudos como (CHOOBINEH; MOHEBBI; KHOO, 2006) e (JULA; KONES, 2013) investigaram diferentes variações do problema, combinando tempos de setup dependentes da sequência com outros critérios de desempenho ou restrições adicionais, e destacaram a necessidade de abordagens heurísticas ou metaheurísticas para instâncias de maior porte.

Apesar desses avanços, a literatura que trata simultaneamente tempos de configuração dependentes da sequência e restrições de precedência com atraso é significativamente mais limitada. (KUO; CHEN; YEH, 2020) destacam explicitamente essa lacuna e propõem uma metaheurística baseada em *Variable Neighbourhood Search* (VNS) para a minimização do *makespan*. A abordagem explora diferentes estruturas de vizinhança e incorpora mecanismos específicos para garantir a viabilidade das soluções frente às restrições de precedência atrasadas, representando um dos primeiros esforços sistemáticos para lidar com a combinação dessas duas características.

Um marco fundamental nessa linha de pesquisa é o trabalho (LIN; YING, 2022), que constitui a principal referência adotada neste estudo. Os autores apresentam uma formulação abrangente do problema de agendamento em máquina única considerando simultaneamente tempos de configuração dependentes da sequência e restrições de precedência com atraso, discutindo suas propriedades estruturais e demonstrando sua complexidade computacional. Além disso, o trabalho propõe uma abordagem baseada no algoritmo *Iterated Greedy*, denominada *Lean Iterated Greedy* (LIG), e disponibiliza um conjunto de instâncias de teste. No presente trabalho, tanto a modelagem do problema quanto o conjunto de instâncias e os resultados de referência adotados seguem diretamente a formulação e os dados apresentados por (LIN; YING, 2022).

Mais recentemente, algoritmos genéticos de chave aleatória têm sido explorados como uma alternativa flexível para problemas de sequenciamento (LONDE et al., 2025), uma vez que permitem desacoplar o processo evolutivo da construção da solução e torna fácil manter a viabilidade entre as gerações usando uma codificação robusta.

Nesse cenário, o presente trabalho se diferencia da literatura existente ao integrar um BRKGA a um decodificador baseado em uma adaptação do algoritmo NEH, projetado especificamente para lidar com tempos de configuração dependentes da sequência e restrições de precedência com atraso. Diferentemente das abordagens baseadas exclusivamente em busca local, como as estratégias VNS propostas anteriormente, a abordagem adotada explora o potencial dos algoritmos genéticos de chave aleatória aliados a uma heurística construtiva eficiente, posicionando-se como uma alternativa complementar às estratégias existentes e diretamente alinhada à formulação estabelecida por (LIN; YING, 2022).

3 Fundamentação Teórica

Este capítulo apresenta os principais conceitos teóricos que fundamentam o desenvolvimento deste trabalho. São introduzidos os conceitos centrais de metaheurísticas e algoritmos genéticos, culminando na apresentação do Algoritmo Genético de Chave Aleatória Tendencioso (BRKGA). Por fim, são discutidos algoritmos construtivos e, em particular, a heurística NEH, que serve de base para o decodificador proposto neste trabalho.

3.1 Metaheurísticas

Metaheurísticas constituem um conjunto de estratégias gerais de busca projetadas para resolver problemas de otimização complexos de forma aproximada. Diferentemente de métodos exatos, essas abordagens não garantem a obtenção da solução ótima, mas são capazes de produzir soluções de alta qualidade em tempos computacionais reduzidos (TALBI, 2009).

Um aspecto central das metaheurísticas é o equilíbrio entre exploração do espaço de busca e intensificação em regiões promissoras. Estratégias como busca local, *simulated annealing*, *tabu search*, *variable neighbourhood search* e algoritmos populacionais exploram esse equilíbrio de diferentes maneiras.

Em problemas de agendamento, metaheurísticas são amplamente utilizadas devido à elevada complexidade combinatória envolvida. A flexibilidade dessas abordagens permite incorporar conhecimento específico do problema por meio de operadores especializados, heurísticas construtivas e mecanismos de reparo de soluções inviáveis.

3.2 Algoritmos Genéticos

Algoritmos Genéticos são metaheurísticas populacionais inspiradas na Teoria da Evolução Natural das espécies. Nessa abordagem, uma população de soluções candidatas evolui ao longo de gerações por meio de operadores de seleção, cruzamento e mutação (HOLLAND,

1992).

Cada solução é representada por um cromossomo, cuja estrutura depende do problema considerado. A qualidade das soluções é avaliada por meio de uma função de aptidão, que orienta o processo evolutivo. Normalmente, essas soluções são divididas em dois subconjuntos de acordo com a sua qualidade, formando os subconjuntos de soluções de elite e de não elite. Soluções com melhor desempenho possuem maior probabilidade de contribuir para a geração seguinte.

Embora algoritmos genéticos clássicos tenham sido aplicados com sucesso a diversos problemas, sua aplicação direta a problemas de sequenciamento pode ser dificultada pela necessidade de garantir a viabilidade das soluções. Isso motivou o desenvolvimento de representações indiretas, nas quais o cromossomo não codifica diretamente a solução, mas fornece informações para um procedimento de decodificação.

3.3 Algoritmo Genético de Chave Aleatória

O Algoritmo Genético de Chave Aleatória introduz uma representação indireta baseada em vetores de números reais no intervalo $[0, 1)$, denominados chaves aleatórias. Cada cromossomo consiste em um conjunto de chaves aleatórias, que são interpretadas por um decodificador específico do problema para construir uma solução viável (BEAN, 1994).

Essa abordagem separa o mecanismo evolutivo do problema sendo abordado, permitindo que o algoritmo genético opere em um espaço contínuo enquanto o decodificador incorpora conhecimento estrutural do problema. O Algoritmo Genético de Chave Aleatória Tendencioso estende essa ideia ao introduzir um processo de cruzamento enviesado, no qual um dos genitores é sempre selecionado do conjunto de elite e a probabilidade de herdar genes desse genitor é maior que a de herdar do genitor não elite.

3.4 Algoritmos construtivos

Algoritmos construtivos constroem soluções de forma incremental, adicionando elementos passo a passo segundo regras heurísticas (TALBI, 2009). Em problemas de agendamento, esses métodos são amplamente utilizados devido à sua simplicidade e eficiência computa-

cional.

Uma característica importante dos algoritmos construtivos é que decisões tomadas em etapas iniciais influenciam fortemente a qualidade da solução final. Por esse motivo, diversas heurísticas combinam procedimentos construtivos com mecanismos de refinamento local ou os utilizam como componentes internos de metaheurísticas.

3.5 Algoritmo NEH

O algoritmo NEH, proposto originalmente para o problema de *flow shop*, é uma das heurísticas construtivas mais conhecidas e eficazes na literatura de agendamento (NAWAZ; JR; HAM, 1983). O método constrói a sequência de forma incremental, inserindo os *jobs* em posições que minimizam o *makespan* parcial.

A eficiência do NEH decorre de sua estratégia de inserção sistemática, que avalia diferentes posições na sequência parcial e seleciona aquela que produz o melhor resultado. Essa característica torna o algoritmo particularmente atrativo para adaptações e extensões (RUIZ; MAROTO, 2005).

4 Abordagem Proposta

A abordagem proposta neste trabalho baseia-se na utilização do Algoritmo Genético de Chave Aleatória Tendencioso, proposto por (GONÇALVES; RESENDE, 2011), como metaheurística de busca global, aliada ao desenvolvimento de decodificadores especializados capazes de transformar cromossomos compostos por chaves aleatórias contínuas em sequências viáveis de *jobs*. A principal contribuição deste trabalho está no módulo de decodificação, responsável por incorporar as restrições estruturais do problema.

4.1 Processo evolutivo do BRKGA

No contexto do BRKGA, o processo evolutivo atua exclusivamente sobre vetores de chaves aleatórias, números reais no intervalo $[0, 1)$, enquanto a interpretação desses vetores como soluções factíveis do problema de agendamento é delegada ao decodificador. Essa separação permite que os operadores genéticos permaneçam independentes das características específicas do problema, ao mesmo tempo em que o conhecimento estrutural é explorado de forma eficiente durante a construção das soluções. De maneira geral, o algoritmo opera a partir da evolução de uma população inicial contendo exatamente p cromossomos gerados aleatoriamente, sendo cada cromossomo composto por n chaves aleatórias contínuas.

4.1.1 Critério de parada

O processo evolutivo é conduzido ao longo de um número finito de gerações, até que um critério de parada previamente definido seja satisfeito. A condição de parada adotada neste trabalho baseia-se em um número máximo de gerações, critério amplamente utilizado em aplicações do BRKGA.

4.1.2 Avaliação da qualidade da população

Em cada iteração do algoritmo, o decodificador é chamado para avaliar a qualidade de cada vetor de chaves aleatórias na população. Após esta etapa de avaliação, os indivíduos são ordenados de acordo com seus valores de aptidão, e a população é então particionada em dois subconjuntos distintos. O primeiro subconjunto corresponde ao conjunto de elite, formado por p_e indivíduos com os melhores valores de aptidão da geração corrente. O segundo subconjunto é composto pelos $p - p_e$ indivíduos restantes, denominados não elite.

4.1.3 Seleção e cruzamento

Tanto o algoritmo genético de chave aleatória de Bean quanto o BRKGA utilizam cruzamento uniforme parametrizado (SPEARS; JONG, 1995) para a geração dos descendentes. A principal diferença reside no fato de que, no BRKGA, um dos genitores é sempre selecionado a partir do conjunto de elite da população, enquanto o outro é escolhido no conjunto de não elite. Além disso, o cruzamento é enviesado de forma que a probabilidade de o descendente herdar a chave aleatória do genitor pertencente à elite é dada por $\rho_e > 0.5$.

Mais especificamente, para definir cada posição j do cromossomo descendente i , gera-se um número aleatório uniforme $r_{ij} \in [0, 1)$. Caso $r_{ij} \leq \rho_e$, o descendente herda a chave aleatória correspondente do genitor pertencente à elite; caso contrário, a chave é herdada do genitor não pertencente à elite. Esse viés sistemático impõe uma pressão seletiva controlada que favorece a propagação de características associadas às melhores soluções da população, acelerando o processo de convergência, ao mesmo tempo em que preserva diversidade genética suficiente para reduzir o risco de convergência prematura.

4.1.4 Mutação

Em vez de utilizar estratégias clássicas de mutação, o BRKGA emprega o conceito de imigração. Isto é, a cada geração, p_m membros da população são gerados aleatoriamente. Esse processo evita a convergência prematura da população, de forma semelhante a um operador de mutação, e conduz a uma formulação simples da convergência.

4.1.5 Formação da nova população

A cada iteração do BRKGA, uma nova população é construída. Cada nova geração é formada por:

- p_e indivíduos pertencentes ao conjunto de elite da geração anterior;
- p_m indivíduos imigrantes gerados aleatoriamente;
- $p - p_e - p_m$ indivíduos gerados através de cruzamento.

Esse ciclo de avaliação, particionamento da população e geração de novos indivíduos é repetido sucessivamente ao longo das gerações, até que o critério de parada seja satisfeito. Ao final do processo evolutivo, a melhor solução encontrada ao longo de todas as gerações é retornada como resultado do algoritmo. A Figura 4.1 ilustra todo este processo.

4.2 Decodificadores

Para a avaliação dos indivíduos, foram desenvolvidos dois decodificadores distintos para transformar os cromossomos em sequências de *jobs*. Nas subseções seguintes, Π representa a sequência e, $\Pi[j]$ representa o j -ésimo *job* da sequência.

4.2.1 Decodificador simples

O decodificador simples é uma adaptação do decodificador apresentado por (BEAN, 1994) para o SMSP com SDST e DP. A cada iteração, seleciona-se o *job* elegível com menor valor de chave aleatória. *Jobs* elegíveis são aqueles que não possuem restrições de precedência ou cujos *jobs* precedentes já foram inseridos na solução parcial. A cada inserção de um *job* na solução, a lista de *jobs* elegíveis é atualizada. Embora computacionalmente eficiente, essa estratégia oferece pouca flexibilidade para rearranjos locais da sequência, o que pode limitar a qualidade das soluções obtidas.

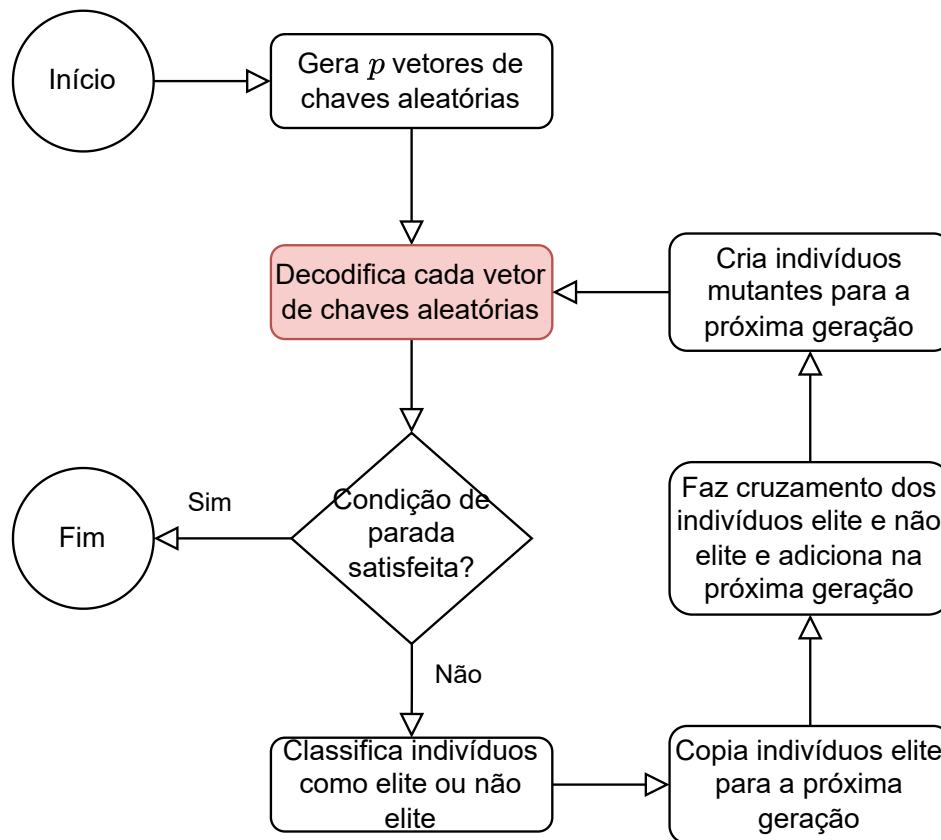


Figura 4.1: Estrutura básica do BRKGA. **Fonte:** Imagem adaptada de (GONÇALVES; RESENDE, 2011).

4.2.2 Decodificador NEH

O decodificador NEH é baseado em uma adaptação da heurística construtiva NEH. Assim como no decodificador simples, a cada iteração o *job* elegível é selecionado para inserção na sequência parcial. Uma vez selecionado o *job*, sua inserção é testada em múltiplas posições da sequência parcial, buscando aquela que minimize o *makespan* da solução parcial sem violar as restrições de precedência. Para controlar o custo computacional do decodificador, especialmente considerando que ele é executado para cada indivíduo da população em todas as gerações, a avaliação é restrita às últimas k posições possíveis da sequência. Essa janela limitada representa um compromisso entre qualidade da solução e tempo de execução: testar todas as posições possíveis exige mais tempo de execução, enquanto restringir a inserção a uma única posição reduziria significativamente a capacidade de melhoria local. O parâmetro k permite ajustar esse equilíbrio de forma controlada.

O Algoritmo 1 apresenta o pseudocódigo do decodificador baseado em NEH para a construção das sequências a partir dos cromossomos gerados pelo BRKGA.

Algoritmo 1: Decodificador baseado no algoritmo NEH

Entrada: Cromossomo r , conjunto de jobs J , janela k

```

1  $\Pi \leftarrow \emptyset$ 
2  $J_c \leftarrow \{j \in J : j \text{ não possui predecessores}\}$ 
3 while  $J_c \neq \emptyset$  do
4    $j \leftarrow \min_{i \in J_c} r_i$ 
5    $\Pi_{\text{best}} \leftarrow \Pi \cup \{j\}$ 
6    $lb \leftarrow \max(0, |\Pi| - k)$ 
7    $pos \leftarrow |\Pi| - 1$ 
8   while  $pos \geq lb$  do
9     if  $j$  viola precedências ao ser inserido na posição  $pos$  then
10      break
11    end if
12     $\Pi_{\text{new}} \leftarrow \text{insere}(j, \Pi, pos)$ 
13    if  $C_{\text{max}}(\Pi_{\text{new}}) < C_{\text{max}}(\Pi_{\text{best}})$  then
14       $\Pi_{\text{best}} \leftarrow \Pi_{\text{new}}$ 
15    end if
16     $pos \leftarrow pos - 1$ 
17  end while
18   $\Pi \leftarrow \Pi_{\text{best}};$ 
19  atualizar  $J_c$  removendo  $j$  e liberando novos jobs elegíveis;
20 end while
21 return  $\Pi$ ;
```

Inicialmente, a sequência parcial Π é inicializada como vazia, e o conjunto J_c é definido contendo apenas os *jobs* que não possuem predecessores, isto é, aqueles que podem ser escalonados no instante inicial (linhas 1–2). O algoritmo então executa um laço principal enquanto houver jobs elegíveis em J_c (linha 3). Em cada iteração, seleciona-se o *job* j cujo valor no cromossomo r é mínimo entre os *jobs* elegíveis (linha 4). Em seguida, uma solução candidata inicial Π_{best} é criada pela inserção do *job* j no final da sequência parcial Π (linha 5). Para limitar o esforço computacional do processo de inserção, define-se uma janela de busca de tamanho k , calculando o limite inferior lb para as posições testadas (linha 6). A variável pos é inicializada com a última posição válida da sequência (linha 7). O laço interno (linha 8) testa diferentes posições de inserção do *job* j dentro da janela definida. Caso a inserção de j em uma determinada posição viole alguma restrição de precedência, o laço é interrompido, uma vez que posições anteriores também seriam inviáveis (linhas 9–10). Para cada posição viável, uma nova sequência Π_{new} é construída

por meio da inserção de j na posição pos (linha 11). O *makespan* da nova sequência é então avaliado, e, se for inferior ao da melhor solução encontrada até o momento, a solução Π_{best} é atualizada (linhas 12–14). A posição de inserção é decrementada e o processo continua até que o limite inferior da janela seja atingido (linha 15).

Após a avaliação de todas as posições viáveis, a sequência parcial Π é atualizada com a melhor solução encontrada para a inserção do *job* j (linha 16). O conjunto J_c é então atualizado, removendo o *job* recém-inserido e incluindo novos *jobs* que se tornaram elegíveis em função do atendimento de suas restrições de precedência (linha 17).

O processo se repete até que todos os *jobs* tenham sido inseridos na sequência. Por fim, o algoritmo retorna a sequência completa Π , correspondente à decodificação do cromossomo r (linha 19).

É importante destacar que o decodificador simples pode ser interpretado como um caso particular do decodificador baseado em NEH, correspondente à configuração $k = 0$. Nessa situação, a inserção do *job* ocorre exclusivamente ao final da sequência parcial, eliminando a avaliação de posições alternativas. Essa relação estrutural entre os dois decodificadores permite uma comparação direta entre estratégias de complexidade distinta e fundamenta a análise experimental apresentada no capítulo seguinte.

4.3 Cálculo do *makespan*

Ambos os decodificadores necessitam calcular o *makespan* (C_{max}) para avaliar a aptidão da solução. O *makespan* é definido como o tempo de conclusão do último *job* da sequência. O tempo de conclusão de um *job* corresponde ao seu instante de início de processamento na máquina acrescido do tempo necessário para a execução de sua operação.

O tempo de início do primeiro *job* da sequência é igual a zero. Para os demais *jobs*, o instante de início do processamento do j -ésimo *job* é determinado pelo maior valor entre: (i) o tempo de conclusão do *job* imediatamente precedente na sequência, acrescido do tempo de *setup* correspondente; e (ii) o tempo de conclusão de cada *job* predecessor sujeito a atraso de precedência, isto é, todo *job* i tal que $(i, \Pi[j]) \in A$, acrescido do respectivo tempo de atraso.

A avaliação de cada inserção candidata exige o cálculo do *makespan* da sequência

resultante. Recalcular integralmente os tempos de conclusão de todas as tarefas a cada tentativa de inserção tornaria o processo computacionalmente inviável, sobretudo considerando o elevado número de avaliações realizadas ao longo do processo evolutivo. Por essa razão, o cálculo do *makespan* é realizado de forma incremental, atualizando apenas as porções da sequência afetadas pela inserção do novo *job* em uma determinada posição.

Esse procedimento considera simultaneamente os tempos de processamento, os tempos de configuração dependentes da sequência e os atrasos de precedência, garantindo a correta avaliação da solução sem a realização de cálculos desnecessários. O Algoritmo 2 descreve formalmente o procedimento adotado para o cálculo do *makespan*.

Algoritmo 2: Cálculo incremental do *makespan*

Entrada: Sequência Π , tempos p , setups s , atrasos d , índice x
Saída: $C_{\max}(\Pi)$

```

1 if  $x = 1$  then
2    $C[1] \leftarrow p(\Pi[1])$ 
3    $x \leftarrow x + 1$ 
4 end if
5 for  $j \leftarrow x$  to  $|\Pi|$  do
6    $b \leftarrow C[j - 1] + s(\Pi[j - 1], \Pi[j])$ 
7   for cada predecessor  $u$  de  $\Pi[j]$  do
8      $b \leftarrow \max(b, C[\text{pos}(u)] + d(u, \Pi[j]))$ 
9   end for
10   $C[j] \leftarrow b + p(\Pi[j])$ 
11 end for
12 return  $C[|\Pi|]$ 

```

Inicialmente, verifica-se se o índice inicial x é igual a 1 (linha 3). Nesse caso, o tempo de conclusão do primeiro *job* da sequência é calculado diretamente como o seu tempo de processamento, isto é, $C[1] = p(\Pi[1])$ (linha 4). Em seguida, o índice x é incrementado, indicando que os cálculos subsequentes devem partir do segundo *job* da sequência (linha 5).

O algoritmo então percorre a sequência Π a partir da posição x até o último *job* (linha 7). Para cada *job* $\Pi[j]$, é inicialmente calculado um tempo de início b , correspondente ao instante de conclusão do *job* anterior somado ao tempo de *setup* entre os *jobs* consecutivos $\Pi[j - 1]$ e $\Pi[j]$ (linha 8).

Em seguida, para cada predecessor u do *job* $\Pi[j]$, o valor de b é atualizado de forma a respeitar as restrições de precedência e os atrasos associados (linha 10). Mais es-

pecificamente, garante-se que o início do processamento de $\Pi[j]$ ocorra após a conclusão de cada predecessor u , acrescida do respectivo atraso $d(u, \Pi[j])$, tomando-se o valor máximo entre essas restrições (linha 11).

Após a determinação do instante mais cedo viável para o início do processamento do *job* $\Pi[j]$, o seu tempo de conclusão é calculado adicionando-se o tempo de processamento $p(\Pi[j])$, resultando em $C[j]$ (linha 13).

Esse procedimento é repetido até que todos os *jobs* da sequência tenham sido avaliados. Por fim, o algoritmo retorna o *makespan* da sequência Π , definido como o tempo de conclusão do último *job*, $C[|\Pi|]$ (linha 15).

Cabe ressaltar que tanto o BRKGA quanto os decodificadores desenvolvidos apresentam uma característica intrinsecamente paralelizável. Como a avaliação de cada cromossomo é independente, a decodificação de indivíduos distintos pode ser distribuída entre múltiplos núcleos de processamento sem necessidade de sincronização ou troca de informações. Essa propriedade permitiria acelerar significativamente o processo evolutivo em ambientes multiprocessados, especialmente para instâncias de grande porte. Contudo, a paralelização não foi explorada neste trabalho, uma vez que o foco está na proposição e análise da heurística de decodificação, e não na otimização da infraestrutura computacional. Todas as execuções foram realizadas em um único núcleo, de modo a isolar o impacto da técnica proposta sobre a qualidade das soluções.

Em síntese, a abordagem apresentada combina a capacidade exploratória do BRKGA com um decodificador eficiente e adaptado às características específicas do problema de agendamento estudado. A utilização de uma estratégia de inserção incremental inspirada no NEH, aliada a um cálculo otimizado do *makespan* e a mecanismos explícitos de respeito aos atrasos de precedência, resulta em um método robusto, flexível e adequado para a resolução do problema considerado.

5 Experimentos e resultados

Experimentos foram conduzidos com o objetivo de avaliar o desempenho do Algoritmo Genético de Chave Aleatória Tendencioso proposto para resolver o problema de agendamento em máquina única com tempos de configuração dependentes de sequência e atrasos de precedência em relação ao algoritmo LIG, estado da arte da literatura (LIN; YING, 2022).

5.1 Instâncias

Todas as instâncias utilizadas nos testes foram obtidas a partir do conjunto disponibilizado por (LIN; YING, 2022). Esse conjunto de testes possui cenários com números de tarefas (n) de 10, 15, 20, 25, 50, 75 e 100; tempos de processamento (p_j) uniformemente distribuídos nos intervalos $[10, 20]$ e $[20, 30]$; tempos de configuração (s_{ij}) uniformemente distribuídos nos intervalos $[5, 10]$ e $[10, 15]$; tempos de atraso d_{ij} uniformemente distribuídos nos intervalos $[20, 40]$ e $[40, 60]$. Para cada combinação destes parâmetros, existem 10 instâncias, totalizando 560 instâncias.

5.2 Ambiente de testes

Todos os testes para o BRKGA foram executados em uma máquina rodando o sistema operacional Ubuntu 24.04 com processador Intel Core i5-1135G7 @ 2.40GHz e 16 GB de memória. O algoritmo foi desenvolvido na linguagem C++ e adotou-se como base o *framework* de (TOSO; RESENDE, 2015), que fornece uma implementação modular, extensível e orientada a objetos do BRKGA na linguagem de programação C++. Esse *framework* organiza o algoritmo em três componentes fundamentais: (i) populações independentes evoluindo em paralelo; (ii) operadores básicos do BRKGA (cruzamento tendencioso, geração de mutantes e seleção de elite); e (iii) um decodificador externo responsável por mapear cromossomos para soluções viáveis do problema. Essa separação

entre mecanismo evolutivo e lógica de decodificação facilita a integração de heurísticas específicas, como a abordagem proposta neste trabalho. Todo o processo evolutivo, incluindo inicialização, seleção, cruzamento e mutação, utiliza as funções disponibilizadas pelo *framework*, enquanto o cálculo do *makespan* e a construção das sequências são realizados através de algoritmos desenvolvidos neste trabalho.

Embora o *framework* permita paralelização em dois níveis, população e em nível de decodificação, com *threads* independentes executando avaliações simultâneas, esses recursos não foram empregados neste trabalho. Todas as execuções foram realizadas de forma sequencial, de modo a manter o foco no comportamento da heurística proposta e evitar que efeitos de paralelismo interfiram na comparação com a literatura, que utiliza um algoritmo sequencial.

5.3 Configuração dos parâmetros

Para a configuração dos parâmetros do BRKGA foram utilizados os valores recomendados em (GONÇALVES; RESENDE, 2016) para o BRKGA. Em (PRASETYO et al., 2015), os autores realizam um levantamento das aplicações do BRKGA na literatura, mostrando que esses parâmetros recomendados são válidos e utilizados nos estudos existentes. Além disso, foram realizados testes com instâncias que combinavam diferentes características para validar a eficiência destes valores. Por fim, a Tabela 5.1 mostra os valores utilizados para o BRKGA, onde p é o tamanho da população, p_e o tamanho do conjunto elite, p_m o número de indivíduos mutantes que serão inseridos em cada geração e ρ_e é a probabilidade do novo indivíduo herdar as características do genitor de melhor qualidade.

Parâmetro	Valor
p	100
p_e	$0.2 \times p$
p_m	$0.1 \times p$
ρ_e	0.7

Tabela 5.1: Parâmetros para o BRKGA

5.4 Resultados

Com as instâncias, o ambiente de testes e os parâmetros do BRKGA definidos, esta seção apresenta os resultados obtidos nos experimentos computacionais. Primeiramente, analisa-se o resultado considerando os decodificadores desenvolvidos, avaliando o impacto do parâmetro k , considerando tanto a qualidade das soluções quanto o tempo de execução. Em seguida, o desempenho do decodificador baseado na heurística NEH, que apresentou os melhores resultados, é comparado com o algoritmo estado da arte da literatura, LIG (LIN; YING, 2022).

5.4.1 Comparação entre decodificadores

Um dos aspectos centrais avaliados nos decodificadores propostos é o tamanho da janela, dado pelo parâmetro k , que define quantas das posições finais da sequência parcial devem ser analisadas para determinar a melhor posição de inserção de um *job*. No decodificador mais simples, correspondente ao caso $k = 0$, o *job* é sempre inserido ao final da sequência parcial, enquanto valores $k > 0$ caracterizam o decodificador baseado na adaptação da heurística NEH, no qual múltiplas posições são avaliadas. Valores reduzidos de k implicam menor custo computacional, mas podem limitar a capacidade de refinamento da sequência; por outro lado, valores maiores ampliam o espaço de busca analisado, ao custo de maior tempo de processamento.

Com o objetivo de comparar o desempenho dos dois decodificadores e analisar o compromisso entre qualidade da solução e tempo de execução, foram realizados experimentos variando-se o valor de k em diferentes escalas relativas ao tamanho das instâncias, adotando-se um critério de parada de 500 gerações. Os resultados obtidos para a instância N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P10, apresentados nas Figuras 5.1 e 5.3, evidenciam uma relação clara entre a redução do *makespan* das soluções e o aumento do tempo computacional à medida que k cresce. Esse padrão de comportamento também foi observado nas demais instâncias avaliadas, variando apenas a magnitude dos ganhos e dos tempos de execução.

Em relação à qualidade das soluções, observa-se na Figura 5.1 que, independentemente do valor de k , o *makespan* apresenta uma tendência de melhora ao longo das

gerações. No entanto, ao comparar o comportamento associado aos diferentes valores de k , verifica-se que, para $k = 0$, a qualidade da solução obtida após 500 gerações é inferior àquela alcançada pelos demais valores de k já nas gerações iniciais. A Figura 5.2 mostra que, a qualidade encontrada pelo algoritmo, utilizando $k = 0$, mesmo após executar por 10 segundos, não consegue encontrar valores de *makespan* menores do que aqueles encontrados quando o algoritmo foi executado utilizando $k = 50$ e $k = 100$ em menos de 1 segundo.

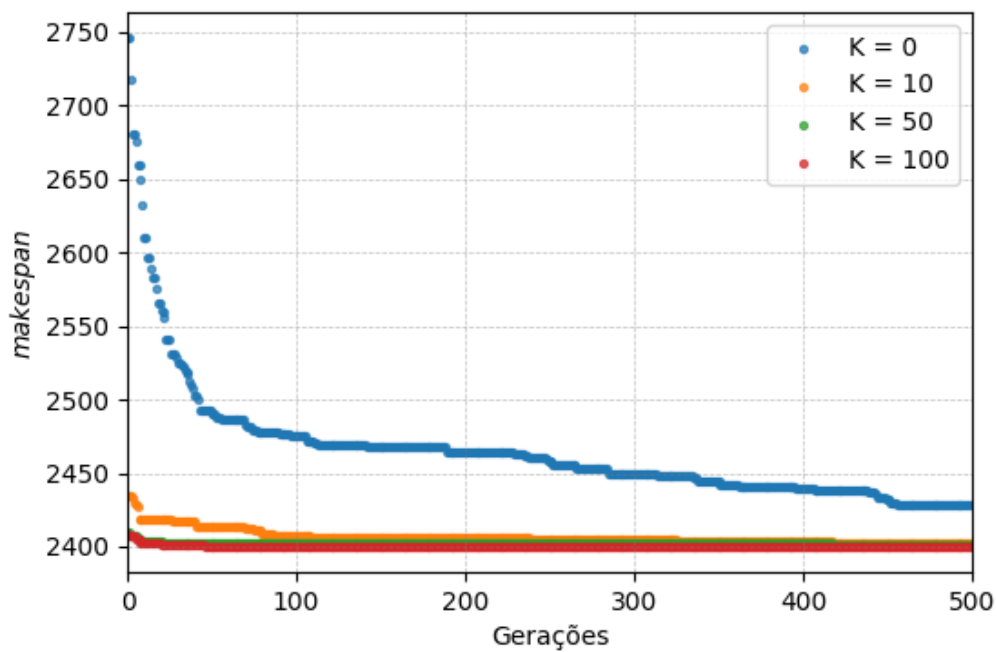
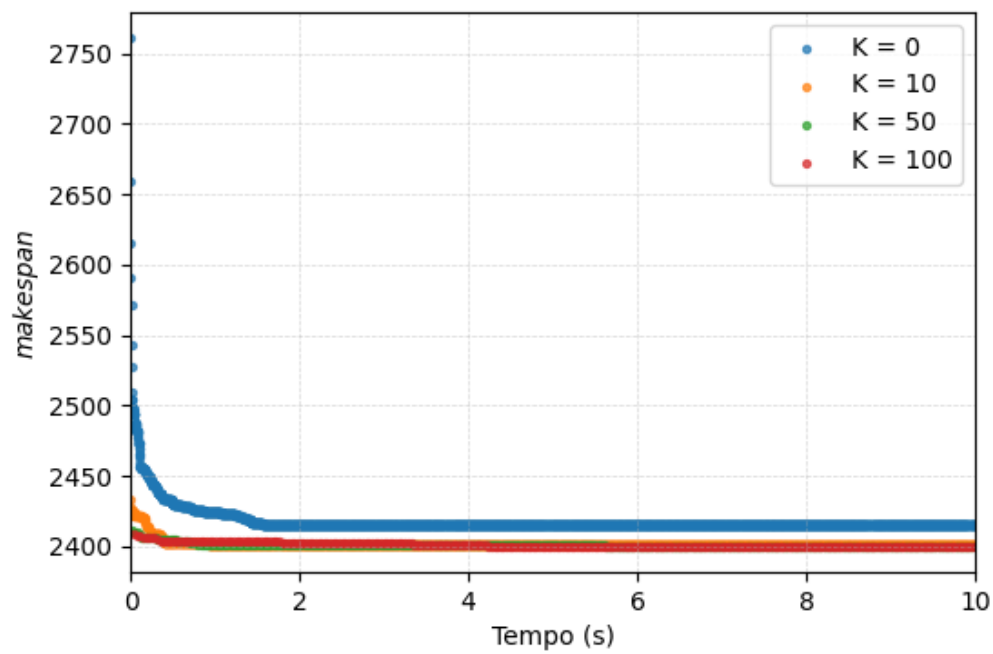
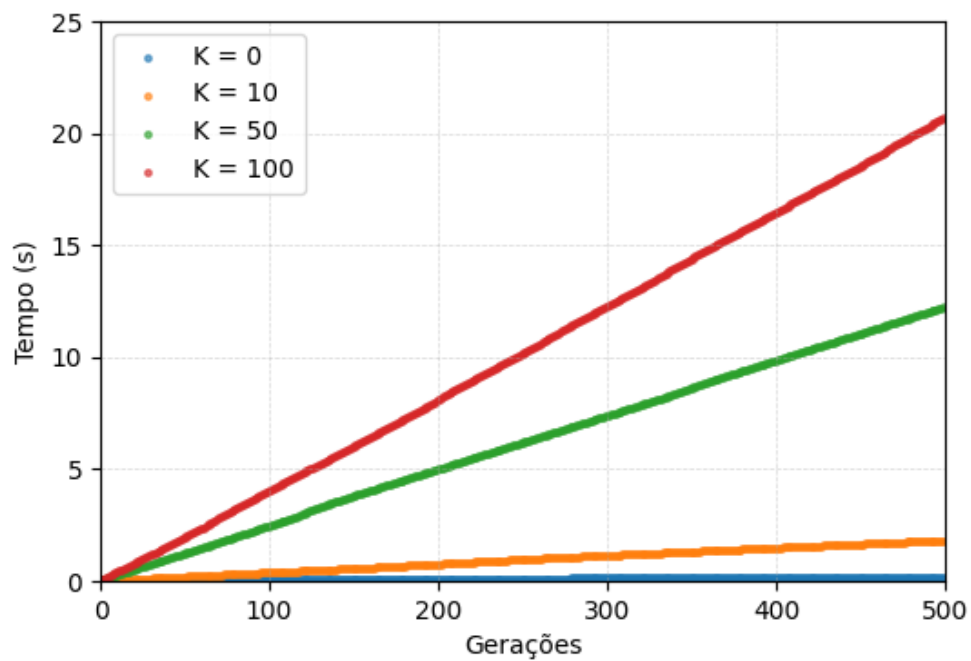


Figura 5.1: Desempenho do decodificador para diferentes valores de k .

Figura 5.2: Tempo $\times$ *makespan*Figura 5.3: Gerações $\times$ Tempo

A Figura 5.3 evidencia que o tempo de execução cresce com o número de gerações para todos os valores de k . Esse comportamento confirma o compromisso entre qualidade da solução e tempo de execução: valores maiores de k ampliam a capacidade de

refinamento da sequência, mas impõem um custo computacional maior no processo de decodificação.

5.4.2 Comparação com os resultados da literatura

Após testes preliminares, observou-se que o algoritmo proposto apresenta tempos de execução compatíveis mesmo quando se utiliza o maior valor possível para a janela de inserção, isto é, $k = n$. Verificou-se também que, para instâncias de pequeno porte, o algoritmo requer um número maior de gerações para atingir soluções competitivas. Em contraste, em instâncias médias e grandes o algoritmo apresentou uma convergência mais eficiente.

Com base nessas observações, o algoritmo foi executado sobre o conjunto completo de instâncias utilizando, para cada caso, o maior valor admissível de k . O número máximo de gerações foi definido em função do tamanho da instância, de modo a equilibrar qualidade da solução e tempo computacional: 500 gerações para instâncias com até 50 *jobs*, 200 gerações para instâncias com 75 *jobs* e 100 gerações para instâncias com 100 *jobs*. Essa estratégia se justifica pelo fato de que, em instâncias maiores, cada geração envolve um custo computacional significativamente mais elevado, enquanto ganhos marginais adicionais tendem a diminuir após um número relativamente pequeno de iterações.

Como citado anteriormente, os testes para o BRKGA foram executados em uma máquina rodando o sistema operacional Ubuntu 24.04 com processador Intel Core i5-1135G7 @ 2.40GHz e 16 GB de memória, enquanto os testes de (LIN; YING, 2022) foram executados em uma máquina com um processador Intel Xeon E5-1620v2 @ 3.70 GHz, com 64 GB de memória, rodando o sistema operacional Windows 10. De acordo com o website www.cpubenchmark.net, o processador i5-1135G7 possui uma pontuação de 9400, enquanto o Xeon E5-1620v2 possui uma pontuação 6506, resultando em uma razão de desempenho de aproximadamente 1,4. Essa razão é utilizada para multiplicar os tempos de execução do algoritmo proposto para tornar a comparação entre as abordagens mais justa.

Assim como nos experimentos de (LIN; YING, 2022), o algoritmo foi executado 5 vezes para cada instância. Dessas cinco execuções foram tabelados o melhor *makespan*

encontrado pelo BRKGA, a média do *makespan* nas cinco execuções e o tempo médio de execução. Os resultados completos, comparando BRKGA com LIG para cada instância individual, podem ser encontrados nas tabelas no apêndice. A Tabela 5.2 sumariza os resultados por combinação dos parâmetros. Para cada combinação é apresentada a média do o tempo médio de execução do BRKGA em segundos (T_BRKGA), a média do tempo médio de execução do LIG em segundos (T_LIG), em quantas instâncias BRKGA conseguiu encontrar valores melhores do que o LIG (M), em quantas ele achou valores iguais (I), quantos ele não conseguiu achar valores melhores ou iguais (P) e o desvio percentual relativo (DPR) entre as melhores soluções encontradas pelas abordagens, dado por:

$$\text{DPR} = \frac{C_{\max}^{\text{BRKGA}} - C_{\max}^{\text{LIG}}}{C_{\max}^{\text{LIG}}} \quad (5.1)$$

Tabela 5.2: Resultados por combinação de parâmetros

n	p_i	s_{ij}	d_{ij}	DPR	T_BRKGA	T_LIG	M	I	P
10	[10,20]	[5,10]	[20,40]	0.00	0.06	1.00	0	10	0
10	[10,20]	[5,10]	[40,60]	0.00	0.06	1.01	0	10	0
10	[10,20]	[10,15]	[20,40]	0.00	0.06	1.01	0	10	0
10	[10,20]	[10,15]	[40,60]	0.00	0.06	1.01	0	10	0
10	[20,30]	[5,10]	[20,40]	0.00	0.06	1.01	0	10	0
10	[20,30]	[5,10]	[40,60]	0.00	0.06	1.01	0	10	0
10	[20,30]	[10,15]	[20,40]	0.00	0.06	1.01	0	10	0
10	[20,30]	[10,15]	[40,60]	0.00	0.06	1.01	0	10	0
15	[10,20]	[5,10]	[20,40]	0.00	0.19	1.51	0	10	0
15	[10,20]	[5,10]	[40,60]	0.00	0.18	1.51	0	10	0
15	[10,20]	[10,15]	[20,40]	0.00	0.20	1.51	0	10	0
15	[10,20]	[10,15]	[40,60]	0.00	0.19	1.51	0	10	0
15	[20,30]	[5,10]	[20,40]	0.00	0.20	1.51	0	10	0
15	[20,30]	[5,10]	[40,60]	0.00	0.19	1.51	0	10	0
15	[20,30]	[10,15]	[20,40]	0.00	0.19	1.51	0	10	0

n	p_i	s_{ij}	d_{ij}	DPR	T_BRKGA	T_LIG	M	I	P
15	[20,30]	[10,15]	[40,60]	0.00	0.20	1.51	0	10	0
20	[10,20]	[5,10]	[20,40]	0.08	0.37	2.01	0	7	3
20	[10,20]	[5,10]	[40,60]	0.10	0.36	2.01	0	6	4
20	[10,20]	[10,15]	[20,40]	0.08	0.37	2.01	0	6	4
20	[10,20]	[10,15]	[40,60]	0.02	0.35	2.01	0	9	1
20	[20,30]	[5,10]	[20,40]	0.03	0.37	2.01	0	8	2
20	[20,30]	[5,10]	[40,60]	0.07	0.36	2.01	0	7	3
20	[20,30]	[10,15]	[20,40]	0.06	0.36	2.01	0	7	3
20	[20,30]	[10,15]	[40,60]	0.07	0.35	2.01	0	6	4
25	[10,20]	[5,10]	[20,40]	0.10	0.68	2.51	0	5	5
25	[10,20]	[5,10]	[40,60]	0.12	0.66	2.51	1	3	6
25	[10,20]	[10,15]	[20,40]	0.05	0.67	2.51	0	7	3
25	[10,20]	[10,15]	[40,60]	0.08	0.65	2.51	1	3	6
25	[20,30]	[5,10]	[20,40]	0.03	0.67	2.51	0	8	2
25	[20,30]	[5,10]	[40,60]	0.04	0.70	2.51	1	5	4
25	[20,30]	[10,15]	[20,40]	0.04	0.67	2.51	1	5	4
25	[20,30]	[10,15]	[40,60]	0.05	0.69	2.51	0	6	4
50	[10,20]	[5,10]	[20,40]	-0.09	4.47	5.01	8	1	1
50	[10,20]	[5,10]	[40,60]	-0.02	4.51	5.01	3	5	2
50	[10,20]	[10,15]	[20,40]	-0.05	4.56	5.01	7	2	1
50	[10,20]	[10,15]	[40,60]	-0.09	4.71	5.01	8	1	1
50	[20,30]	[5,10]	[20,40]	-0.02	4.62	5.01	4	5	1
50	[20,30]	[5,10]	[40,60]	-0.03	4.41	5.01	5	4	1
50	[20,30]	[10,15]	[20,40]	-0.04	4.51	5.01	7	2	1
50	[20,30]	[10,15]	[40,60]	-0.04	4.70	5.01	6	3	1
75	[10,20]	[5,10]	[20,40]	-0.08	5.65	7.51	9	1	0
75	[10,20]	[5,10]	[40,60]	-0.12	5.61	7.51	9	1	0
75	[10,20]	[10,15]	[20,40]	-0.07	5.62	7.51	9	1	0
75	[10,20]	[10,15]	[40,60]	-0.07	5.52	7.51	9	1	0

n	p_i	s_{ij}	d_{ij}	DPR	T_BRKGA	T_LIG	M	I	P
75	[20,30]	[5,10]	[20,40]	-0.06	5.77	7.51	9	1	0
75	[20,30]	[5,10]	[40,60]	-0.06	5.62	7.50	10	0	0
75	[20,30]	[10,15]	[20,40]	-0.05	5.55	7.51	8	2	0
75	[20,30]	[10,15]	[40,60]	-0.05	5.59	7.50	10	0	0
100	[10,20]	[5,10]	[20,40]	-0.09	5.89	10.01	10	0	0
100	[10,20]	[5,10]	[40,60]	-0.12	5.83	10.01	10	0	0
100	[10,20]	[10,15]	[20,40]	-0.06	5.86	10.01	9	1	0
100	[10,20]	[10,15]	[40,60]	-0.09	5.90	10.01	10	0	0
100	[20,30]	[5,10]	[20,40]	-0.05	5.92	10.01	9	1	0
100	[20,30]	[5,10]	[40,60]	-0.07	5.86	10.01	10	0	0
100	[20,30]	[10,15]	[20,40]	-0.04	5.89	10.01	9	1	0
100	[20,30]	[10,15]	[40,60]	-0.06	5.92	10.01	10	0	0

Os resultados indicam que o decodificador baseado no NEH, integrado ao BRKGA, apresenta desempenho limitado em instâncias de pequeno porte, nas quais encontra maior dificuldade em superar as soluções da literatura. Em contrapartida, para instâncias de médio e grande porte, observa-se uma melhora significativa no desempenho da abordagem proposta. Em particular, nas instâncias com 75 e 100 *jobs*, o BRKGA não perde em nenhuma das instâncias avaliadas, obtendo resultados superiores na maioria dos casos, além de apresentar tempos de execução menores.

Assim, ficou demonstrado que a combinação do BRKGA com uma heurística de construção adaptada às características do problema constitui uma abordagem robusta e eficiente. Esse resultado reforça o papel central do decodificador no desempenho final do BRKGA e confirma a adequação da integração entre a estrutura do algoritmo evolutivo e o mecanismo de inserção incremental proposto.

6 Conclusão

Este trabalho abordou o problema de agendamento de tarefas em máquina única com tempos de configuração dependentes da sequência e atrasos de precedência, cujo objetivo é a minimização do *makespan*. Trata-se de um problema de elevada complexidade combinatória, especialmente quando essas duas restrições são consideradas simultaneamente, o que inviabiliza o uso de métodos exatos para instâncias de porte médio e grande. Motivado por essas características e tendo como principal referência o trabalho de (LIN; YING, 2022), foi proposta uma abordagem metaheurística baseada no Algoritmo Genético de Chave Aleatória Tendencioso (BRKGA), combinada a decodificadores especificamente projetados para o problema em estudo.

A principal contribuição deste trabalho consiste no desenvolvimento e na análise de um decodificador baseado em uma adaptação da heurística construtiva NEH, capaz de transformar cromossomos de chaves aleatórias em sequências viáveis, respeitando simultaneamente os tempos de configuração dependentes da sequência e os atrasos de precedência. Diferentemente de abordagens que operam diretamente sobre representações explícitas de sequências, a estratégia adotada permite desacoplar o processo evolutivo da construção da solução, explorando a capacidade do BRKGA de realizar busca global enquanto incorpora conhecimento estrutural do problema no processo de decodificação.

Além do decodificador baseado em NEH, foi também implementado um decodificador mais simples, que insere os *jobs* diretamente ao final da sequência parcial, permitindo analisar o impacto do refinamento local proporcionado pela análise de múltiplas posições. Mostrou-se que esse decodificador simples constitui um caso particular do decodificador NEH quando o parâmetro de janela k é igual a zero, o que possibilitou uma análise sistemática da influência desse parâmetro no desempenho do algoritmo.

Os experimentos computacionais foram conduzidos utilizando o conjunto de instâncias disponibilizado por (LIN; YING, 2022), permitindo uma comparação direta com os resultados da literatura. Os resultados indicaram que, para instâncias pequenas, o BRKGA não apresenta vantagens significativas em relação ao algoritmo estado da arte, o que pode ser

explicado pelo espaço de busca reduzido e pela eficiência de heurísticas determinísticas especializadas nesse cenário. No entanto, à medida que o número de *jobs* aumenta, o BRKGA passa a apresentar desempenho superior, superando consistentemente o método de referência em instâncias médias e grandes. Esse comportamento evidencia que abordagens populacionais tornam-se mais adequadas conforme a complexidade combinatória do problema cresce, especialmente quando combinadas com decodificadores capazes de realizar refinamentos locais eficientes.

A análise do parâmetro k demonstrou que existe um compromisso claro entre custo computacional e qualidade da solução. Valores muito pequenos limitam a capacidade de melhoria da sequência, enquanto valores muito grandes aumentam o tempo de execução sem ganhos proporcionais de qualidade. Os resultados indicaram que valores intermediários de k oferecem a melhor relação entre tempo e desempenho, reforçando a importância do ajuste adequado do decodificador no contexto do BRKGA.

Como trabalhos futuros, diversas extensões metodológicas podem ser investigadas a partir da abordagem proposta. Uma possibilidade natural é a exploração de estratégias baseadas em múltiplas populações independentes, que podem evoluir de forma parcialmente desacoplada e trocar informações periodicamente, aumentando a diversidade genética e reduzindo o risco de convergência prematura. De forma complementar, mecanismos de *restart* controlado podem ser incorporados ao processo evolutivo, permitindo reinicializar parcialmente a população quando não há melhora significativa ao longo de várias gerações.

Outra direção promissora consiste na integração do BRKGA com procedimentos de busca local, resultando em uma abordagem híbrida. Nesse contexto, o decodificador poderia ser estendido para aplicar movimentos locais adicionais sobre as sequências construídas, explorando vizinhanças mais amplas após a etapa de decodificação e potencialmente refinando ainda mais as soluções obtidas pelo algoritmo evolutivo.

Além disso, a estratégia de decodificação baseada na heurística NEH adaptada pode ser estendida para outras variantes do problema de agendamento. Em particular, a generalização do modelo para ambientes com múltiplas máquinas representa uma linha de pesquisa relevante, uma vez que muitas aplicações práticas envolvem configurações mais

complexas do que a máquina única. A adaptação do decodificador e do cálculo incremental do *makespan* para esses ambientes permitiria avaliar a robustez e a flexibilidade da abordagem proposta em cenários ainda mais desafiadores.

7 Apêndice

7.1 Apêndice A: Resultados por instâncias

Neste apêndice são encontrados os resultados dos testes executados para o BRKGA e a comparação destes resultados com os resultados da literatura para cada instância no conjunto de testes. O tempo médio do BRKGA foi multiplicado pela constante de performance, assim como explicado na Subseção 5.4.2 para tornar esta comparação mais justa. As tabelas que seguem mostram, para cada instâncias, o *makespan* médio, o tempo de execução médio em segundos e o melhor *makespan* encontrado para o LIG e para o BRKGA e esses dois resultados são comparados utilizando a desvio percentual relativo (DPR).

Tabela 7.1: Resultados para instâncias com 10 *jobs*

Instância	LIG			BRKGA			DPR
	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P1	193.0	1.01	193	193.0	0.06	193	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P2	188.0	1.0	188	188.0	0.05	188	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P3	190.0	1.0	190	190.0	0.07	190	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P4	196.0	1.0	196	196.0	0.07	196	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P5	201.0	1.0	201	201.0	0.06	201	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P6	223.0	1.0	223	223.0	0.06	223	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P7	198.0	1.0	198	198.0	0.06	198	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P8	195.0	1.0	195	195.0	0.07	195	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P9	190.0	1.0	190	190.0	0.06	190	0.0
N10_P(10-20)_S(5-10)_A(20-40)_W5_Wo2_P10	209.0	1.01	209	209.0	0.06	209	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P1	193.0	1.01	193	193.0	0.06	193	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P2	192.0	1.01	192	192.0	0.06	192	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P3	190.0	1.01	190	190.0	0.06	190	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P4	197.0	1.01	197	197.0	0.06	197	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P5	254.0	1.01	254	254.0	0.05	254	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P6	267.0	1.01	267	267.0	0.06	267	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P7	202.0	1.01	202	202.0	0.07	202	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P8	195.0	1.01	195	195.0	0.06	195	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P9	192.0	1.01	192	192.0	0.06	192	0.0
N10_P(10-20)_S(5-10)_A(40-60)_W5_Wo2_P10	211.0	1.01	211	211.0	0.06	211	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P1	236.0	1.01	236	236.0	0.06	236	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P2	231.0	1.01	231	231.0	0.05	231	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P3	235.0	1.01	235	235.0	0.07	235	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P4	241.0	1.01	241	241.0	0.06	241	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P5	246.0	1.01	246	246.0	0.06	246	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P6	264.0	1.01	264	264.0	0.06	264	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P7	243.0	1.01	243	243.0	0.06	243	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P8	240.0	1.0	240	240.0	0.06	240	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P9	235.0	1.01	235	235.0	0.06	235	0.0
N10_P(10-20)_S(10-15)_A(20-40)_W5_Wo2_P10	252.0	1.01	252	252.0	0.06	252	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P1	238.0	1.0	238	238.0	0.06	238	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P2	235.0	1.01	235	235.0	0.06	235	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P3	235.0	1.01	235	235.0	0.07	235	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P4	241.0	1.01	241	241.0	0.06	241	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P5	264.0	1.01	264	264.0	0.06	264	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P6	281.0	1.01	281	281.0	0.06	281	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P7	243.0	1.01	243	243.0	0.06	243	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P8	240.0	1.01	240	240.0	0.07	240	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P9	235.0	1.01	235	235.0	0.06	235	0.0
N10_P(10-20)_S(10-15)_A(40-60)_W5_Wo2_P10	255.0	1.01	255	255.0	0.06	255	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P1	291.0	1.01	291	291.0	0.06	291	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P2	286.0	1.01	286	286.0	0.05	286	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P3	290.0	1.01	290	290.0	0.08	290	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P4	296.0	1.01	296	296.0	0.06	296	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P5	301.0	1.01	301	301.0	0.06	301	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P6	319.0	1.01	319	319.0	0.06	319	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P7	298.0	1.01	298	298.0	0.07	298	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P8	295.0	1.01	295	295.0	0.07	295	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P9	290.0	1.0	290	290.0	0.06	290	0.0
N10_P(20-30)_S(5-10)_A(20-40)_W5_Wo2_P10	307.0	1.01	307	307.0	0.07	307	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P1	293.0	1.01	293	293.0	0.07	293	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P2	290.0	1.01	290	290.0	0.06	290	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P3	290.0	1.01	290	290.0	0.07	290	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P4	296.0	1.0	296	296.0	0.06	296	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P5	317.0	1.01	317	317.0	0.06	317	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P6	336.0	1.01	336	336.0	0.06	336	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P7	298.0	1.01	298	298.0	0.06	298	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P8	295.0	1.01	295	295.0	0.06	295	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P9	290.0	1.01	290	290.0	0.05	290	0.0
N10_P(20-30)_S(5-10)_A(40-60)_W5_Wo2_P10	310.0	1.01	310	310.0	0.06	310	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P1	336.0	1.01	336	336.0	0.07	336	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P2	331.0	1.01	331	331.0	0.05	331	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P3	335.0	1.01	335	335.0	0.07	335	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P4	341.0	1.01	341	341.0	0.07	341	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P5	343.0	1.01	343	343.0	0.07	343	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P6	364.0	1.01	364	364.0	0.06	364	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P7	343.0	1.01	343	343.0	0.07	343	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P8	340.0	1.01	340	340.0	0.06	340	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P9	335.0	1.0	335	335.0	0.06	335	0.0
N10_P(20-30)_S(10-15)_A(20-40)_W5_Wo2_P10	352.0	1.01	352	352.0	0.06	352	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P1	338.0	1.01	338	338.0	0.07	338	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P2	333.0	1.0	333	333.0	0.06	333	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P3	335.0	1.0	335	335.0	0.07	335	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P4	341.0	1.0	341	341.0	0.07	341	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P5	346.0	1.01	346	346.0	0.06	346	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P6	368.0	1.01	368	368.0	0.06	368	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P7	343.0	1.01	343	343.0	0.06	343	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P8	340.0	1.01	340	340.0	0.07	340	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P9	335.0	1.01	335	335.0	0.06	335	0.0
N10_P(20-30)_S(10-15)_A(40-60)_W5_Wo2_P10	354.0	1.01	354	354.0	0.06	354	0.0

Tabela 7.2: Resultados para instâncias com 15 *jobs*

Instância	LIG			BRKGA			DPR
	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P1	285.0	1.51	285	285.0	0.25	285	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P2	278.0	1.5	278	278.0	0.18	278	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P3	284.0	1.51	284	284.0	0.19	284	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P4	296.0	1.51	296	296.0	0.21	296	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P5	282.0	1.51	282	282.0	0.14	282	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P6	311.0	1.51	311	311.0	0.19	311	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P7	298.0	1.51	298	298.0	0.17	298	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P8	294.0	1.51	294	294.0	0.18	294	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P9	290.0	1.51	290	290.0	0.19	290	0.0
N15_P(10-20)_S(5-10)_A(20-40)_W7_Wo3_P10	306.0	1.51	306	306.0	0.22	306	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P1	285.0	1.51	285	285.0	0.2	285	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P2	278.0	1.51	278	278.2	0.16	278	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P3	285.0	1.51	285	285.0	0.19	285	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P4	297.0	1.51	297	297.0	0.21	297	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P5	284.0	1.51	284	284.0	0.17	284	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P6	313.0	1.51	313	313.0	0.17	313	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P7	298.0	1.51	298	298.0	0.16	298	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P8	295.0	1.51	295	295.0	0.16	295	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P9	290.0	1.5	290	290.0	0.21	290	0.0
N15_P(10-20)_S(5-10)_A(40-60)_W7_Wo3_P10	306.0	1.51	306	306.0	0.19	306	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P1	355.0	1.51	355	355.0	0.2	355	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P2	348.0	1.51	348	348.0	0.22	348	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P3	353.0	1.51	353	353.0	0.22	353	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P4	366.0	1.51	366	366.0	0.22	366	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P5	351.0	1.51	351	351.0	0.17	351	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P6	380.0	1.51	380	380.0	0.21	380	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P7	368.0	1.51	368	368.0	0.21	368	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P8	363.0	1.51	363	363.0	0.19	363	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P9	360.0	1.51	360	360.0	0.17	360	0.0
N15_P(10-20)_S(10-15)_A(20-40)_W7_Wo3_P10	376.0	1.51	376	376.0	0.2	376	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P1	355.0	1.51	355	355.0	0.21	355	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P2	348.0	1.51	348	348.0	0.16	348	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P3	354.0	1.51	354	354.0	0.19	354	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P4	366.0	1.51	366	366.0	0.22	366	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P5	352.0	1.51	352	352.0	0.16	352	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P6	382.0	1.51	382	382.0	0.19	382	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P7	368.0	1.51	368	368.0	0.19	368	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P8	364.0	1.51	364	364.8	0.17	364	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P9	360.0	1.51	360	360.0	0.22	360	0.0
N15_P(10-20)_S(10-15)_A(40-60)_W7_Wo3_P10	376.0	1.51	376	376.0	0.18	376	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P1	435.0	1.51	435	435.0	0.24	435	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P2	428.0	1.51	428	428.0	0.2	428	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P3	433.0	1.51	433	433.0	0.22	433	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P4	446.0	1.51	446	446.0	0.2	446	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P5	431.0	1.51	431	431.0	0.17	431	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P6	460.0	1.5	460	460.0	0.18	460	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P7	448.0	1.51	448	448.0	0.18	448	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P8	443.0	1.51	443	443.0	0.18	443	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P9	440.0	1.51	440	440.0	0.23	440	0.0
N15_P(20-30)_S(5-10)_A(20-40)_W7_Wo3_P10	456.0	1.51	456	456.0	0.19	456	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P1	435.0	1.51	435	435.0	0.18	435	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P2	428.0	1.51	428	428.0	0.21	428	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P3	434.0	1.51	434	434.0	0.16	434	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P4	446.0	1.51	446	446.0	0.22	446	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P5	432.0	1.51	432	432.0	0.16	432	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P6	462.0	1.51	462	462.0	0.2	462	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P7	448.0	1.51	448	448.0	0.19	448	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P8	444.0	1.51	444	444.0	0.16	444	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P9	440.0	1.51	440	440.0	0.17	440	0.0
N15_P(20-30)_S(5-10)_A(40-60)_W7_Wo3_P10	456.0	1.51	456	456.0	0.22	456	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P1	505.0	1.51	505	505.0	0.23	505	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P2	497.0	1.51	497	497.0	0.14	497	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P3	503.0	1.51	503	503.0	0.22	503	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P4	516.0	1.51	516	516.0	0.19	516	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P5	501.0	1.51	501	501.0	0.19	501	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P6	530.0	1.51	530	530.0	0.17	530	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P7	518.0	1.51	518	518.0	0.18	518	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P8	513.0	1.5	513	513.0	0.17	513	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P9	510.0	1.51	510	510.0	0.22	510	0.0
N15_P(20-30)_S(10-15)_A(20-40)_W7_Wo3_P10	526.0	1.51	526	526.0	0.14	526	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P1	505.0	1.51	505	505.0	0.2	505	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P2	498.0	1.51	498	498.0	0.23	498	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P3	504.0	1.51	504	504.0	0.2	504	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P4	516.0	1.5	516	516.0	0.2	516	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P5	502.0	1.5	502	502.0	0.17	502	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P6	531.0	1.51	531	531.0	0.18	531	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P7	518.0	1.51	518	518.0	0.2	518	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P8	514.0	1.51	514	514.0	0.18	514	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P9	510.0	1.51	510	510.0	0.21	510	0.0
N15_P(20-30)_S(10-15)_A(40-60)_W7_Wo3_P10	526.0	1.51	526	526.0	0.21	526	0.0

Tabela 7.3: Resultados para instâncias com 20 jobs

Instância	LIG			BRKGA			DPR
	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P1	386.0	2.01	386	386.0	0.4	386	0.0
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P2	369.0	2.01	369	370.0	0.42	370	0.27
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P3	385.0	2.01	385	385.6	0.39	385	0.0
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P4	395.8	2.01	395	396.0	0.31	396	0.25
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P5	375.0	2.01	375	376.0	0.4	376	0.27
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P6	415.0	2.01	415	415.6	0.35	415	0.0
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P7	396.0	2.01	396	396.4	0.35	396	0.0
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P8	388.2	2.01	388	389.2	0.36	388	0.0
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P9	406.0	2.01	406	406.0	0.32	406	0.0
N20_P(10-20)_S(5-10)_A(20-40)_W10_Wo5_P10	393.0	2.01	393	393.0	0.41	393	0.0
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P1	386.0	2.01	386	386.6	0.37	386	0.0
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P2	369.4	2.01	369	370.0	0.41	370	0.27
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P3	385.0	2.0	385	386.0	0.39	386	0.26
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P4	396.0	2.01	396	396.4	0.38	396	0.0
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P5	375.6	2.01	375	376.0	0.35	376	0.27
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P6	415.0	2.01	415	416.0	0.31	416	0.24
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P7	396.0	2.01	396	396.6	0.41	396	0.0
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P8	390.0	2.01	390	390.2	0.32	390	0.0
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P9	414.0	2.01	414	414.8	0.3	414	0.0
N20_P(10-20)_S(5-10)_A(40-60)_W10_Wo5_P10	393.0	2.01	393	393.0	0.4	393	0.0
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P1	481.0	2.01	481	481.0	0.38	481	0.0
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P2	464.0	2.01	464	465.0	0.41	465	0.22
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P3	479.0	2.01	479	479.0	0.37	479	0.0
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P4	490.6	2.01	490	491.0	0.37	491	0.2
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P5	469.0	2.01	469	469.0	0.38	469	0.0
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P6	510.0	2.01	510	510.8	0.32	510	0.0
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P7	491.0	2.01	491	491.4	0.39	491	0.0
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P8	483.0	2.01	483	484.0	0.34	484	0.21
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P9	501.0	2.01	501	501.0	0.29	501	0.0
N20_P(10-20)_S(10-15)_A(20-40)_W10_Wo5_P10	488.0	2.01	488	489.0	0.43	489	0.2
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P1	481.0	2.01	481	481.0	0.31	481	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P2	464.0	2.01	464	464.4	0.35	464	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P3	480.0	2.01	480	480.8	0.31	480	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P4	490.2	2.01	490	490.8	0.41	490	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P5	470.0	2.01	470	471.0	0.33	471	0.21
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P6	510.0	2.01	510	510.0	0.34	510	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P7	491.0	2.01	491	491.4	0.41	491	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P8	483.2	2.01	483	484.2	0.39	483	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P9	505.0	2.01	505	505.0	0.32	505	0.0
N20_P(10-20)_S(10-15)_A(40-60)_W10_Wo5_P10	488.0	2.01	488	488.6	0.35	488	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P1	586.0	2.01	586	586.0	0.39	586	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P2	569.0	2.01	569	569.4	0.41	569	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P3	584.0	2.01	584	584.0	0.36	584	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P4	595.6	2.01	595	596.0	0.41	596	0.17
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P5	574.0	2.01	574	574.8	0.33	574	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P6	615.0	2.01	615	615.2	0.31	615	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P7	596.0	2.01	596	596.0	0.42	596	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P8	588.2	2.01	588	589.0	0.34	589	0.17
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P9	606.0	2.01	606	606.0	0.32	606	0.0
N20_P(20-30)_S(5-10)_A(20-40)_W10_Wo5_P10	593.0	2.01	593	593.4	0.37	593	0.0
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P1	586.0	2.01	586	586.0	0.32	586	0.0
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P2	569.0	2.01	569	570.6	0.33	570	0.18
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P3	585.0	2.01	585	585.2	0.34	585	0.0
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P4	595.2	2.01	595	596.0	0.43	596	0.17
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P5	575.0	2.0	575	575.0	0.41	575	0.0
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P6	615.0	2.01	615	615.0	0.32	615	0.0
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P7	596.0	2.01	596	596.0	0.44	596	0.0
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P8	588.6	2.0	588	590.0	0.35	590	0.34
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P9	610.0	2.01	610	610.0	0.3	610	0.0
N20_P(20-30)_S(5-10)_A(40-60)_W10_Wo5_P10	593.0	2.01	593	593.0	0.35	593	0.0
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P1	681.0	2.0	681	681.0	0.32	681	0.0
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P2	664.0	2.01	664	664.6	0.42	664	0.0
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P3	679.0	2.01	679	679.0	0.39	679	0.0
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P4	690.6	2.01	690	691.0	0.4	691	0.14
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P5	669.0	2.01	669	669.2	0.39	669	0.0
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P6	710.0	2.01	710	711.0	0.3	711	0.14
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P7	691.0	2.01	691	691.0	0.35	691	0.0
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P8	683.2	2.01	683	685.0	0.36	685	0.29
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P9	701.0	2.01	701	701.0	0.33	701	0.0
N20_P(20-30)_S(10-15)_A(20-40)_W10_Wo5_P10	688.0	2.01	688	688.4	0.35	688	0.0
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P1	681.0	2.01	681	681.0	0.3	681	0.0
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P2	664.0	2.01	664	665.2	0.36	665	0.15
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P3	680.0	2.01	680	680.0	0.37	680	0.0
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P4	690.8	2.01	690	691.2	0.38	691	0.14
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P5	669.6	2.01	669	670.0	0.42	670	0.15
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P6	710.0	2.01	710	710.2	0.33	710	0.0
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P7	691.0	2.01	691	691.0	0.39	691	0.0
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P8	683.2	2.01	683	685.0	0.3	685	0.29
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P9	701.0	2.01	701	701.0	0.3	701	0.0
N20_P(20-30)_S(10-15)_A(40-60)_W10_Wo5_P10	688.0	2.01	688	688.2	0.35	688	0.0

Tabela 7.4: Resultados para instâncias com 25 jobs

Instância	LIG			BRKGA			DPR
	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P1	482.4	2.5	482	483.6	0.67	483	0.21
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P2	465.0	2.5	465	465.8	0.74	465	0.0
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P3	478.0	2.5	478	479.0	0.66	479	0.21
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P4	489.0	2.51	489	489.4	0.73	489	0.0
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P5	468.2	2.51	468	468.8	0.71	468	0.0
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P6	504.0	2.51	504	504.0	0.69	504	0.0
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P7	490.8	2.51	490	491.0	0.64	491	0.2
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P8	492.0	2.51	491	491.6	0.58	491	0.0
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P9	480.8	2.51	480	481.8	0.55	481	0.21
N25_P(10-20)_S(5-10)_A(20-40)_W12_Wo6_P10	488.8	2.51	488	489.0	0.78	489	0.2
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P1	483.0	2.51	483	483.2	0.73	482	-0.21
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P2	464.4	2.51	464	465.4	0.71	465	0.22
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P3	478.2	2.51	478	479.0	0.57	479	0.21
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P4	489.8	2.51	489	490.0	0.62	490	0.2
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P5	468.6	2.51	468	469.6	0.67	468	0.0
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P6	503.8	2.5	503	504.4	0.67	504	0.2
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P7	491.0	2.5	491	491.0	0.69	491	0.0
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P8	491.4	2.51	491	492.6	0.71	492	0.2
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P9	481.4	2.5	480	482.6	0.53	482	0.42
N25_P(10-20)_S(5-10)_A(40-60)_W12_Wo6_P10	489.2	2.5	489	489.6	0.68	489	0.0
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P1	602.8	2.51	602	603.0	0.73	603	0.17
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P2	584.8	2.51	584	584.2	0.64	584	0.0
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P3	598.2	2.51	598	599.0	0.6	599	0.17
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P4	609.0	2.51	609	609.0	0.71	609	0.0
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P5	588.2	2.51	588	588.4	0.67	588	0.0
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P6	623.6	2.51	623	624.2	0.81	624	0.16
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P7	610.6	2.5	610	610.8	0.67	610	0.0
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P8	611.6	2.5	611	611.8	0.6	611	0.0
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P9	601.0	2.5	601	601.6	0.62	601	0.0
N25_P(10-20)_S(10-15)_A(20-40)_W12_Wo6_P10	609.0	2.51	609	609.2	0.68	609	0.0
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P1	602.4	2.51	602	603.0	0.71	603	0.17
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P2	585.0	2.51	585	586.4	0.66	586	0.17
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P3	598.2	2.51	598	598.6	0.63	598	0.0
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P4	609.0	2.51	609	609.0	0.55	609	0.0
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P5	588.4	2.51	588	589.6	0.67	589	0.17
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P6	624.0	2.51	624	624.2	0.66	623	-0.16
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P7	611.0	2.5	611	611.0	0.65	611	0.0
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P8	612.2	2.5	611	612.4	0.6	612	0.16
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P9	600.6	2.51	600	601.2	0.59	601	0.17
N25_P(10-20)_S(10-15)_A(40-60)_W12_Wo6_P10	608.8	2.51	608	609.0	0.76	609	0.16
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P1	732.6	2.51	732	733.0	0.75	733	0.14
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P2	715.0	2.5	715	715.0	0.69	715	0.0
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P3	728.4	2.51	728	728.8	0.64	728	0.0
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P4	739.0	2.51	739	739.0	0.7	739	0.0
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P5	718.2	2.5	718	718.8	0.7	718	0.0
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P6	753.8	2.5	753	754.0	0.6	754	0.13
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P7	741.0	2.51	741	741.0	0.65	741	0.0
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P8	742.2	2.5	742	742.4	0.63	742	0.0
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P9	730.6	2.51	730	730.8	0.68	730	0.0
N25_P(20-30)_S(5-10)_A(20-40)_W12_Wo6_P10	739.0	2.51	739	739.0	0.66	739	0.0
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P1	732.4	2.51	732	732.8	0.78	732	0.0
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P2	714.8	2.51	714	715.6	0.8	714	0.0
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P3	728.2	2.51	728	729.4	0.77	729	0.14
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P4	739.0	2.51	739	739.8	0.73	739	0.0
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P5	718.2	2.51	718	719.0	0.68	719	0.14
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P6	753.6	2.51	753	754.2	0.73	754	0.13
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P7	741.0	2.5	741	741.0	0.64	741	0.0
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P8	742.0	2.51	741	741.8	0.59	741	0.0
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P9	730.6	2.51	730	731.8	0.56	731	0.14
N25_P(20-30)_S(5-10)_A(40-60)_W12_Wo6_P10	739.0	2.51	739	738.8	0.7	738	-0.14
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P1	852.2	2.51	852	853.0	0.74	853	0.12
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P2	835.0	2.51	835	835.6	0.62	835	0.0
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P3	848.0	2.51	848	849.0	0.63	849	0.12
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P4	859.0	2.51	859	859.0	0.71	859	0.0
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P5	838.2	2.51	838	838.4	0.64	838	0.0
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P6	873.6	2.5	873	873.6	0.68	873	0.0
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P7	860.6	2.5	860	861.2	0.77	861	0.12
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P8	862.0	2.51	861	862.8	0.65	862	0.12
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P9	850.6	2.5	850	851.2	0.62	850	0.0
N25_P(20-30)_S(10-15)_A(20-40)_W12_Wo6_P10	859.0	2.51	859	858.6	0.64	858	-0.12
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P1	852.2	2.51	852	852.4	0.68	852	0.0
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P2	834.8	2.5	834	835.6	0.64	835	0.12
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P3	848.0	2.51	848	848.4	0.7	848	0.0
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P4	859.0	2.51	859	859.4	0.81	859	0.0
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P5	838.2	2.51	838	838.6	0.67	838	0.0
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P6	873.4	2.5	873	874.0	0.7	874	0.11
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P7	860.8	2.5	860	861.0	0.66	861	0.12
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P8	862.4	2.51	862	863.0	0.68	863	0.12
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P9	850.8	2.51	850	851.2	0.65	850	0.0
N25_P(20-30)_S(10-15)_A(40-60)_W12_Wo6_P10	859.0	2.51	859	859.0	0.75	859	0.0

Tabela 7.5: Resultados para instâncias com 50 jobs

Instância	LIG			BRKGA			DPR
	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P1	984.0	5.01	984	984.8	4.16	983	-0.1
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P2	950.6	5.01	950	949.4	4.59	949	-0.11
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P3	974.2	5.01	973	973.8	4.38	973	0.0
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P4	964.2	5.01	964	963.0	4.73	962	-0.21
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P5	956.4	5.01	956	954.8	4.55	954	-0.21
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P6	957.6	5.01	957	956.8	4.33	956	-0.1
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P7	971.2	5.01	971	969.6	4.45	969	-0.21
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P8	988.6	5.01	988	987.2	4.39	987	-0.1
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P9	954.6	5.01	954	953.4	4.62	953	-0.1
N50_P(10-20)_S(5-10)_A(20-40)_W25_Wo12_P10	961.0	5.0	959	961.4	4.49	961	0.21
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P1	985.4	5.01	984	986.0	4.47	984	0.0
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P2	950.8	5.01	950	951.4	4.31	950	0.0
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P3	974.8	5.01	973	975.2	4.64	974	0.1
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P4	963.8	5.01	963	963.2	4.68	962	-0.1
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P5	956.0	5.01	955	955.4	4.16	954	-0.1
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P6	957.8	5.01	956	957.6	4.55	956	0.0
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P7	970.6	5.01	970	971.0	4.57	970	0.0
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P8	989.0	5.0	988	988.4	4.95	988	0.0
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P9	955.2	5.01	955	953.4	4.28	952	-0.31
N50_P(10-20)_S(5-10)_A(40-60)_W25_Wo12_P10	964.4	5.01	962	966.2	4.51	964	0.21
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P1	1228.8	5.01	1228	1228.6	4.27	1227	-0.08
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P2	1195.4	5.01	1195	1194.8	4.21	1194	-0.08
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P3	1219.0	5.01	1218	1218.4	4.39	1218	0.0
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P4	1208.8	5.0	1208	1208.2	4.65	1208	0.0
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P5	1200.4	5.0	1200	1200.2	4.76	1199	-0.08
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P6	1202.8	5.0	1202	1202.0	4.28	1201	-0.08
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P7	1216.0	5.01	1215	1215.0	5.06	1214	-0.08
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P8	1233.8	5.0	1233	1232.6	4.68	1232	-0.08
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P9	1198.6	5.01	1198	1197.0	4.59	1197	-0.08
N50_P(10-20)_S(10-15)_A(20-40)_W25_Wo12_P10	1205.6	5.01	1204	1205.8	4.71	1205	0.08
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P1	1229.4	5.01	1229	1228.6	4.58	1226	-0.24
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P2	1196.0	5.01	1195	1195.4	4.3	1194	-0.08
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P3	1219.2	5.01	1218	1218.8	4.84	1218	0.0
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P4	1208.8	5.01	1208	1208.8	4.47	1207	-0.08
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P5	1201.4	5.01	1201	1199.6	4.67	1199	-0.17
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P6	1203.4	5.01	1203	1202.0	4.93	1201	-0.17
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P7	1216.8	5.01	1216	1215.2	4.57	1214	-0.16
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P8	1233.6	5.01	1233	1233.2	4.78	1232	-0.08
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P9	1199.2	5.01	1198	1198.6	5.0	1197	-0.08
N50_P(10-20)_S(10-15)_A(40-60)_W25_Wo12_P10	1207.6	5.01	1206	1209.0	4.94	1208	0.17
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P1	1483.4	5.01	1482	1482.8	4.17	1482	0.0
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P2	1450.4	5.0	1450	1449.6	4.75	1449	-0.07
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P3	1473.6	5.01	1473	1473.6	4.79	1473	0.0
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P4	1463.8	5.01	1463	1463.2	4.67	1463	0.0
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P5	1455.4	5.01	1454	1455.6	4.09	1455	0.07
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P6	1458.0	5.0	1457	1456.4	4.22	1456	-0.07
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P7	1470.8	5.01	1470	1470.0	4.78	1469	-0.07
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P8	1488.4	5.01	1487	1488.4	5.0	1487	0.0
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P9	1453.6	5.01	1453	1452.8	4.54	1452	-0.07
N50_P(20-30)_S(5-10)_A(20-40)_W25_Wo12_P10	1460.6	5.0	1459	1460.0	5.15	1459	0.0
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P1	1484.6	5.01	1484	1484.2	4.1	1484	0.0
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P2	1450.6	5.01	1449	1450.2	4.33	1450	0.07
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P3	1474.0	5.01	1474	1473.6	4.45	1473	-0.07
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P4	1463.8	5.01	1463	1463.2	4.25	1463	0.0
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P5	1456.6	5.0	1456	1456.2	4.53	1455	-0.07
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P6	1458.2	5.01	1458	1457.4	4.7	1457	-0.07
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P7	1470.6	5.01	1470	1471.8	4.37	1470	0.0
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P8	1489.0	5.01	1488	1487.6	4.34	1487	-0.07
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P9	1454.2	5.0	1453	1453.0	4.33	1452	-0.07
N50_P(20-30)_S(5-10)_A(40-60)_W25_Wo12_P10	1462.0	5.0	1461	1463.2	4.74	1461	0.0
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P1	1729.4	5.01	1728	1726.8	4.21	1726	-0.12
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P2	1695.4	5.01	1695	1695.0	4.32	1694	-0.06
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P3	1719.0	5.01	1718	1718.0	5.0	1718	0.0
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P4	1708.6	5.01	1708	1708.2	4.64	1708	0.0
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P5	1700.8	5.01	1700	1699.8	4.9	1699	-0.06
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P6	1702.4	5.0	1702	1702.6	4.2	1701	-0.06
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P7	1715.2	5.01	1714	1715.4	4.35	1715	0.06
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P8	1733.8	5.01	1733	1733.0	4.76	1732	-0.06
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P9	1698.8	5.01	1698	1697.4	4.29	1697	-0.06
N50_P(20-30)_S(10-15)_A(20-40)_W25_Wo12_P10	1706.2	5.01	1706	1706.6	4.4	1705	-0.06
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P1	1729.0	5.01	1729	1727.6	4.39	1726	-0.17
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P2	1695.6	5.01	1695	1694.6	4.4	1694	-0.06
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P3	1719.0	5.01	1718	1719.0	5.02	1718	0.0
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P4	1709.0	5.01	1708	1708.6	4.6	1708	0.0
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P5	1701.2	5.01	1700	1700.2	4.3	1700	0.0
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P6	1702.4	5.01	1702	1701.0	5.26	1701	-0.06
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P7	1715.6	5.0	1715	1714.4	4.88	1714	-0.06
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P8	1733.4	5.01	1733	1732.6	4.84	1732	-0.06
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P9	1698.8	5.01	1698	1697.4	4.36	1697	-0.06
N50_P(20-30)_S(10-15)_A(40-60)_W25_Wo12_P10	1706.6	5.01	1705	1706.2	4.98	1706	0.06

Tabela 7.6: Resultados para instâncias com 75 jobs

Instância	LIG			BRKGA			DPR
	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P1	1451.0	7.51	1450	1449.0	5.78	1449	-0.07
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P2	1438.4	7.51	1438	1437.6	5.95	1437	-0.07
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P3	1455.0	7.51	1454	1453.4	5.71	1453	-0.07
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P4	1457.0	7.51	1456	1455.2	6.11	1455	-0.07
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P5	1440.6	7.51	1439	1439.8	5.13	1439	0.0
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P6	1457.0	7.51	1456	1454.6	5.76	1454	-0.14
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P7	1459.6	7.51	1459	1458.2	5.59	1458	-0.07
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P8	1466.8	7.5	1466	1465.2	5.29	1465	-0.07
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P9	1453.6	7.51	1452	1451.8	5.65	1451	-0.07
N75_P(10-20)_S(5-10)_A(20-40)_W37_Wo18_P10	1432.8	7.5	1432	1431.2	5.53	1430	-0.14
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P1	1451.4	7.51	1451	1449.2	5.87	1449	-0.14
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P2	1439.2	7.51	1439	1437.6	5.44	1437	-0.14
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P3	1455.0	7.51	1453	1453.0	5.59	1453	0.0
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P4	1457.0	7.51	1456	1455.6	5.68	1455	-0.07
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P5	1441.0	7.5	1440	1439.2	5.46	1438	-0.14
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P6	1458.0	7.51	1457	1455.2	5.52	1454	-0.21
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P7	1459.6	7.51	1459	1458.4	5.36	1458	-0.07
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P8	1467.6	7.51	1467	1465.4	5.7	1465	-0.14
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P9	1453.6	7.51	1453	1452.4	5.68	1452	-0.07
N75_P(10-20)_S(5-10)_A(40-60)_W37_Wo18_P10	1433.4	7.51	1433	1431.4	5.83	1430	-0.21
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P1	1820.4	7.51	1820	1819.2	6.3	1819	-0.05
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P2	1809.4	7.51	1809	1807.2	5.31	1807	-0.11
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P3	1825.4	7.51	1825	1823.2	5.83	1823	-0.11
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P4	1826.6	7.51	1826	1825.2	5.5	1825	-0.05
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P5	1810.2	7.5	1809	1808.0	5.41	1808	-0.06
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P6	1826.8	7.5	1826	1824.6	6.15	1824	-0.11
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P7	1829.2	7.51	1828	1828.0	5.23	1828	0.0
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P8	1836.6	7.5	1836	1835.2	5.48	1835	-0.05
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P9	1823.0	7.51	1822	1821.4	5.83	1821	-0.05
N75_P(10-20)_S(10-15)_A(20-40)_W37_Wo18_P10	1802.8	7.51	1802	1801.4	5.19	1800	-0.11
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P1	1821.2	7.51	1821	1819.6	5.57	1819	-0.11
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P2	1809.6	7.51	1809	1807.6	5.17	1807	-0.11
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P3	1824.8	7.5	1823	1823.2	5.57	1823	0.0
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P4	1827.0	7.5	1826	1825.0	5.46	1825	-0.05
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P5	1811.0	7.51	1809	1810.0	5.18	1808	-0.06
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P6	1827.4	7.51	1826	1824.8	5.74	1824	-0.11
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P7	1829.6	7.51	1829	1828.0	5.82	1828	-0.05
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P8	1837.2	7.51	1836	1835.6	5.5	1835	-0.05
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P9	1823.6	7.51	1822	1821.0	5.5	1821	-0.05
N75_P(10-20)_S(10-15)_A(40-60)_W37_Wo18_P10	1803.4	7.51	1802	1801.2	5.69	1800	-0.11
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P1	2200.8	7.51	2200	2199.2	6.25	2199	-0.05
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P2	2189.4	7.5	2189	2187.0	5.77	2187	-0.09
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P3	2205.4	7.51	2205	2203.0	5.51	2203	-0.09
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P4	2206.8	7.51	2206	2205.8	5.69	2205	-0.05
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P5	2190.4	7.51	2190	2188.6	5.35	2188	-0.09
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P6	2206.6	7.51	2206	2204.8	5.96	2204	-0.09
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P7	2209.2	7.51	2208	2208.0	5.54	2208	0.0
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P8	2216.4	7.51	2216	2215.2	5.92	2215	-0.05
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P9	2203.0	7.51	2202	2201.4	5.53	2201	-0.05
N75_P(20-30)_S(5-10)_A(20-40)_W37_Wo18_P10	2182.8	7.51	2182	2180.8	6.18	2180	-0.09
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P1	2201.4	7.5	2200	2199.6	5.63	2199	-0.05
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P2	2189.8	7.51	2189	2187.0	5.51	2187	-0.09
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P3	2204.6	7.51	2204	2203.0	6.12	2203	-0.05
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P4	2207.0	7.51	2206	2205.0	5.46	2205	-0.05
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P5	2190.6	7.5	2190	2188.6	5.34	2188	-0.09
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P6	2206.8	7.5	2206	2204.4	5.34	2204	-0.09
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P7	2209.4	7.51	2209	2208.0	5.57	2208	-0.05
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P8	2217.2	7.5	2216	2216.0	5.31	2215	-0.05
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P9	2202.8	7.5	2202	2201.8	6.04	2201	-0.05
N75_P(20-30)_S(5-10)_A(40-60)_W37_Wo18_P10	2183.2	7.51	2182	2181.0	5.88	2180	-0.09
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P1	2571.2	7.51	2571	2569.4	5.7	2569	-0.08
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P2	2559.2	7.51	2559	2557.6	5.39	2557	-0.08
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P3	2575.0	7.51	2574	2573.0	5.67	2573	-0.04
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P4	2576.6	7.51	2576	2575.4	5.76	2575	-0.04
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P5	2561.8	7.51	2561	2558.8	5.05	2558	-0.12
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P6	2576.6	7.5	2576	2574.6	5.89	2574	-0.08
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P7	2578.6	7.51	2578	2578.0	5.58	2578	0.0
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P8	2586.4	7.51	2585	2585.4	5.63	2585	0.0
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P9	2572.6	7.5	2572	2571.4	5.3	2571	-0.04
N75_P(20-30)_S(10-15)_A(20-40)_W37_Wo18_P10	2552.2	7.5	2551	2551.0	5.51	2550	-0.04
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P1	2571.2	7.51	2570	2569.2	5.6	2569	-0.04
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P2	2558.4	7.5	2558	2557.2	5.82	2557	-0.04
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P3	2574.8	7.5	2574	2573.2	5.45	2573	-0.04
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P4	2576.8	7.5	2576	2575.4	5.47	2575	-0.04
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P5	2561.0	7.51	2560	2558.6	5.18	2558	-0.08
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P6	2577.0	7.5	2576	2574.4	5.58	2574	-0.08
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P7	2579.2	7.51	2579	2578.0	6.01	2578	-0.04
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P8	2586.8	7.51	2586	2585.6	5.79	2585	-0.04
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P9	2573.8	7.51	2573	2571.4	5.74	2571	-0.08
N75_P(20-30)_S(10-15)_A(40-60)_W37_Wo18_P10	2553.2	7.5	2552	2551.2	5.3	2550	-0.08

Tabela 7.7: Resultados para instâncias com 100 *jobs*

Instância	LIG			BRKGA			DPR
	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	$C_{\max}$ médio	T médio (s)	$C_{\max}$ melhor	
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P1	1932.6	10.0	1931	1930.4	5.64	1929	-0.1
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P2	1931.0	10.01	1930	1928.6	5.68	1928	-0.1
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P3	1954.2	10.01	1953	1952.8	5.78	1952	-0.05
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P4	1940.2	10.01	1939	1938.0	5.85	1938	-0.05
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P5	1919.0	10.01	1918	1917.2	5.82	1916	-0.1
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P6	1946.2	10.01	1945	1942.6	5.83	1942	-0.15
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P7	1935.0	10.0	1934	1933.0	6.37	1933	-0.05
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P8	1936.6	10.01	1935	1933.8	5.74	1933	-0.1
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P9	1954.4	10.01	1953	1952.0	6.23	1951	-0.1
N100_P(10-20)_S(5-10)_A(20-40)_W50_Wo25_P10	1907.6	10.01	1906	1905.4	5.97	1905	-0.05
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P1	1932.0	10.01	1931	1929.8	5.52	1929	-0.1
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P2	1930.6	10.01	1930	1928.2	5.46	1928	-0.1
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P3	1954.6	10.01	1954	1953.2	5.63	1952	-0.1
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P4	1941.2	10.01	1940	1937.8	6.05	1937	-0.15
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P5	1919.8	10.01	1919	1916.8	5.78	1916	-0.16
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P6	1946.6	10.01	1945	1943.2	5.99	1942	-0.15
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P7	1936.2	10.01	1935	1933.4	6.2	1933	-0.1
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P8	1936.0	10.01	1934	1934.2	5.66	1933	-0.05
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P9	1955.2	10.0	1954	1952.0	6.13	1951	-0.15
N100_P(10-20)_S(5-10)_A(40-60)_W50_Wo25_P10	1908.0	10.01	1907	1905.8	5.87	1905	-0.1
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P1	2426.4	10.01	2425	2424.6	5.45	2424	-0.04
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P2	2425.2	10.01	2424	2423.6	5.57	2423	-0.04
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P3	2449.6	10.01	2448	2447.4	6.14	2447	-0.04
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P4	2434.8	10.01	2433	2433.0	6.45	2432	-0.04
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P5	2414.0	10.01	2413	2411.6	6.02	2411	-0.08
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P6	2440.8	10.01	2440	2437.8	5.42	2437	-0.12
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P7	2430.4	10.01	2428	2428.2	5.6	2428	0.0
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P8	2431.4	10.01	2430	2428.2	5.77	2428	-0.08
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P9	2448.6	10.01	2448	2447.2	6.22	2446	-0.08
N100_P(10-20)_S(10-15)_A(20-40)_W50_Wo25_P10	2403.0	10.01	2402	2400.2	5.92	2400	-0.08
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P1	2427.8	10.0	2427	2424.8	5.86	2424	-0.12
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P2	2426.2	10.01	2425	2423.0	6.06	2423	-0.08
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P3	2450.8	10.0	2450	2447.6	6.13	2447	-0.12
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P4	2435.6	10.01	2435	2433.4	5.6	2433	-0.08
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P5	2414.8	10.01	2414	2411.6	6.17	2411	-0.12
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P6	2441.0	10.01	2440	2438.0	5.71	2437	-0.12
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P7	2430.2	10.01	2430	2428.2	6.34	2428	-0.08
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P8	2430.6	10.01	2429	2428.6	5.35	2428	-0.04
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P9	2449.0	10.0	2448	2446.6	5.62	2446	-0.08
N100_P(10-20)_S(10-15)_A(40-60)_W50_Wo25_P10	2402.4	10.01	2401	2400.2	6.12	2400	-0.04
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P1	2931.6	10.01	2930	2929.8	5.61	2929	-0.03
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P2	2929.8	10.01	2929	2928.4	5.71	2928	-0.03
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P3	2954.8	10.0	2953	2952.8	6.07	2952	-0.03
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P4	2939.8	10.01	2938	2937.0	6.73	2937	-0.03
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P5	2919.0	10.01	2918	2916.8	5.75	2916	-0.07
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P6	2945.8	10.01	2945	2942.4	5.61	2942	-0.1
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P7	2935.4	10.01	2933	2933.2	5.88	2933	0.0
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P8	2936.0	10.01	2935	2934.0	5.68	2933	-0.07
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P9	2953.6	10.01	2953	2951.6	6.12	2951	-0.07
N100_P(20-30)_S(5-10)_A(20-40)_W50_Wo25_P10	2907.0	10.01	2907	2905.0	6.03	2905	-0.07
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P1	2932.2	10.01	2931	2930.8	5.56	2929	-0.07
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P2	2931.6	10.01	2931	2928.4	5.7	2928	-0.1
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P3	2955.6	10.0	2955	2952.4	5.74	2952	-0.1
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P4	2939.4	10.01	2938	2938.0	5.95	2937	-0.03
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P5	2919.6	10.01	2919	2916.4	6.25	2916	-0.1
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P6	2945.8	10.01	2944	2942.8	5.97	2942	-0.07
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P7	2935.4	10.01	2935	2933.4	5.93	2933	-0.07
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P8	2934.8	10.01	2934	2933.8	5.85	2933	-0.03
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P9	2954.8	10.0	2953	2951.6	5.79	2951	-0.07
N100_P(20-30)_S(5-10)_A(40-60)_W50_Wo25_P10	2906.8	10.01	2906	2905.4	5.83	2905	-0.03
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P1	3426.0	10.01	3425	3424.4	5.57	3424	-0.03
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P2	3426.0	10.01	3424	3423.0	5.57	3423	-0.03
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P3	3449.6	10.01	3449	3447.4	5.64	3447	-0.06
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P4	3436.0	10.01	3435	3432.8	6.42	3432	-0.09
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P5	3414.4	10.01	3413	3411.6	6.12	3411	-0.06
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P6	3439.8	10.01	3439	3437.4	5.73	3437	-0.06
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P7	3429.6	10.01	3428	3428.4	5.91	3428	0.0
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P8	3431.6	10.0	3430	3429.2	5.61	3428	-0.06
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P9	3448.8	10.01	3447	3446.6	6.5	3446	-0.03
N100_P(20-30)_S(10-15)_A(20-40)_W50_Wo25_P10	3401.8	10.01	3401	3400.6	5.78	3400	-0.03
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P1	3427.8	10.01	3427	3424.6	5.4	3424	-0.09
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P2	3426.0	10.01	3425	3423.2	5.78	3423	-0.06
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P3	3449.6	10.01	3448	3447.6	6.32	3447	-0.03
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P4	3435.0	10.01	3434	3432.8	5.92	3432	-0.06
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P5	3414.4	10.0	3413	3411.6	6.18	3411	-0.06
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P6	3440.6	10.01	3438	3437.8	5.61	3437	-0.03
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P7	3429.8	10.01	3429	3428.4	6.5	3428	-0.03
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P8	3431.6	10.01	3431	3428.2	5.92	3428	-0.09
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P9	3449.6	10.0	3449	3446.8	5.71	3446	-0.09
N100_P(20-30)_S(10-15)_A(40-60)_W50_Wo25_P10	3402.6	10.01	3402	3400.4	5.89	3400	-0.06

Bibliografia

- ALLAHVERDI, A. The third comprehensive survey on scheduling problems with setup times/costs. *European journal of operational research*, Elsevier, v. 246, n. 2, p. 345–378, 2015.
- BALAS, E.; LENSTRA, J. K.; VAZACOPOULOS, A. The one-machine problem with delayed precedence constraints and its use in job shop scheduling. *Management Science*, INFORMS, v. 41, n. 1, p. 94–109, 1995.
- BEAN, J. C. Genetic algorithms and random keys for sequencing and optimization. *ORSA journal on computing*, INFORMS, v. 6, n. 2, p. 154–160, 1994.
- BRUCKER, P. Scheduling algorithms. *Journal-Operational Research Society*, Springer, v. 50, p. 774–774, 1999.
- CHOOBINEH, F. F.; MOHEBBI, E.; KHOO, H. A multi-objective tabu search for a single-machine scheduling problem with sequence-dependent setup times. *European Journal of Operational Research*, Elsevier, v. 175, n. 1, p. 318–337, 2006.
- FINTA, L.; LIU, Z. Single machine scheduling subject to precedence delays. *Discrete Applied Mathematics*, Elsevier, v. 70, n. 3, p. 247–266, 1996.
- GONÇALVES, J. F.; RESENDE, M. G. Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, Springer, v. 17, n. 5, p. 487–525, 2011.
- GONÇALVES, J. F.; RESENDE, M. G. Random-key genetic algorithms. *Handbook of heuristics*, Springer, p. 1–13, 2016.
- GRAHAM, R. et al. Optimization and approximation in deterministic sequencing and scheduling: a survey. In: HAMMER, P.; JOHNSON, E.; KORTE, B. (Ed.). *Discrete Optimization II*. Elsevier, 1979, (Annals of Discrete Mathematics, v. 5). p. 287–326. Disponível em: <https://www.sciencedirect.com/science/article/pii/S016750600870356X>.
- HOLLAND, J. H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. [S.l.]: MIT press, 1992.
- JULA, P.; KONES, I. Continuous-time algorithms for scheduling a single machine with sequence-dependent setup times and time window constraints in coordinated chains. *International Journal of Production Research*, Taylor & Francis, v. 51, n. 12, p. 3654–3670, 2013.
- KUO, Y.; CHEN, S.-I.; YEH, Y.-H. Single machine scheduling with sequence-dependent setup times and delayed precedence constraints. *Operational Research*, v. 20, n. 2, p. 927–942, Jun 2020. ISSN 1866-1505. Disponível em: <https://doi.org/10.1007/s12351-017-0349-y>.
- LIN, S.-W.; YING, K.-C. Single machine scheduling problems with sequence-dependent setup times and precedence delays. *Scientific Reports*, v. 12, n. 1, p. 9430, Jun 2022. ISSN 2045-2322. Disponível em: <https://doi.org/10.1038/s41598-022-13278-y>.

- LONDE, M. A. et al. Biased random-key genetic algorithms: A review. *European Journal of Operational Research*, Elsevier, v. 321, n. 1, p. 1–22, 2025.
- NAWAZ, M.; JR, E. E. E.; HAM, I. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega*, Elsevier, v. 11, n. 1, p. 91–95, 1983.
- PINEDO, M. L. Design and implementation of scheduling systems: More advanced concepts. In: *Scheduling: Theory, algorithms, and systems*. [S.l.]: Springer, 2016. p. 485–508.
- PRASETYO, H. et al. Survey on applications of biased-random key genetic algorithms for solving optimization problems. In: IEEE. *2015 IEEE international conference on industrial engineering and engineering management (IEEM)*. [S.l.], 2015. p. 863–870.
- RUIZ, R.; MAROTO, C. A comprehensive review and evaluation of permutation flowshop heuristics. *European journal of operational research*, Elsevier, v. 165, n. 2, p. 479–494, 2005.
- SPEARS, W.; JONG, K. D. On the virtues of parameterized uniform crossover. 1995.
- TALBI, E.-G. *Metaheuristics: from design to implementation*. [S.l.]: John Wiley & Sons, 2009.
- TOSO, R. F.; RESENDE, M. G. A c++ application programming interface for biased random-key genetic algorithms. *Optimization Methods and Software*, Taylor & Francis, v. 30, n. 1, p. 81–93, 2015.
- WIKUM, E. D.; LLEWELLYN, D. C.; NEMHAUSER, G. L. One-machine generalized precedence constrained scheduling problems. *Operations Research Letters*, Elsevier, v. 16, n. 2, p. 87–99, 1994.