



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Londrina



**DESENVOLVIMENTO DE UM ALGORITMO DE BUSCA EM
VIZINHANÇA VARIÁVEL PARA O PROBLEMA DE
SEQUENCIAMENTO DE PRODUÇÃO EM JOB SHOPS
FLEXÍVEIS**

Londrina

2019

LUANA FIORI FELICIANO

**DESENVOLVIMENTO DE UM ALGORITMO DE BUSCA EM
VIZINHANÇA VARIÁVEL PARA O PROBLEMA DE
SEQUENCIAMENTO DE PRODUÇÃO EM JOB SHOPS
FLEXÍVEIS**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia de Produção, do Departamento de Engenharia de Produção da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Rafael Henrique Palma Lima.

Londrina

2019

TERMO DE APROVAÇÃO

DESENVOLVIMENTO DE UM ALGORITMO DE BUSCA EM VIZINHANÇA VARIÁVEL PARA O PROBLEMA DE SEQUENCIAMENTO DE PRODUÇÃO EM JOB SHOPS FLEXÍVEIS

POR

LUANA FIORI FELICIANO

Esta Monografia foi apresentada às 16 horas do dia 19 de novembro de 2019 como requisito parcial para obtenção do título de bacharel em ENGENHARIA DE PRODUÇÃO, Universidade Tecnológica Federal do Paraná – Câmpus Londrina. O candidato foi arguido pela Banca Examinadora composta pelos professores relacionados abaixo. Após deliberação, a Banca Examinadora considerou o trabalho: **APROVADO**.

Prof. Me. Bruno Samways Dos Santos
Banca Examinadora

Prof. Dr. Marco Antonio Ferreira
Banca Examinadora

Prof. Dr. Rafael Henrique Palma Lima
Presidente da Banca Examinadora
Orientador

RESUMO

A crescente competitividade no ambiente organizacional tem estimulado uma maior flexibilidade de processos e máquinas por parte das indústrias de manufatura. Isso faz com que ambientes industriais caracterizados como *Job Shop* Flexível, que possuem ordens de produção com diferentes roteiros produtivos e máquinas capazes de realizar diferentes operações, sejam cada vez mais comuns. O sequenciamento de produção em *job shop* flexível é um processo decisório importante a ser realizado pelas fábricas, o qual consiste em designar as operações às máquinas e definir a ordem em que elas serão executadas em casa máquina. Esse processo decisório é denominado como problema de sequenciamento de *job shop* flexível (FJSP), que é um problema classificado como NP-Difícil na literatura científica, sendo de alta complexidade de resolução em tempos computacionais curtos. Para resolução deste problema, este trabalho propõe um algoritmo baseado na metaheurística *Variable Neighborhood Search* (VNS). Sendo assim, quatro procedimentos de busca em vizinhança variável foram desenvolvidos em quatro diferentes configurações, a fim de se solucionar o FJSP, minimizando o *makespan*. Para testes do algoritmo, 198 instâncias da literatura foram utilizadas, além da aplicação em um caso ilustrativo para análise de viabilidade de aplicação em empresas reais. Os resultados obtidos demonstram a capacidade do algoritmo em alcançar soluções competitivas quando comparadas com as melhores soluções conhecidas na literatura. Além disso, o caso ilustrativo exemplifica como o método proposto poderia se adequar a uma situação real, apresentando resultados melhores do que soluções aleatórias.

Palavras-chave: Sequenciamento de Produção; *Job Shop* Flexível; Busca em Vizinhança Variável.

ABSTRACT

The growing competitiveness in the organizational environment has stimulated the increase in the flexibility of process and machines by manufacturing industries. In order to comply with this, industrial environment characterized as Flexible *Job Shop*, which have production orders with different production routings and machines capable of process different operations, has become more common. The flexible job shop scheduling is an important decision making process for factories, which consists of assign the operations to machines and define a sequence in which they will be processed on each machine. This decision making process, called Flexible Job Shop Problem (FJSP), is a problem classified as NP-hard in the scientific literature, with high resolution complexity in short computational times. To solve this problem, this paper proposes an algorithm based on a metaheuristic called Variable Neighborhood Search. Therefore, four local search procedures with systematic neighborhood changes were developed in four different configurations in order to solve the FJSP minimizing the makespan. For algorithm tests, 198 benchmark problems were used, besides the application in an illustrative case, in order to analyze the feasibility of application in real companies. The results obtained demonstrate the ability of the algorithm to reach competitive solutions when compared to the best known solutions in the literature. Furthermore, the illustrative case exemplifies how the proposed method could be appropriate for a real situation with better results than random solutions.

Key-words: Production Scheduling; Flexible Job Shop Problem; Variable Neighborhood Search.

LISTA DE ILUSTRAÇÕES

Figura 1 - Níveis de Planejamento da Produção	14
Figura 2 - Gráfico de <i>Gantt</i> da solução	20
Figura 3 - Estrutura básica do VNS	26
Figura 4 - Representação da Solução	29
Figura 5 - Exemplo da Estrutura de Vizinhança V_1	32
Figura 6 - Exemplo da Estrutura de Vizinhança V_2	33
Figura 7 - Exemplo da Estrutura de Vizinhança V_3	34
Figura 8 - Gráfico comparativo de quantidade de máquinas <i>versus gap</i> das instâncias do grupo <i>edata</i> de Hurink <i>et. al</i> (1994)	43
Figura 9 - Gráfico comparativo de quantidade de máquinas <i>versus gap</i> das instâncias do grupo <i>rdata</i> de Hurink <i>et. al</i> (1994)	43
Figura 10 - Gráfico comparativo de quantidade de máquinas <i>versus gap</i> das instâncias do grupo <i>vdata</i> de Hurink <i>et. al</i> (1994)	44
Figura 11 - Gráfico comparativo de quantidade de <i>jobs versus gap</i> das instâncias do grupo <i>edata</i> de Hurink <i>et. al</i> (1994)	45
Figura 12 - Gráfico comparativo de quantidade de <i>jobs versus gap</i> das instâncias do grupo <i>rdata</i> de Hurink <i>et. al</i> (1994)	45
Figura 13 - Gráfico comparativo de quantidade de <i>jobs versus gap</i> das instâncias do grupo <i>vdata</i> de Hurink <i>et. al</i> (1994)	46
Figura 14 - Gráfico comparativo de tamanho da instância <i>versus gap</i> das instâncias do grupo <i>edata</i> de Hurink <i>et. al</i> (1994)	46
Figura 15 - Gráfico comparativo de tamanho da instância <i>versus gap</i> das instâncias do grupo <i>rdata</i> de Hurink <i>et. al</i> (1994)	47
Figura 16 - Gráfico comparativo de tamanho da instância <i>versus gap</i> das instâncias do grupo <i>vdata</i> de Hurink <i>et. al</i> (1994)	47
Figura 17 - Gráfico comparativo de quantidade de máquinas <i>versus</i> tempo médio do grupo <i>edata</i> de Hurink <i>et. al</i> (1994)	49
Figura 18 - Gráfico comparativo de quantidade de máquinas <i>versus</i> tempo médio do grupo <i>rdata</i> de Hurink <i>et. al</i> (1994)	49
Figura 19 - Gráfico comparativo de quantidade de máquinas <i>versus</i> tempo médio do grupo <i>vdata</i> de Hurink <i>et. al</i> (1994)	50

Figura 20 - Gráfico comparativo de quantidade de <i>jobs versus</i> tempo médio do grupo <i>edata</i> de Hurink <i>et. al</i> (1994).....	50
Figura 21 - Gráfico comparativo de quantidade de <i>jobs versus</i> tempo médio do grupo <i>rdata</i> de Hurink <i>et. al</i> (1994).....	51
Figura 22 - Gráfico comparativo de quantidade de <i>de jobs versus</i> tempo médio do grupo <i>vdata</i> de Hurink <i>et. al</i> (1994).....	51
Figura 23 - Gráfico comparativo de quantidade de operações <i>versus</i> tempo médio do grupo <i>edata</i> de Hurink <i>et. al</i> (1994).....	52
Figura 24 - Gráfico comparativo de quantidade de operações <i>versus</i> tempo médio do grupo <i>rdata</i> de Hurink <i>et. al</i> (1994).....	52
Figura 25 - Gráfico comparativo de quantidade de operações <i>versus</i> tempo médio do grupo <i>vdata</i> de Hurink <i>et. al</i> (1994).....	53

LISTA DE TABELAS

Tabela 1 - Demonstração de um problema FJSP	19
Tabela 2 - Métodos de resolução do FJSP	23
Tabela 3 - Demonstração 2 de um problema FJSP	29
Tabela 4 - <i>Score</i> dos resultados obtidos com o grupo de instâncias <i>rdata</i> do autor Hurink <i>et al.</i> (1994).....	36
Tabela 5 - Resumo dos resultados obtidos pelo <i>Hill-Climbing</i> com o grupo de instâncias <i>rdata</i> do autor Hurink <i>et al.</i> (1994).....	37
Tabela 6 - Configurações do algoritmo proposto.....	38
Tabela 7 - Resumo dos resultados obtidos com o grupo de instâncias <i>edata</i> do autor Hurink <i>et al.</i> (1994).....	40
Tabela 8 - Resumo dos resultados obtidos com o grupo de instâncias <i>rdata</i> do autor Hurink <i>et al.</i> (1994).....	40
Tabela 9 - Resumo dos resultados obtidos com o grupo de instâncias <i>vdata</i> do autor Hurink <i>et al.</i> (1994).....	41
Tabela 10 - <i>Score</i> de cada configuração para o melhor resultado	42
Tabela 11 - Tempo médio (segundos) de execução por iteração do algoritmo.....	48
Tabela 12 - Exemplo de instância com base no processo produtivo em estudo	54
Tabela 13 - Resultados obtidos através da instância exemplo	55

LISTA DE SIGLAS E ABREVIATURAS

CR	<i>Critical Ratio</i>
DS	<i>Dynamic Slack</i>
EDD	<i>Earliest Due Date</i>
FIFO	<i>First In First Out</i>
FSFO	<i>First in the System, First Out</i>
FJSP	<i>Flexible Job Shop Problem</i>
MRP	<i>Material Requirement Planning</i>
SOT	<i>Shortest Operation Time</i>
SOT1	<i>Shortes Operation Time 1</i>
SS	<i>Static Slack</i>
UTFPR	Universidade Tecnológica Federal do Paraná
VND	<i>Variable Neighborhood Descent</i>
VNS	<i>Variable Neighborhood Search</i>

SUMÁRIO

1. INTRODUÇÃO	11
2. REFERENCIAL TEÓRICO	13
2.1. SEQUENCIAMENTO DE PRODUÇÃO.....	13
2.2. SEQUENCIAMENTO DE PRODUÇÃO EM SISTEMAS <i>JOB SHOP</i>	16
2.3. SEQUENCIAMENTO DE PRODUÇÃO EM SISTEMAS <i>JOB SHOP</i> FLEXÍVEL	17
2.4. MÉTODOS DE RESOLUÇÃO.....	22
2.5. <i>VARIABLE NEIGHBORHOOD SEARCH</i> (VNS)	26
3. ALGORITMO PROPOSTO	28
3.1. REPRESENTAÇÃO DA SOLUÇÃO.....	28
3.2. AVALIAÇÃO DA SOLUÇÃO	30
3.3. SOLUÇÃO INICIAL	30
3.4. ESTRUTURAS DE VIZINHANÇA	31
3.4.1. Estrutura de Vizinhança <i>V1</i>	31
3.4.2. Estrutura de Vizinhança <i>V2</i>	32
3.4.3. Estrutura de Vizinhança <i>V3</i>	33
3.4.4. Estrutura de Vizinhança <i>V4</i>	33
4. RESULTADOS E DISCUSSÃO	35
4.1. TESTE PRELIMINAR DAS ESTRUTURAS DE VIZINHANÇA.....	36
4.2. CONFIGURAÇÃO DO VNS	38
4.3. ANÁLISE DA QUALIDADE DAS SOLUÇÕES OBTIDAS (<i>MAKESPAN</i>)	40
4.4. ANÁLISE DO TEMPO DE EXECUÇÃO DO ALGORITMO	47
5. CASO ILUSTRATIVO	54
6. CONCLUSÕES	57
REFERÊNCIAS	59
APÊNDICE A – TABELA DE RESULTADOS DE <i>MAKESPAN</i> OBTIDOS NO TESTE DE VIZINHANÇAS COM O GRUPO DE INSTÂNCIAS <i>RDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994).....	62
APÊNDICE B - TABELA DE RESULTADOS RELACIONADOS À EXECUÇÃO DO <i>HILL-CLIMBING</i> OBTIDOS NOS TESTES DAS VIZINHANÇAS <i>V1</i> E <i>V2</i> COM O GRUPO DE INSTÂNCIAS <i>RDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994).....	65
APÊNDICE C - TABELA DE RESULTADOS RELACIONADOS À EXECUÇÃO DO <i>HILL-CLIMBING</i> OBTIDOS NOS TESTES DAS VIZINHANÇAS <i>V3</i> E <i>V4</i> COM O GRUPO DE INSTÂNCIAS <i>RDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994).....	68
APÊNDICE D - TABELA DE RESULTADOS DA CONFIGURAÇÃO 1 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>EDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	71

APÊNDICE E - TABELA DE RESULTADOS DA CONFIGURAÇÃO 2 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>EDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	74
APÊNDICE F - TABELA DE RESULTADOS DA CONFIGURAÇÃO 3 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>EDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	77
APÊNDICE G - TABELA DE RESULTADOS DA CONFIGURAÇÃO 4 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>EDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	80
APÊNDICE H - TABELA DE RESULTADOS DA CONFIGURAÇÃO 1 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>RDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	83
APÊNDICE I - TABELA DE RESULTADOS DA CONFIGURAÇÃO 2 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>RDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	86
APÊNDICE J - TABELA DE RESULTADOS DA CONFIGURAÇÃO 3 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>RDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	89
APÊNDICE K - TABELA DE RESULTADOS DA CONFIGURAÇÃO 4 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>RDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	92
APÊNDICE L - TABELA DE RESULTADOS DA CONFIGURAÇÃO 1 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>VDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	95
APÊNDICE M - TABELA DE RESULTADOS DA CONFIGURAÇÃO 2 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>VDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	98
APÊNDICE N - TABELA DE RESULTADOS DA CONFIGURAÇÃO 3 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>VDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	101
APÊNDICE O - TABELA DE RESULTADOS DA CONFIGURAÇÃO 4 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS <i>VDATA</i> DO AUTOR HURINK <i>ET. AL</i> (1994)	104

1. INTRODUÇÃO

O sequenciamento de produção é um processo decisório crucial que influencia na eficiência dos sistemas operacionais, visto que tem como finalidade alocar recursos finitos ao longo do tempo para realização de um determinado conjunto de tarefas. Em âmbito de economia empresarial é comumente aplicado em fábricas de manufatura e em sistemas de serviços. Incorpora um problema com muitas tarefas, decompostas em operações, que precisam ser designadas a recursos específicos ou a um conjunto de recursos disponíveis de forma a atender um determinado objetivo da melhor forma possível (KUPFAHL, 2016).

Em indústrias de manufatura, o nível de complexidade para se encontrar a melhor sequência das tarefas se dá pelo ambiente industrial, as restrições de seus processos e a meta de desempenho adotada a ser alcançada. Um dos problemas de sequenciamento considerados de alta complexidade é o problema denominado *Flexible Job Shop Problem* (FJSP), no qual há um conjunto de tarefas divididas em operações as quais precisam ser sequenciadas, obedecendo a certa ordem de processamento, e que podem ser alocadas em qualquer máquina dentre um conjunto de máquinas factíveis (PEZZELLA, 2008).

A alta complexidade computacional do FJSP, considerada *NP-hard*, faz com que métodos exatos baseados em programação linear não sejam adequados (ZHANG; GAO; SHI, 2011). A obtenção de soluções ótimas para instâncias de tamanho médio ou grande se tornaria inviável devido aos longos tempos computacionais envolvidos. Isso justifica a adoção de métodos heurísticos e metaheurísticos para sua resolução, os quais, em tempos razoáveis, fornecem boas soluções. A metaheurística *Variable Neighborhood Search* (VNS) tem obtido sucesso na resolução deste tipo de problema. Devido ao fato de ser baseada em uma mudança sistemática de estruturas de vizinhança em seu processo de busca de soluções, possui a capacidade de escapar de ótimos locais (AMIRI; ZANDIEH; YASDANI, 2010).

Diante disto, o objetivo geral deste trabalho é desenvolver um algoritmo de busca em vizinhança variável capaz de obter soluções competitivas para o FJSP em tempos computacionais curtos. Assim, os objetivos específicos são:

- Realizar um levantamento acerca métodos de resolução do problema de sequenciamento de produção em *job shops* flexíveis;
- Formular um algoritmo baseado na metaheurística *Variable Neighborhood Search*, através de programação computacional, para resolução do FJSP;
- Testar o algoritmo em instâncias de benchmark, comparando os resultados obtidos com os resultados apresentados pela literatura;
- Demonstrar a aplicação do algoritmo em um caso ilustrativo com processo de produção caracterizado como *job shop* flexível;

Por se tratar de um problema de sequenciamento de produção de difícil resolução e que se enquadra nos processos produtivos atuais, os quais buscam maior flexibilidade de seus processos para estar à frente perante a concorrência, o desenvolvimento de métodos que o solucionem em tempo ágil é de grande relevância para as organizações, justificando a presente pesquisa. Para Bekkar *et. al* (2016), em um problema *NP-hard*, a velocidade computacional supera a qualidade da solução, visto que em um contexto de manufatura o ganho de tempo de produção representa um fator essencial. Esta é uma razão pela qual as heurísticas são utilizadas para fornecer soluções, próximas da ótima, que reduzam o tempo de produção em curtos espaços de tempo. Outro fator de justificativa, em âmbito científico, é o crescente interesse em desenvolvimento de técnicas de resolução do problema, visto a alta complexidade deste. Em um estudo realizado por Chaudry e Khan (2016), 191 artigos a cerca do problema foram abordados, apresentando, no total, 197 diferentes técnicas de resolução.

Dessa forma, os próximos capítulos serão divididos da seguinte forma: no Capítulo 2 será apresentado uma revisão da bibliografia acerca do problema e seus métodos de resolução. No Capítulo 3 será apresentado o algoritmo proposto para resolução do problema, e no Capítulo 4 serão discutidos os resultados obtidos. O Capítulo 5 contempla o caso ilustrativo acerca de um processo produtivo do tipo *Job Shop Flexível*. A conclusão do trabalho será apresentada no Capítulo 6.

2. REFERENCIAL TEÓRICO

2.1. SEQUENCIAMENTO DE PRODUÇÃO

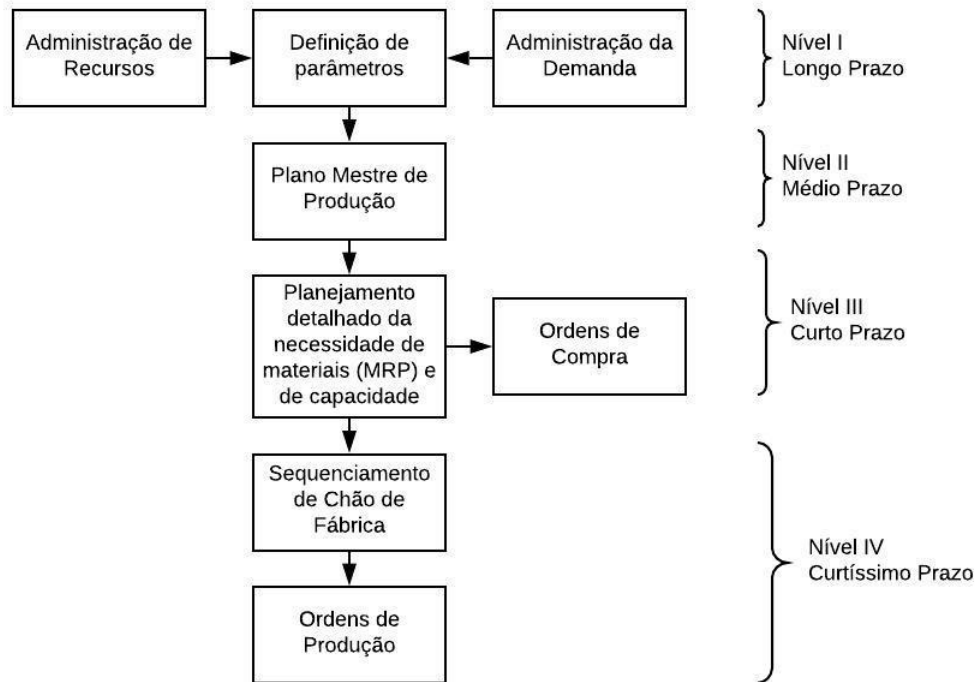
Em meio a uma limitação de tempo existente no cotidiano das pessoas, as atividades a serem realizadas ao longo do dia começaram a ser programadas por elas a fim de cumpri-las da melhor maneira possível, obedecendo-se o tempo disponível (SULE, 2008).

Em um ambiente industrial, isso não é diferente. Dado um local com diversos recursos, tarefas, restrições e objetivos, o sequenciamento de produção é uma das estratégias utilizadas para a melhoria do desempenho das organizações. Para Pinedo (2016), o sequenciamento de produção é um processo de tomada de decisão cotidiano utilizado por indústrias de manufatura e de serviços a fim de se alocar recursos para realização de tarefas, em um dado período de tempo, com o intuito de otimização de determinados objetivos.

Segundo Sule (2008), o planejamento e sequenciamento da produção são dois tópicos importantes que proporcionam o aumento da eficiência em processos de manufatura e a melhora da eficácia no serviço prestado ao cliente. O planejamento da produção decide o que, quando e quanto será produzido para atender as necessidades do cliente. O sequenciamento determina como atingir as metas estabelecidas no planejamento quando os recursos são limitados e, caso as metas não possam ser alcançadas, estabelece como melhor definir novos objetivos que são ótimos de acordo com os recursos disponíveis.

De acordo com a metodologia de Planejamento Hierárquico da Produção, adotada por Corrêa e Corrêa (2019), o sequenciamento da produção é um subproblema do problema de planejamento da produção. Esse planejamento ocorre em quatro níveis, que são determinados de acordo com o horizonte de planejamento, representados na Figura 1.

Figura 1 - Níveis de Planejamento da Produção



Fonte: Adaptado de Sule (2008)

Conforme Figura 1, o primeiro nível, denominado de planejamento de vendas e operações, o qual ocorre em longo prazo e agrega os produtos em famílias ou linhas de produto e os recursos em sistemas produtivos. O segundo nível do planejamento estabelece o Programa Mestre de Produção, em médio prazo. Em seguida, a um terceiro nível, em curto prazo, é realizado o MRP que, em concordância com o planejamento mestre de produção definido nos níveis anteriores, define os materiais necessários e a capacidade produtiva para a produção. É nesta fase em que são emitidas ordens de compra de insumos necessários. O Sequenciamento é o último nível, que a curtíssimo prazo, define a sequencia das atividades a serem realizadas e faz a alocação dessas atividades nos recursos disponíveis, de tal forma a atender as restrições e objetivos delimitados nos nível anterior (CORRÊA; CORRÊA, 2019).

É neste momento em que as ordens de produção são liberadas para o chão de fábrica contendo informações necessárias aos setores produtivos, como código e especificação do item, quantidade, a prioridade das operações e datas de início e fim, para que os responsáveis possam executar as atividades (TUBINO, 2017).

Pinedo (2016) aborda que os fatores que afetam diretamente o processo de sequenciamento se baseiam nas decisões tomadas nos níveis hierárquicos de planejamento anteriores a ele, bem como, a eventos inesperados ocorridos em chão de fábrica como quebra de máquina e tempos de processamento das operações maiores do que o esperado. Dessa forma, o fluxo de informações entre todos os setores e processos relacionados à etapa de sequenciamento deve ocorrer de forma integrada e eficiente para que a produção ocorra conforme os objetivos estabelecidos e, caso haja necessidade de mudanças e ressequenciamentos, a resposta possa ser realizada da forma mais rápida possível.

O conjunto de decisões do sequenciamento de produção, então, norteia a definição e detalhamento das atividades (ordens de produção) a serem desempenhados, o momento (tempo de acordo com as ordens de processamento definidas) em que estas devem ser realizadas, e quais os recursos (máquinas, equipamentos, mão-de-obra, insumos, instalações, entre outros) a serem utilizados. Os fatores e as restrições correlacionadas a esse conjunto de decisões variam conforme o tipo de sistema produtivo, o que pode ocasionar um problema de sequenciamento de alta complexidade, que dificulta a solução por métodos intuitivos. Faz-se necessário a utilização de métodos baseados em formulação matemática, como simulação e algoritmos (CORRÊA; GIANESI; CAON, 2019).

Muitos estudos mostram que a utilização de métodos heurísticos para o sequenciamento de problemas considerados de alta complexidade computacional é recorrente devido ao fato da possibilidade de obtenção de bons resultados em curtos espaços de tempo, fator prescindível em um ambiente organizacional. Segundo Pezella *et al.* (2008) métodos exatos de otimização não são aplicáveis a instâncias com grande número de máquinas e tarefas, ao passo que as heurísticas oferecem resultados eficazes para a maioria das instância em tempos razoáveis.

Os modelos matemáticos para resolução dos problemas de sequenciamento levam em consideração as variáveis e restrições definidas e buscam a minimização ou maximização de um objetivo. A função objetivo varia conforme o objetivo de desempenho adota pela empresa.

O modelo do problema varia, também, de acordo ao ambiente industrial a que está inserido. De acordo com Pinedo (2016), os modelos de sequenciamento

são caracterizados em: máquina única, máquinas idênticas em paralelo, máquinas em paralelo com diferentes velocidades, máquinas não relacionadas em paralelo, *flow shop*, *flexible flow shop*, *job shop*, *flexible job shop* e *open shop*. Além disso, é categorizado, também, como um problema determinístico, em que todas as condições do sistema são conhecidas, e estocásticas, no qual as incertezas levam a necessidade de utilização de distribuições de probabilidade.

2.2. SEQUENCIAMENTO DE PRODUÇÃO EM SISTEMAS *JOB SHOP*

Um sistema de produção classificado como *job shop* consiste em um sistema em que contém múltiplas operações a serem processadas podem passar ou não por múltiplos centros de trabalho. O problema de sequenciamento em *job shop* leva em consideração as variáveis deste tipo de sistema, que pode ser em termos de ordens, recursos e operações. As ordens de produção podem apresentar diferentes datas de entrega ao cliente e podem conter mais de um roteiro de produção viável. Os roteiros de produção podem possuir diferentes produtividades, considerando-se que exista mais de uma máquina que desempenha uma mesma função, mas que possuem eficiências diferentes. Os recursos (sejam eles máquinas, mão-de-obra ou matéria-prima) podem estar indisponíveis e as operações podem necessitar passar por um mesmo centro de trabalho mais uma vez (CORRÊA; CORRÊA, 2019).

Para resolução de um problema de sequenciamento, existem regras de sequenciamento que definem as prioridades das ordens a serem processadas. As regras são classificadas em diferentes categorias, que podem ser estáticas, não se alteram mesmo que haja mudanças no sistema, e dinâmicas, que se alteram de acordo com as mudanças. Podem, também, ser divididas em regras de prioridades simples, que abrangem também combinação de regras de prioridades simples e regras com índices ponderados, e regras heurísticas sofisticadas, que envolvem inteligência artificial, algoritmos e simulações (TUBINO, 2017).

De acordo com Corrêa e Corrêa (2019), as regras simples mais usuais para definição de prioridades em problemas de sequenciamento em *job shop* são:

- FIFO (*First in First Out*): a primeira tarefa que chega a um centro de trabalho é a primeira tarefa a ser processada;
- FSFO (*First in the System, First Out*): a primeira tarefa que chega ao chão de fábrica é a primeira tarefa a ser processada;
- SOT (*Shortest Operation Time*): a tarefa de menor tempo de processamento é a primeira a ser processada;
- SOT1: mesma lógica do SOT, mas com um tempo limite de espera;
- EDD (*Earliest Due Date*): a tarefa com menor prazo de entrega é a primeira a ser processada;
- SS (*Static Slack*): de acordo com a folga estática (diferença entre o tempo da data de entrega e o tempo faltante de processamento);
- DS (*Dynamic Slack*): de acordo com a folga dinâmica (divisão do SS pelo número de operações a serem executadas);
- CR (*Critical Ratio*): de acordo com o índice crítico (divisão do tempo até a data de entrega e o tempo total de processamento faltante).

Segundo Rigo (2018), essas regras de sequenciamento consideradas simples podem ser utilizadas para formulação de soluções iniciais em algoritmos mais complexos, a fim de se obter um sequenciamento inicial com resultados melhores quando comparados a uma solução inicial totalmente aleatória.

2.3. SEQUENCIAMENTO DE PRODUÇÃO EM SISTEMAS *JOB SHOP* FLEXÍVEL

Para estar à frente da concorrência e conseguir atender as exigências do mercado, as indústrias precisaram não só automatizar e flexibilizar suas máquinas, como também flexibilizar seus processos para que pudessem ter uma maior eficiência produtiva (FATTAHI *et al.*, 2007). É neste contexto que surge o sistema de produção do tipo *job shop* flexível.

Segundo Chaudry e Khan (2016), o modelo *job shop* flexível é uma extensão do *job shop*, em que uma mesma operação pode ser processada por mais de uma máquina de um determinado conjunto de máquinas. Ainda de acordo com os

autores, há algumas suposições a serem consideradas no problema *job shop* flexível:

- Todas as máquinas e todos os *jobs* estão disponíveis no tempo zero;
- Cada operação pode ser processada apenas por uma máquina por vez;
- Os *jobs* são independentes uma das outras (não há restrições de precedência entre elas);
- Interrupções não são permitidas. Uma vez que o processamento de uma operação é iniciado, não poderá ser interrompida;
- Tempos de transporte e de *setup* são incluídos no tempo de processamento da operação.

O problema do tipo *job shop* flexível é dividido em dois subproblemas: o problema de roteamento, que determina a máquina, dentre um conjunto de máquinas viáveis, que irá processar cada operação; e o problema de sequenciamento, que sequencia as operações designadas nas máquinas selecionadas a fim de se obter um sequenciamento viável que minimize um objetivo já pré-definido. Por se tratar de um sistema que reduz as restrições das máquinas, mas que aumenta o número de soluções factíveis, o FJSP (*flexible job shop problem*) é um problema considerado de alta complexidade (ZHANG *et al.*, 2018).

De acordo com Chaudry e Khan (2016), o FJSP é um problema clássico de pesquisa operacional que tem sido considerado desde os anos 50 um difícil problema de otimização combinatória (*NP-hard*), o que dificulta a obtenção de soluções ótimas do problema.

O FJSP se baseia na ordenação de n operações em m máquinas. Há um conjunto de máquinas $Ma = \{M_1, M_2, \dots, M_m\}$ e um conjunto de tarefas $J = \{J_1, J_2, \dots, J_n\}$. Para cada tarefa (*job*) J_i , existe uma ordem de n_i operações a serem processadas. Para cada operação O_{ij} (operação j da tarefa i) existe um subconjunto de máquinas Ma_{ij} pertencentes ao conjunto Ma , disponíveis para processamento (AMIRI; ZANDIEH; YASDANI, 2010). Um exemplo de um problema de sequenciamento em *job shop* flexível é representado na Tabela 1.

Tabela 1 - Demonstração de um problema FJSP

Job (J_i)	Operação (O_{ij})	M_1	M_2	M_3
1	1	5	4	-
1	2	8	-	8
1	3	2	3	5
2	1	-	-	10
2	2	6	9	-

Fonte: Adaptado de Rigo (2018).

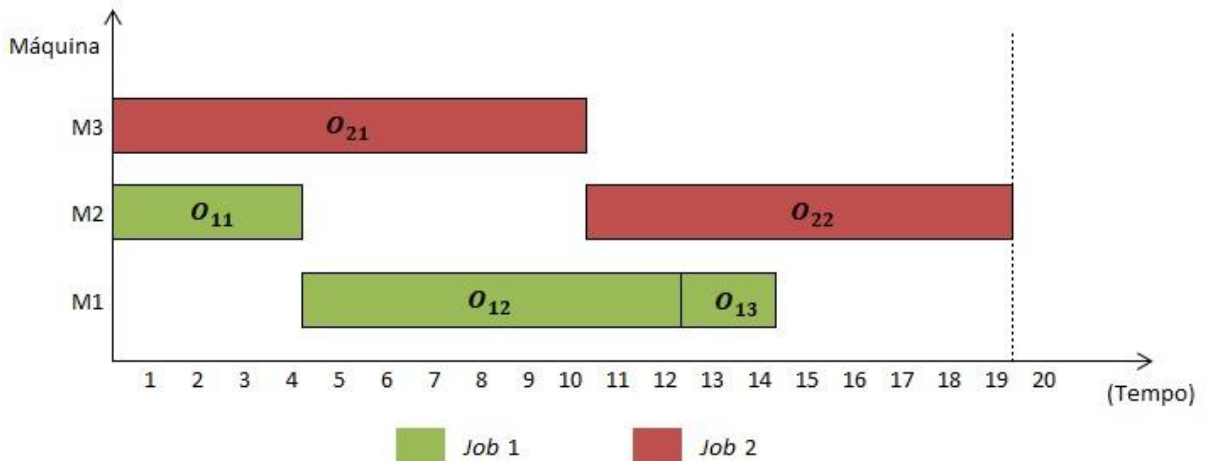
Conforme Tabela 1, o conjunto de tarefas a serem realizadas e o conjunto de máquinas existentes são, respectivamente, $J_a = \{J_1, J_2\}$ e $Ma = \{M_1, M_2, M_3\}$. Ou seja, há duas tarefas (*jobs*) a serem realizadas e três máquinas no total. A tarefa 1 possui três operações, enquanto que a tarefa 2 possui duas operações, totalizando cinco operações ($O_{11}, O_{12}, O_{13}, O_{21}, O_{22}$) a serem sequenciadas e alocadas em três máquinas. A operação O_{11} (operação 1 do *job* 1), por exemplo, pode ser processada tanto pela máquina M_1 , com um tempo de processamento de 5 unidades de tempo, como pela máquina M_2 , com um tempo de processamento de 4 unidades de tempo. A máquina M_3 não é uma máquina disponível para realização da operação, sendo assim, o conjunto de máquinas disponíveis para a operação O_{11} é $Ma_{11} = \{M_1, M_2\}$.

Ainda de acordo com o exemplo apresentado na Tabela 1, a operação O_{12} só poderá ser iniciada caso a operação O_{11} já esteja finalizada e a operação O_{13} somente poderá ser processada se a operação O_{12} já foi processada. As operações da tarefa 2 seguem o mesmo critério, porém, como não há restrições de precedência entre os *Jobs*, estes não precisam ser processados em ordem, ou seja, pode-se processar, por exemplo, a operação O_{21} antes da operação O_{11} . A resolução do problema consiste em sequenciar as cinco operações e alocá-las nas máquinas disponíveis, de forma a maximizar ou minimizar uma determinada função objetivo.

As funções objetivo conhecidas são abordadas por Chaudry e Khan (2016) como sendo: *makespan* (tempo máximo de conclusão das tarefas), tempo médio de conclusão, tempo máximo de fluxo, tempo médio de fluxo, atraso total, atraso médio, atraso ponderado total, atraso máximo, número de tarefas (*jobs*) atrasadas, carga total de trabalho das máquinas e carga da máquina crítica.

Uma possível solução do problema da tabela 1 é $S = \{(O_{21}, M_3), (O_{11}, M_2), (O_{12}, M_1), (O_{22}, M_2), (O_{13}, M_1)\}$. A figura 2 contém a representação da solução através de um gráfico de *Gantt*.

Figura 2 - Gráfico de *Gantt* da solução



Fonte: Do autor, 2019.

Considerando a figura 2, a primeira operação sequenciada é a operação O_{21} , a qual é alocada na máquina M_3 . Como nenhuma operação foi alocada à máquina 3 anteriormente, o processamento da operação O_{21} começa no tempo 0. A mesma consideração a respeito do início de processamento é feita na operação O_{11} (segunda operação sequenciada) realizada na máquina M_2 . Já a operação O_{12} , terceira operação sequenciada, é iniciada no tempo 4, pois, mesmo que a máquina M_1 esteja disponível no tempo 0, a operação só pode ser processada quando a O_{11} esteja finalizada. A mesma situação ocorre com a próxima operação sequenciada, O_{22} . Como a operação O_{21} é finalizada somente no tempo 10, o processamento da operação O_{22} só pode ser iniciado no tempo 10, e não no tempo 4, que é tempo disponível para a máquina M_2 . Já a operação O_{13} , como o tempo disponível da máquina e o tempo de finalização da O_{12} é 12, é neste momento que a operação pode ser iniciada.

Segundo Fattahi *et al.* (2007), o problema FJSP pode ser formulado seguindo um modelo de programação linear mista, denominado *MILP*, como representado a seguir:

Minimizar

C_{max}

$$\begin{aligned}
\text{Sujeito a} \quad C_{max} &\geq t_{j,h_j} + Ps_{j,h_j} && \text{para } j = 1, \dots, n && (1) \\
\sum_i y_{i,j,h} \cdot p_{i,j,h} &&& \text{para } j = 1, \dots, n; h = 1, \dots, h_j && (2) \\
t_{j,h} + Ps_{j,h} &\leq t_{j,h+1} && \text{para } j = 1, \dots, n; h = 1, \dots, h_{j+1} && (3) \\
Tm_{i,k} + Ps_{j,h} \cdot x_{i,j,h,k} &\leq Tm_{i,k+1} && \text{para } i = 1, \dots, m; j = 1, \dots, n; \\ &&& h = 1, \dots, h_j; k = 1, \dots, k_{i-1} && (4) \\
Tm_{i,k} &\leq t_{j,h} + (1 - x_{i,j,h,k}) \cdot L && \text{para } i = 1, \dots, m; j = 1, \dots, n; \\ &&& h = 1, \dots, h_j; k = 1, \dots, k_i && (5) \\
Tm_{i,k} + (1 - x_{i,j,h,k}) \cdot L &\geq t_{j,h} && \text{para } i = 1, \dots, m; j = 1, \dots, n; \\ &&& h = 1, \dots, h_j; k = 1, \dots, k_i && (6) \\
y_{i,j,h} &\leq a_{i,j,h} && \text{para } i = 1, \dots, m; j = 1, \dots, n; \\ &&& h = 1, \dots, h_j && (7) \\
\sum_j \sum_h x_{i,j,h,k} &= 1 && \text{para } i = 1, \dots, m; k = 1, \dots, k_i && (8) \\
\sum_i y_{i,j,h} &= 1 && \text{para } j = 1, \dots, n; h = 1, \dots, h_j && (9) \\
\sum_k x_{i,j,h,k} &= y_{i,j,h} && \text{para } i = 1, \dots, m; j = 1, \dots, n; \\ &&& h = 1, \dots, h_j && (10) \\
t_{j,h} &\geq 0 && \text{para } j = 1, \dots, n; h = 1, \dots, h_j && (11) \\
Ps_{j,h} &\geq 0 && \text{para } j = 1, \dots, n; h = 1, \dots, h_j && (12) \\
Tm_{i,k} &\geq 0 && \text{para } i = 1, \dots, m; k = 1, \dots, k_i && (13) \\
x_{i,j,h,k} &\in \{0,1\} && \text{para } i = 1, \dots, m; j = 1, \dots, n; \\ &&& h = 1, \dots, h_j; k = 1, \dots, k_i && (14) \\
y_{i,j,h} &\in \{0,1\} && \text{para } i = 1, \dots, m; j = 1, \dots, n; \\ &&& h = 1, \dots, h_j && (15)
\end{aligned}$$

A equação (1) estabelece o *makespan*, o qual a minimização é a função objetivo do problema. A restrição (2) determina o cálculo do tempo de processamento da operação na máquina selecionada. A restrição (3) impõe que cada tarefa (*job*) tem uma sequencia de operações a serem processadas. A restrição (4) limita a máquina a processar uma operação de cada vez. As restrições (5) e (6) determinam que cada operação só pode ser iniciada caso a máquina esteja

em seu estado ocioso e a operação precedente da tarefa (*job*) em questão já foi processada. A restrição (7) estabelece a capacidade da máquina para cada operação. A restrição (8) garante a designação da operação em uma máquina. A restrição (9) e (10) delimitam que uma operação pode ser processada em apenas uma máquina e com apenas uma prioridade. Já as restrições (11), (12), (13), (14) e (15) representam as variáveis do FJSP.

2.4. MÉTODOS DE RESOLUÇÃO

O sequenciamento de produção em *job shop* flexível, por se tratar de um problema de alta complexidade, tem ganhado destaque no desenvolvimento de técnicas de resolução por diversos autores. Para Chaudry e Khan (2016), essa classe de problema vem sendo amplamente abordada ao longo dos últimos 25 anos. Segundo os autores, diversos métodos envolvendo desde técnicas matemáticas a metaheurísticas foram desenvolvidos, como por exemplo: algoritmos evolutivos, otimização por colônia de formigas, otimização por enxame de partículas, entre outros.

Dentre outras técnicas presentes na literatura, encontram-se os Algoritmos Genéticos, abordado por Pezzella, Morganti e Ciaschetti (2008), Zhang, Gao e Shi (2011), Teekeng e Thammano (2012) e Rigo (2018). Já Li *et. al* (2017), utiliza um algoritmo combinado baseado na colônia artificial de abelhas e na busca Tabu. Yuan, Xu e Yang (2013) propõe um algoritmo híbrido baseado na metaheurística *Harmony Search* e em abordagem integrada, enquanto que, Tamssaouet, Dauzère-Pérès e Yugma (2018) optaram por formular duas metaheurísticas, *Simulated Annealing* e *Tabu Search*, sendo a primeira baseada em busca local probabilística, e a segunda, em busca Tabu. Zhang e Wang (2015) utiliza uma abordagem híbrida do Algoritmo Genético em conjunto com a teoria das restrições (Pulmão, Tambor, Corda) para identificação de gargalos. Já Wang *et. al* (2012) propõe um algoritmo de estimação de distribuição denominado *Estimation of Distribution Algorithm*, que utiliza um modelo de probabilidade para encontrar soluções em locais mais promissores.

Outro método que tem sido abordado e obtido sucesso é a metaheurística *Variable Neighborhood Search* (VNS). Zhang *et. al* (2018) propõe a utilização do

VNS combinado com o Algoritmo Genético para resolução do FJSP, enquanto que Gao *et. al* (2008) também faz uma combinação com o Algoritmo Genético, porém com uma variação do VNS, o *Variable Neighborhood Descent* (VND). A metaheurística VNS também é abordada por autores como Bagheri e Zandieh (2011), Lei e Guo (2014) e Amiri, Zandieh e Yasdani (2010). Um resumo dos métodos dos autores abordados se encontra na Tabela 2.

Tabela 2 - Métodos de resolução do FJSP

Autores	Método utilizado	Objetivo	Solução Inicial	Estruturas de Vizinhaça
Pezzella, Morganti e Ciaschetti (2018)	Algoritmo Genético	Minimizar o <i>makespan</i>	Combinação de: regras com base em processamento mínimo global e permutação aleatória, MWR e MOR	Crossover e mutação
Zhang, Gao e Shi (2011)	Algoritmo Genético	Minimizar o <i>makespan</i>	GS (Seleção Global) e LS (Seleção Local)	Crossover e mutação
Teekeng e Thammano (2012)	Algoritmo Genético	Minimizar o <i>makespan</i>	Solução Aleatória	Crossover e mutação
Rigo (2018)	Algoritmo Genético	Minimizar o <i>makespan</i>	Solução Aleatória de operações combinada com escolha de máquinas com menor carga alocada	Crossover (OX1) e mutação
Li <i>et. al</i> (2017)	Algoritmo Híbrido (Colônia artificial de formigas e Busca Tabu)	Minimizar o <i>makespan</i>	Método de Roleta de agrupamento de <i>clusters</i>	Crossover e Mutação (Permutação, Inserção e Inversão) e <i>N5</i> (baseada em caminho crítico)
Yuan, Xu e Yang (2013)	Algoritmo Híbrido (<i>Harmony Search</i> e abordagem integrada)	Minimizar o <i>makespan</i>	Abordagem por localização (AL) e MWR	Exclusão e Inserção baseado no caminho crítico

Tamssaouet, Dauzère-Pérès e Yugma (2018)	<i>Simulated Annealing</i> e busca Tabu	Minimizar o <i>makespan</i>	Heurística construtiva	Permutação
Zhang e Wang (2015)	Algoritmo Híbrido (Algoritmo Genético e teoria das restrições)	Minimizar o <i>makespan</i>	Teoria das restrições (baseado na determinação de gargalos)	Crossover e mutação
Wang <i>et. al</i> (2012)	Estimação de Distribuição	Minimizar o <i>makespan</i>	Matrizes de probabilidade	Amostragem de matrizes de probabilidade
Zhang <i>et. al</i> (2018)	Algoritmo Híbrido (Algoritmo Genético e VNS)	Minimizar o <i>makespan</i>	Máquina: Seleção aleatória, seleção local e seleção global; Operação: método aleatório	Crossover e mutação, <i>Re</i> , <i>In</i> (Inserção) e <i>Ex</i>
Gao <i>et. al</i> (2018)	Algoritmo Híbrido (Algoritmo Genético e VNS)	Minimizar o <i>makespan</i> , minimizar a carga da máquina de maior tempo, minimizar a carga total das máquinas.	Solução Aleatória	Crossover, Mutação (baseado em alelo e imigração) e Inserção
Bagheri e Zandieh (2011)	VNS	Minimizar o <i>makespan</i> e minimizar o atraso médio	Solução Aleatória	<i>Re</i> , <i>AS</i> , <i>Co</i>
Lei e Guo (2014)	VNS	Minimizar o <i>makespan</i>	Solução Aleatória	Permutação (Swap), Inserção, <i>Assign</i> , <i>Change</i>
Amiri, Zandieh e Yasdani (2010)	VNS	Minimizar o <i>makespan</i>	Baseado na Solução Inicial de Pezzella (2008)	Se_1 , <i>As</i> , <i>Co</i> , Se_2 , I_1 , I_2

Fonte: Do Autor, 2019.

Percebe-se, através da Tabela 2, que o objetivo mais adotado dentre os autores abordados é a minimização do *makespan*, com exceção de Gao *et. al* (2018) e Bagheri e Zandieh (2011), que buscam além da minimização do *makespan*, a minimização de carga alocada às máquinas e a minimização do atraso médio, respectivamente.

Com relação à geração inicial de soluções, a maioria dos autores apresentados utilizam métodos baseados em heurísticas e regras de sequenciamento para obtenção de uma solução inicial com o intuito de garantir, em sua inicialização, soluções de maior qualidade e que convirjam mais rapidamente para a solução final.

Tendo em consideração as estruturas de vizinhança adotadas nas pesquisas, conforme apresentado na Tabela 2, são utilizados métodos baseados em caminhos críticos, *crossover* e mutação, inserção, permutação, entre outros. Lei e Guo (2014), dentre as quatro estruturas utilizadas, aborda a vizinhança denominada *Swap*, a qual é baseada na troca de duas operações, selecionadas aleatoriamente, da solução. Neste método, a coerência de operações de uma mesma tarefa (*job*) e a máquina disponível para a operação é mantida. Já Li *et. al* (2017), em seu processo de geração de população, utiliza como um dos operadores de mutação o método de inversão, que consiste na seleção aleatória de duas posições na solução, sendo invertido a ordem de todas as operações entre essas posições.

Outras duas estruturas de vizinhança, apresentadas por Amiri, Zandieh e Yasdani (2010), são intituladas como Se_1 e I_2 . A primeira, Se_1 , se baseia na seleção aleatória de uma operação da solução e, se for factível, há a troca de posição da operação selecionada com a operação predecessora. Caso não seja possível a troca com a operação predecessora, a operação sucessora é analisada para permutação. Já a segunda vizinhança, I_2 , considerada inteligente, possui como método a seleção aleatória de uma operação dentre as operações da solução designadas à máquina de maior tempo de processamento (responsável pela determinação do *makespan*). Essa operação é, então, alocada a máquina de menor tempo de processamento e, caso a máquina de menor tempo de processamento não seja factível para a operação, é selecionado outra máquina disponível para alocação.

2.5. VARIABLE NEIGHBORHOOD SEARCH (VNS)

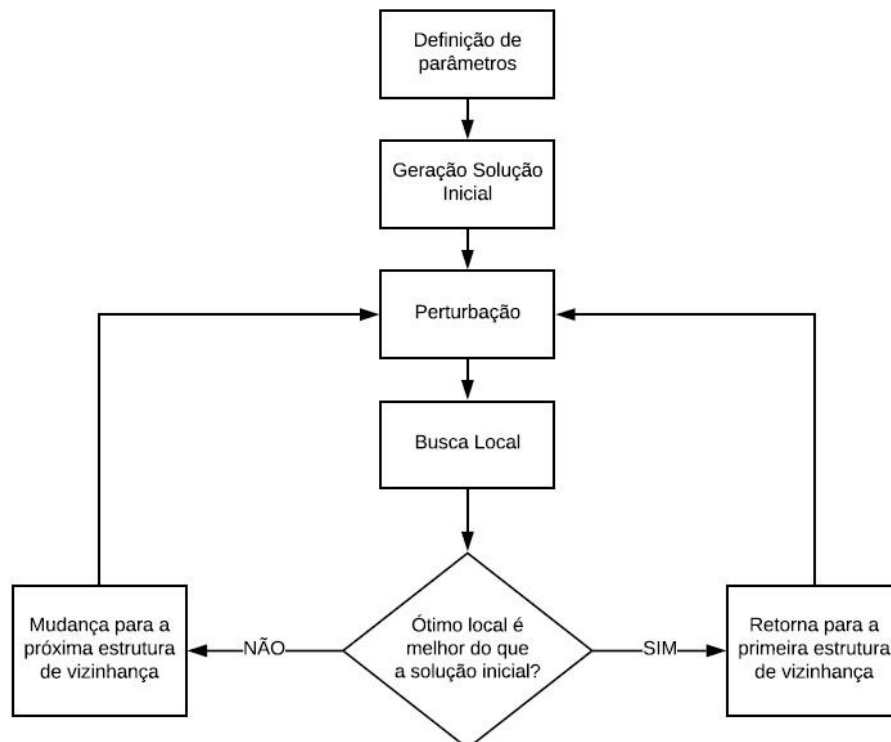
A metaheurística *Variable Neighborhood Search*, proposta por Mandlenovic, possui como princípio básico uma mudança sistemática de vizinhanças no seu processo de busca de soluções combinando métodos determinísticos e estocásticos (GLOVER; KOCHENBERGER, 2013). A mudança de vizinhança tem como objetivo obter diferentes ótimos locais e, conseqüentemente, escapar deles para que melhores soluções possam ser obtidas (TALBI, 2013).

Para Glover e Kochenberger (2013), três fatores básicos são considerados no VNS:

- 1) Um ótimo local em uma determinada vizinhança não é, necessariamente, um ótimo local em outra vizinhança;
- 2) Um ótimo global é um ótimo local para todas as estruturas de vizinhança;
- 3) Em muitos problemas, os ótimos locais das diferentes estruturas de vizinhança são próximos e fornecem informações sobre o ótimo global.

Os passos básicos do VNS são descritos na Figura 3.

Figura 3 - Estrutura básica do VNS



Fonte: Adaptado de Amiri, Zandieh e Yasdani (2010)

Conforme apresentado na Figura 3, durante a inicialização da metaheurística há a definição dos parâmetros, onde são determinados conjuntos de estruturas de vizinhanças para a etapa de perturbação, denominada etapa *shaking*, e para a etapa de busca local, bem como, é estabelecido o critério de parada do algoritmo. Sendo assim, uma solução inicial é gerada e definida como a solução atual. A solução atual é então passada por um processo de perturbação, o qual gera uma solução vizinha através da primeira estrutura de vizinhança do conjunto de vizinhanças da etapa *shaking*. Logo após, a solução vizinha gerada é utilizada como uma solução de entrada do processo de busca local. O ótimo local encontrado no processo de busca local é então comparado à solução de entrada, caso o ótimo local seja melhor do que a solução de entrada, ele passa a ser a solução inicial do algoritmo e recomeça os passos de perturbação e busca local novamente considerando a primeira estrutura de vizinha dos conjuntos, senão, uma nova estrutura de vizinhança é adotada. Uma iteração do algoritmo é finalizada quando todas as vizinhanças dos conjuntos de estruturas de vizinhança são utilizadas. Quando finalizada, a solução encontrada passa a ser a solução inicial da próxima iteração. As iterações do algoritmo são realizadas até o critério de parada (AMIRI; ZANDIEH; YASDANI, 2010).

Os critérios de parada a serem adotados no VNS podem ser tempo máximo de processamento permitido, número máximo de iterações ou número máximo de iterações entre duas melhorias (GLOVER; KOCHENBERGER, 2013). Um fator importante a ser considerado no VNS são os tipos de vizinhança a serem adotados. Para Amiri, Zandieh e Yasdani (2010) o desempenho dessa metaheurística é consideravelmente atrelado à eficiência das estruturas de vizinhança utilizadas.

O *Variable Neighborhood Search*, apesar de ser uma metaheurística pouco explorada, foi formulado para solucionar problemas de otimização combinatória e global, com aplicação a problemas de programação linear inteira, mista, e não linear, sendo um deles o problema de sequenciamento de produção em *job shop* flexível (CHAUDRY; KHAN, 2016).

3. ALGORITMO PROPOSTO

O algoritmo de busca em vizinhança variável proposto neste trabalho tem como base a metaheurística *Variable Neighborhood Search*. Conforme capítulo 2, seção 2.5, a estrutura básica do VNS consiste em:

- Parâmetros de inicialização: determinação dos conjuntos de estruturas de vizinhança e critério de parada do algoritmo. No caso do algoritmo proposto nesta pesquisa, o critério de parada é o tempo estabelecido pelo interlocutor;
- Geração de solução inicial;
- *Shaking*: geração de uma solução de entrada da busca local conforme a determinação do conjunto de vizinhanças;
- Busca Local: realização de busca até o ótimo local a ser realizada conforme a determinação do conjunto de vizinhanças;

Além disso, a forma de representação da solução e a função objetivo a ser adotada para resolução do problema em estudo também são necessárias. Dessa forma, os passos adotados para formulação do algoritmo sugerido não descritos nos próximos capítulos.

3.1. REPRESENTAÇÃO DA SOLUÇÃO

Na representação da solução de um FJSP, por se tratar de um problema que se subdivide em sequenciamento de operações e designação das máquinas, é necessário que se evidencie a sequência das tarefas, e suas respectivas operações, a serem realizadas e a máquina determinada para o processamento de cada uma. Assim, a representação da solução deste trabalho se baseia no modelo apresentado por Rigo (2018), no qual a solução é dividida em dois vetores: o primeiro contém a sequência de realização das operações de cada tarefa (*job*) e o segundo representa a máquina selecionada para o processamento da operação.

Um exemplo de um problema FJSP e uma possível solução são apresentados na Tabela 3 e Figura 2, respectivamente.

Tabela 3 - Demonstração 2 de um problema FJSP

Job (J_i)	Operação (O_{ij})	M_1	M_2	M_3
1	1	5	5	-
1	2	7	-	9
2	1	5	4	7
2	2	-	10	10

Fonte: Adaptado de Rigo (2018)

Figura 4 - Representação da Solução

Job	2	1	1	2
Máquina	3	2	1	2
Posição	1	2	3	4

Fonte: Adaptado de Rigo (2018)

O comprimento total dos dois vetores é determinado de acordo com o número total de operações do problema. Portanto, considerando-se o problema apresentado na Tabela 3 de dois *jobs*, cada um contendo duas operações, e três máquinas, o número total de operações é quatro, sendo, assim, o comprimento do vetor da figura 4, que representa uma solução.

O primeiro vetor especifica o índice do *job* correspondente em ordem de sequenciamento, sendo o índice da operação considerado em ordem crescente partindo da primeira operação de cada *job*. De acordo com a figura 4, a primeira operação sequenciada (de posição 1) corresponde à primeira operação do J_2 , ou seja, do segundo *job*, a segunda operação sequenciada é a primeira operação do J_1 , a terceira é a segunda operação do J_1 e a última a segunda operação do J_2 .

O segundo vetor indica o índice da máquina designada, sendo o índice a ordem das máquinas disponíveis para cada operação. Considerando a figura 4, a designação das máquinas é feita da seguinte forma: a máquina designada para a primeira operação do J_2 , operação O_{21} , é máquina M_3 , que correspondente à terceira máquina disponível para a operação (índice 3); a máquina designada para a operação O_{11} é a máquina M_2 , que é a segunda máquina disponível (índice 2); a máquina M_1 é a máquina correspondente da operação O_{12} , sendo a primeira máquina disponível (índice 1); e a operação O_{22} é alocada à segunda máquina disponível (índice 2), máquina M_3 .

3.2. AVALIAÇÃO DA SOLUÇÃO

A função objetivo selecionada para esta pesquisa é a minimização do *makespan*, ou seja, a avaliação da solução adotada se baseia no cálculo do *makespan*. O *makespan* é determinado através da soma dos tempos de processamento das operações, considerando-se o tempo de término de cada operação nas respectivas máquinas alocadas. Assim, o momento de conclusão de todas operações na máquina correspondente ao maior tempo de processamento é considerado como o *makespan*. Assim, sendo a função objetivo estabelecida como minimização do *makespan*, a qualidade da solução é atrelada ao menor *makespan* encontrado para o problema.

3.3. SOLUÇÃO INICIAL

Para obtenção de uma solução inicial que apresente qualidade relativamente boa, a fim de que a conversão para melhores soluções seja, possivelmente, mais rápida, optou-se por utilizar como embasamento o método apresentado por Zhang *et. al* (2018), o qual utiliza sequenciamento aleatório das operações e, para cada operação, designa a máquina que possui o menor tempo de processamento até então (método de seleção local).

Para a geração da solução, o vetor correspondente ao sequenciamento, a fim de se garantir uma geração de solução inicial não determinística, é gerado aleatoriamente contendo todos os *jobs* e todas as operações do problema. Após, para a composição do vetor que especifica a máquina designada para cada operação, é verificado os tempos de processamento das máquinas disponíveis para cada operação em análise. Inicialmente, todas as máquinas possuem tempo de processamento zero, portanto, as alocações começam a ser feitas de acordo com a primeira máquina disponível. Conforme os tempos de processamento são acumulados nas máquinas, aloca-se a operação a todas as máquinas disponíveis e aquela que possuir o menor tempo de processamento após a locação, é escolhida para processamento da operação em análise.

3.4. ESTRUTURAS DE VIZINHANÇA

As estruturas de vizinhança desempenham um papel importante no VNS, já que a metaheurística se baseia em explorar diferentes locais do espaço de solução de acordo com mudança sistemática de vizinhança. Assim, neste trabalho, quatro diferentes estruturas de vizinhança são adotadas tendo como base a vizinhança de troca de dois pontos proposta por Lei e Guo (2014), o método de inversão utilizado por Li *et. al* (2017), e as vizinhanças Se_1 e I_2 apresentadas por Amiri, Zandieh e Yasdani (2010). A escolha destas vizinhanças levou em consideração a facilidade de implementação e pela relevância na literatura.

3.4.1. Estrutura de Vizinhança V_1

A estrutura de vizinhança V_1 , baseada na Se_1 proposta por Amiri, Zandieh e Yasdani (2010) tem como princípio a troca de duas operações adjacentes no vetor de solução. A geração de vizinhos é realizada da seguinte forma: uma posição da solução candidata é selecionada aleatoriamente, sendo assim, analisa-se a posição predecessora à posição selecionada. Caso o *job* contido na posição selecionada seja diferente do *job* da posição predecessora, ocorre a troca das operações nestas duas posições. Caso os *jobs* sejam iguais, analisa-se a posição sucessora à posição selecionada para troca. Se tanto o *job* da posição predecessora como o da posição sucessora forem iguais ao *job* da posição selecionada, não há troca. Nesta estrutura, a designação das máquinas corresponde às operações não muda.

Considere a Figura 5 como um exemplo de geração de vizinhos através da estrutura V_1 . No exemplo (a), a posição cinco da solução é selecionada aleatoriamente. Assim, analisa-se sua posição predecessora, a posição quatro. Como o *job* contido na posição cinco é o *job* 1, e o *job* contido na posição quatro é o *job* 2, a troca ocorre, mantendo-se as máquinas designadas. Já no exemplo (b), a posição selecionada é a posição seis. Como os *jobs* da posição seis e da posição cinco são iguais, a troca não acontece e a análise passa a ser com a posição sucessora, a posição sete. Como o *job* contido na posição sete é o *job* 3, a troca acontece.

Figura 5 - Exemplo da Estrutura de Vizinhaça V_1

Job	1	1	3	2	1	2	3	3	3
Máquina	3	4	3	2	1	1	2	4	3
Posição	1	2	3	4	5	6	7	8	9

↓

Job	1	1	3	1	2	2	3	3	3
Máquina	3	4	3	1	2	1	2	4	3
Posição	1	2	3	4	5	6	7	8	9

(a)

Job	1	1	3	1	2	2	3	3	3
Máquina	3	4	3	1	2	1	2	4	3
Posição	1	2	3	4	5	6	7	8	9

↓

Job	1	1	3	1	2	3	2	3	3
Máquina	3	4	3	1	2	2	1	4	3
Posição	1	2	3	4	5	6	7	8	9

(b)

Fonte: Do autor, 2019.

3.4.2. Estrutura de Vizinhaça V_2

A estrutura de vizinhaça V_2 , também apresentada por Amiri, Zandieh e Yasdani (2010) como I_2 , tem como princípio a troca de operações entre máquinas com base na máquina de maior tempo de processamento, ou seja, aquele que é responsável pela determinação do *makespan*. Neste tipo de geração de vizinhos, uma operação é selecionada aleatoriamente dentre as operações alocadas à máquina de maior tempo de processamento. Assim, a operação passa a ser candidata à alocação na máquina de menor tempo de processamento. Caso a máquina de menor tempo de processamento não seja uma máquina disponível para processamento da operação em análise, uma máquina, dentre o conjunto de máquinas disponíveis para aquela operação, é escolhida aleatoriamente para alocação. Caso a operação selecionada possa ser realizada por apenas uma máquina, a troca não é realizada.

3.4.3. Estrutura de Vizinhança V_3

A estrutura de vizinhança troca V_3 tem é uma adaptação da estrutura de vizinhança denominada *Swap* proposta por Lei e Guo (2014). Para a geração de vizinhos, dois pontos da solução candidata são selecionados aleatoriamente e as operações contidas nestes são trocadas. Na Figura 6, os pontos dois e seis são selecionados e, sendo assim, são trocados. Neste tipo de estrutura, a coerência de ordem das máquinas dentro do intervalo dos pontos selecionados, é mantida. Como por exemplo, na Figura 6, dentro do intervalo dos pontos dois e seis a primeira vez que o *job* 2 aparece (posição quatro), a máquina designada é máquina 2. Sendo assim, no vizinho gerado, na primeira vez que o *job* 2 aparece (posição dois), a máquina designada também é a máquina 2, e assim sucessivamente em todas as operações presentes no intervalo.

Figura 6 - Exemplo da Estrutura de Vizinhança V_2

Job	1	1	3	2	1	2	3	3	3
Máquina	3	4	3	2	1	1	2	4	3
Posição	1	2	3	4	5	6	7	8	9

Job	1	2	3	1	2	1	3	3	3
Máquina	3	2	3	4	1	1	2	4	3
Posição	1	2	3	4	5	6	7	8	9

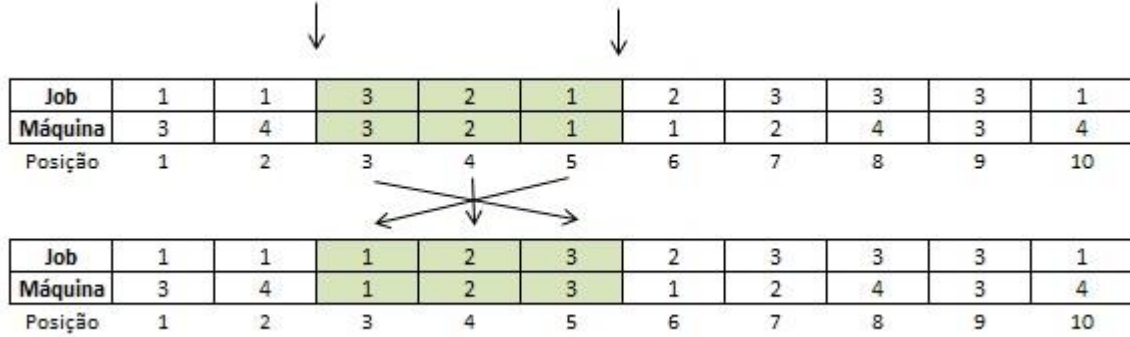
Fonte: Do autor, 2019.

3.4.4. Estrutura de Vizinhança V_4

A estrutura de vizinhança V_4 é adaptada do método da etapa de mutação apresentado por Li *et. al* (2017). Nesta vizinhança, um tamanho do subtrecho a ser invertido é pré-estabelecido. No caso do exemplo da Figura 7, o tamanho escolhido é de 30% do tamanho total da solução. Assim, uma posição da solução candidata é aleatoriamente selecionada e a direção do subtrecho é escolhida da seguinte maneira: gera-se um número aleatório de 0 a 1, se o número gerado for menor ou igual a 0,5 a direção é para a esquerda, e caso maior que 0,5 a direção é para a direita. No exemplo, a posição dois é selecionada e a direção adotada é para a direita (número aleatório gerado maior que 0,5). A partir disto, o subtrecho de

tamanho três após a posição dois, é invertido. Nessa geração de vizinhos, a coerência de ordem das máquinas, assim como na vizinhança V_3 , deve ser mantida.

Figura 7 - Exemplo da Estrutura de Vizinhança V_3



Fonte: Do autor, 2019.

4. RESULTADOS E DISCUSSÃO

O algoritmo proposto neste trabalho foi desenvolvido conforme procedimentos e critérios descritos no Capítulo 3. Para a definição prévia dos parâmetros de inicialização do algoritmo, bem como para a validação de seu desempenho, algumas instâncias da literatura foram utilizadas (MASTROLILLI, 2018).

Os conjuntos de instâncias de Hurink *et. al* (1994) foram escolhidas para os testes prévios e para a aplicação do algoritmo. O autor desenvolveu três grupos de instâncias adaptadas ao problema de *Job Shop* Flexível, os quais possuem probabilidades de flexibilidade de máquinas diferentes:

- **edata:** conjunto de 66 instâncias, nas quais somente algumas operações a serem sequenciadas podem ser alocadas em mais de uma máquina;
- **rdata:** conjunto de 66 instâncias, nas quais a maioria das operações a serem sequenciadas podem ser alocadas em mais de uma máquina;
- **vdata:** conjunto de 66 instâncias, nas quais todas as operações a serem sequenciadas podem ser alocadas em mais de uma máquina.

Para a análise dos resultados e avaliação da qualidade da solução alcançada pelo algoritmo, o melhor *makespan* já encontrado para cada instância dos três grupos de Hurink *et. al*, apresentado pelo trabalho de Behnke e Geiger (2012), foi utilizado para comparação. Além disso, um algoritmo de geração de soluções totalmente aleatórias foi executado, obtendo-se para cada instância o melhor resultado aleatório dentre as 1000 soluções geradas.

A comparação das soluções encontradas foi realizada através do cálculo do *gap* (porcentagem de diferença com relação ao melhor *makespan* conhecido), descrito pela equação (16). Outro desempenho analisado foi o tempo de execução por iteração do algoritmo, descrito pela equação (17).

$$Gap = \frac{(Resultado\ obtido - Melhor\ makespan\ conhecido)}{Melhor\ makespan\ conhecido} * 100 \quad (16)$$

$$Tempo\ por\ iteração = \frac{Tempo\ total\ de\ execução}{Quantidade\ de\ iterações\ realizadas} \quad (17)$$

A implementação do algoritmo foi realizada através da linguagem *Excel Vba* e os testes realizados em computadores com processador Intel Core i7 e memória instalada de 8 GB.

4.1. TESTE PRELIMINAR DAS ESTRUTURAS DE VIZINHANÇA

Conforme abordado nos Capítulos 2 e 3, a definição dos conjuntos de estruturas de vizinhança das etapas de perturbação e de busca local é necessária na etapa de inicialização do algoritmo.

De forma que o VNS tem como base a mudança sistemática de vizinhança, diferentes conjuntos de vizinhanças podem resultar em diferentes configurações do algoritmo proposto. Da mesma forma, a quantidade de vizinhos a serem gerados pelas vizinhanças também precisam ser determinados. Para isso, um algoritmo baseado no *Hill-Climbing* foi desenvolvido para que as vizinhanças propostas pudessem ser testadas individualmente, a fim de se analisar seu comportamento e desempenho e estabelecer a configuração do VNS.

O grupo *edata* das instâncias de Hurink *et. al* (1994) foi utilizado para testes das vizinhanças, as quais obtiveram os resultados descritos nos Apêndices A, B e C. Os resultados contidos nos Apêndices A, B e C contemplam o *gap* com relação à melhor solução, a quantidade de iterações realizadas, o número da iteração do resultado obtido, a quantidade de melhorias até o ótimo global, o tempo de execução do algoritmo e o tempo de execução do algoritmo por iteração. Um resumo dos resultados é apresentado nas Tabelas 4 e 5.

Tabela 4 - *Score* dos resultados obtidos com o grupo de instâncias *rdata* do autor Hurink *et al.* (1994)

Vizinhanças	V_1	V_2	V_3	V_4
Melhor Resultado	14	15	27	14
Maior Quantidade de Iterações Realizadas	59	0	0	7
Menor quantidade de Iterações até o ótimo local	11	16	25	14
Maior Quantidade de Melhorias	25	15	10	27

Fonte: Do Autor, 2019

Tabela 5 - Resumo dos resultados obtidos pelo *Hill-Climbing* com o grupo de instâncias rdata do autor Hurink *et al.* (1994)

Vizinhanças	V_1	V_2	V_3	V_4
Gap médio em relação ao melhor resultado da literatura	30,71%	30,95%	29,65%	30,74%
Tempo médio por iteração (ms)	22,0	46,8	52,2	24,1
Quantidade média de iterações até o ótimo local	379	173	153	277
Quantidade média de iterações entre melhorias	7	6	6	7

Fonte: Do Autor, 2019

A Tabela 4 apresenta o *score* de cada vizinhança para as categorias de melhor resultado, maior quantidade de iterações realizadas, maior quantidade de iterações realizadas até o ótimo local e a maior quantidade de melhorias no ótimo global. Entende-se por *score* a quantidade de vezes que cada vizinhança obteve o melhor resultado dentre as quatro vizinhanças para cada categoria. A somatória de cada categoria pode resultar em um valor maior que 66, quantidade total de instâncias utilizadas, pois pode haver empate nos resultados. A Tabela 5 descreve o *gap* médio obtido para cada vizinhança, o tempo médio por iteração do algoritmo em milissegundos, a quantidade média de iterações até o ótimo local e a quantidade média de iterações entre melhorias.

Conforme os dados descritos nas Tabelas 4 e 5, a vizinhança V_3 obteve o melhor desempenho com relação ao *makespan*, apresentando um *gap* médio de 29,65%, e a maior quantidade de melhores resultados ao longo das 66 instâncias. Já com relação à rapidez de execução do algoritmo, em 59 das 66 instâncias, a vizinhança V_1 foi a estrutura que executou a maior quantidade de iterações e obteve o menor tempo médio de execução por iteração, de 22 milissegundos. No que diz respeito ao alcance do ótimo local, a vizinhança V_3 apresenta a menor quantidade de iterações e é, também, a vizinhança que apontou a menor quantidade média de iterações entre melhorias, juntamente com a estrutura V_2 . A estrutura V_4 exibiu a melhor performance com relação à maior quantidade de melhorias durante a execução do algoritmo, sendo a melhor vizinhança para essa categoria 27 vezes.

4.2. CONFIGURAÇÃO DO VNS

Conforme os resultados obtidos nos testes das vizinhanças, através da aplicação do algoritmo baseado no *Hill-Climbing*, diferentes configurações do algoritmo proposto foram programadas com base em diferentes configurações dos conjuntos de vizinhança. A Tabela 6 apresenta as quatro configurações testadas nesta pesquisa. Para cada configuração, a Tabela 6 descreve o conjunto das estruturas de vizinhança, em ordens de utilização pelo algoritmo, a quantidade de vizinhos a serem gerados por cada estrutura e o tamanho do subtrecho a ser invertido na vizinhança V_4 .

Tabela 6 - Configurações do algoritmo proposto

Configuração	Etapa <i>Shaking</i>			Etapa Busca Local		
	Ordem das Vizinhanças	Quantidade de vizinhos	Tamanho do subtrecho	Ordem das Vizinhanças	Quantidade de vizinhos	Tamanho do subtrecho
1	V_1	5	20%	V_3	10	20%
	V_4	5		V_1	15	
	V_2	5		V_4	15	
	V_3	5		V_2	20	
2	V_3	3	20%	V_1	5	10%
	V_1	3		V_4	5	
	V_4	3		V_2	10	
	V_2	3		V_3	10	
	V_3	3		V_1	15	
	V_1	3		V_4	15	
3	V_1	3	20%	V_3	5	20%
	V_4	3		V_1	5	
	V_2	3		V_4	10	
	V_3	3		V_2	10	
	V_1	3		V_3	15	
	V_4	3		V_1	15	
4	V_2	3	20%	V_3	5	20%
	V_3	3		V_2	5	
	V_1	3		V_4	10	
	V_4	3		V_1	10	
	V_2	3		V_3	15	
	V_3	3		V_2	15	

Fonte: Do Autor, 2019

A Configuração 1 foi estabelecida com base na velocidade das vizinhanças, em ordem crescente do tempo médio por iteração, para a etapa de perturbação (*shaking*) e com base na ordem crescente do *gap* médio para a etapa de busca local. Para essa configuração, determinou-se uma menor quantidade de estruturas de vizinhança e maior quantidade de vizinhos gerados por cada estrutura, a fim de se analisar se uma maior exploração da vizinhança é mais promissora do que uma maior troca de estruturas.

Já na Configuração 2, a ordem das estruturas de vizinhança da etapa de perturbação levou em consideração a ordem crescente do *gap* médio, utilizando-se primeiramente as quatro diferentes estruturas de vizinhança e retornando, ao final, para as duas de menor *gap*. Para a etapa de busca local, a velocidade da vizinhança foi levada em consideração, em ordem crescente do tempo médio por iteração, com a mesma lógica de utilização das quatro diferentes estruturas e retorno para as duas melhores da categoria. Nesta configuração, estabeleceu-se uma maior quantidade de troca entre estruturas de vizinhança e menor quantidade de vizinhos gerados.

Para a Configuração 3 a mesma lógica da configuração 2 foi utilizada, porém, adotando-se a ordem crescente do tempo médio por iteração na etapa de perturbação do algoritmo e a ordem crescente do *gap* médio na etapa de busca local. Optou-se por analisar uma configuração que gerasse soluções de entrada para a busca local mais rapidamente (etapa de perturbação) e realizasse uma busca local com propensão a uma qualidade melhor da solução.

Na Configuração 4, os critérios de quantidade média de iterações entre melhorias e quantidade média de iterações até o ótimo local da tabela 5 foram levados em consideração. Para a etapa *shaking*, a ordem das vizinhanças foi estabelecida de acordo com a ordem crescente de quantidade média de iterações entre melhorias, e para a etapa de busca local a ordem foi determinada conforme a ordem crescente de quantidade média de iterações até o ótimo local. A mesma premissa adotada nas Configurações 2 e 3 com relação a quantidade de estruturas de vizinhança e quantidade de vizinhos geradas por cada uma também foi adotada na Configuração 4.

Ajustes nas ordens das vizinhanças foram realizados a fim de se evitar que a vizinhança utilizada na etapa de *shaking* para a geração da solução de entrada da busca local fosse a mesma adotada na busca local.

4.3. ANÁLISE DA QUALIDADE DAS SOLUÇÕES OBTIDAS (*MAKESPAN*)

A partir da programação do algoritmo em quatro diferentes configurações, conforme descrito na seção 4.2, utilizou-se os grupos de instâncias *edata*, *rdata* e *vdata* de Hurink *et. al* (1994) para a realização dos testes. Os Apêndices D, E, F e G descrevem os resultados obtidos para cada instância do grupo *edata* nas quatro configurações, respectivamente. Os Apêndices H, I, J e K apresentam os resultados obtidos para cada instância do grupo *rdata* nas quatro configurações, respectivamente. Já os Apêndices L, M, N e O demonstram os resultados obtidos para cada instância do grupo *vdata* nas quatro configurações do algoritmo, respectivamente. Entende-se por resultados obtidos o *gap* calculado com relação ao melhor resultado aleatório encontrado, o *gap* calculado com relação ao melhor resultado conhecido e, para o conjunto *edata*, o *gap* calculado com relação ao melhor resultado obtido pelo *Hill-Climbing*.

A Tabela 7 apresenta o resumo dos resultados das quatro configurações para o grupo de instâncias *edata*, as Tabelas 8 e 9 descrevem o resumo dos resultados para os conjuntos *rdata* e *vdata*, respectivamente.

Tabela 7 - Resumo dos resultados obtidos com o grupo de instâncias *edata* do autor Hurink *et al.* (1994)

Configurações	1	2	3	4
Gap médio com relação à melhor solução aleatória	-18,72%	-20,31%	-19,55%	-17,47%
Gap médio com relação ao melhor resultado conhecido na literatura	16,02%	13,83%	14,84%	17,72%

Fonte: Do Autor, 2019

Tabela 8 - Resumo dos resultados obtidos com o grupo de instâncias *rdata* do autor Hurink *et al.* (1994)

Configurações	1	2	3	4
Gap médio com relação à melhor solução aleatória	-22,70%	-23,04%	-23,50%	-22,07%
Gap médio com relação ao melhor resultado <i>Hill-Climbing</i>	-5,34%	-5,75%	-6,34%	-4,53%
Gap médio com relação ao melhor resultado conhecido na literatura	21,03%	20,59%	19,71%	22,17%

Fonte: Do Autor, 2019

Tabela 9 - Resumo dos resultados obtidos com o grupo de instâncias *vdata* do autor Hurink *et al.* (1994)

Configurações	1	2	3	4
Gap médio com relação à melhor solução aleatória	-24,14%	-24,03%	-24,34%	-23,50%
Gap médio com relação ao melhor resultado conhecido	20,82%	21,29%	20,45%	22,17%

Fonte: Do Autor, 2019

Conforme os resultados das Tabelas 7, 8 e 9, percebe-se, através do *gap* médio obtido com relação à melhor solução aleatória encontrada, que o algoritmo proposto nas quatro configurações é capaz de obter resultados significativamente melhores do que uma solução gerada aleatoriamente e, também, de obter resultados melhores do que os encontrados através do *Hill-Climbing* para as instâncias do grupo *edata*. Constata-se, também, que o algoritmo apresenta bons resultados, relativamente próximos dos melhores resultados já conhecidos, apresentando o maior *gap* médio entre as quatro configurações de 17,72%, 22,17% e 22,17% para os grupos *edata*, *rdata* e *vdata*, respectivamente.

O desempenho positivo do algoritmo também é afirmado através dos resultados individuais de cada instância. De acordo com o Apêndice D, para o conjunto *edata*, o algoritmo apresentou um resultado melhor do que o melhor *makespan* conhecido para a instância de código car6‡, com um *gap* de -0,39%. Além disso, para diversas instâncias dos três grupos, o algoritmo apresentou resultados iguais aos melhores resultados conhecidos ou bem próximos, apresentando um *gap* próximo de 0%.

Considerando-se a tendência de desempenho das outras instâncias, a obtenção de um resultado melhor do que os melhores *makespans* conhecidos pode ter sido ocasionado pela aleatoriedade presente no algoritmo. Constatou-se, também, que para algumas instâncias, o algoritmo apresentou um *gap* mais elevado. Este fato pode ter sido ocasionado pelas estruturas de vizinhança adotadas, as quais obtiveram certa dificuldade em escapar de ótimos locais na etapa de perturbação, fator pelo qual os resultados obtidos pelo VNS não foram tão significativamente melhores do que os obtidos pelo *Hill-Climbing*.

Para cada configuração programada, a Tabela 10 descreve quantas vezes cada configuração apresentou o melhor resultado dentre as quatro configurações.

Tabela 10 - Score de cada configuração para o melhor resultado

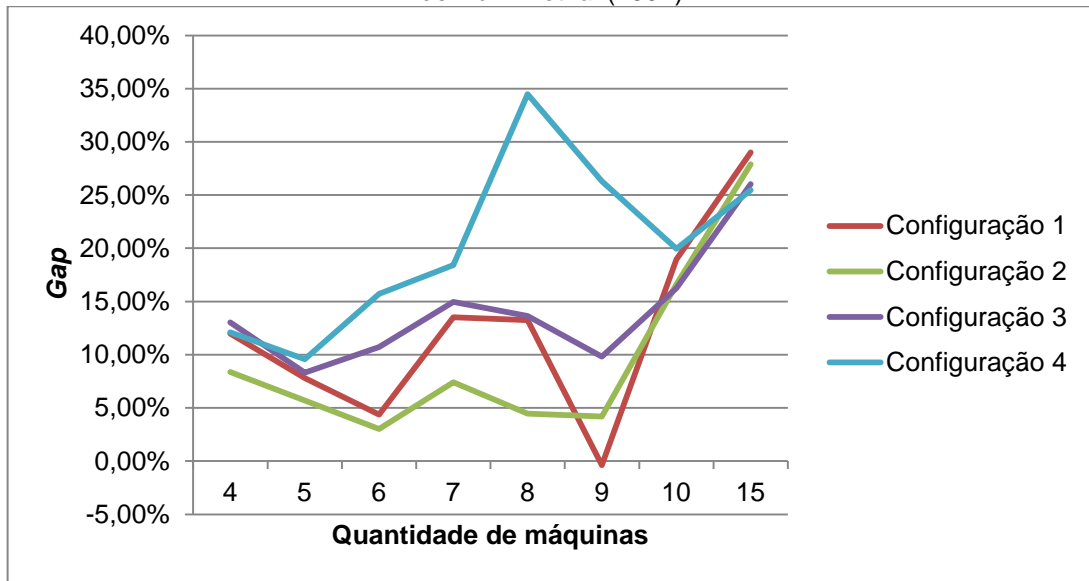
Configurações	1	2	3	4
<i>edata</i>	13	30	19	10
<i>rdata</i>	10	13	23	23
<i>vdata</i>	18	9	20	22

Fonte: Do Autor, 2019

No que diz respeito ao desempenho de cada configuração, apesar da similaridade dos resultados das quatro configurações, para o grupo de instâncias *edata* a Configuração 2 apresentou a melhor performance, com um *gap* médio de 13,83% relativo ao melhor *makespan* conhecido, e foi a configuração que obteve o maior número de melhores resultados, conforme Tabela 10. Para o grupo de instâncias *rdata*, a Configuração 3 foi a que obteve o melhor *gap* médio, de 19,71%, bem como obteve o melhor *score*. Já para o grupo *vdata*, apesar da Configuração 4 obter mais vezes o melhor resultado dentre as quatro configurações, a Configuração 3 foi a que apresentou o melhor desempenho para o *gap* médio, de 20,45%, com relação ao melhor *makespan* já conhecido.

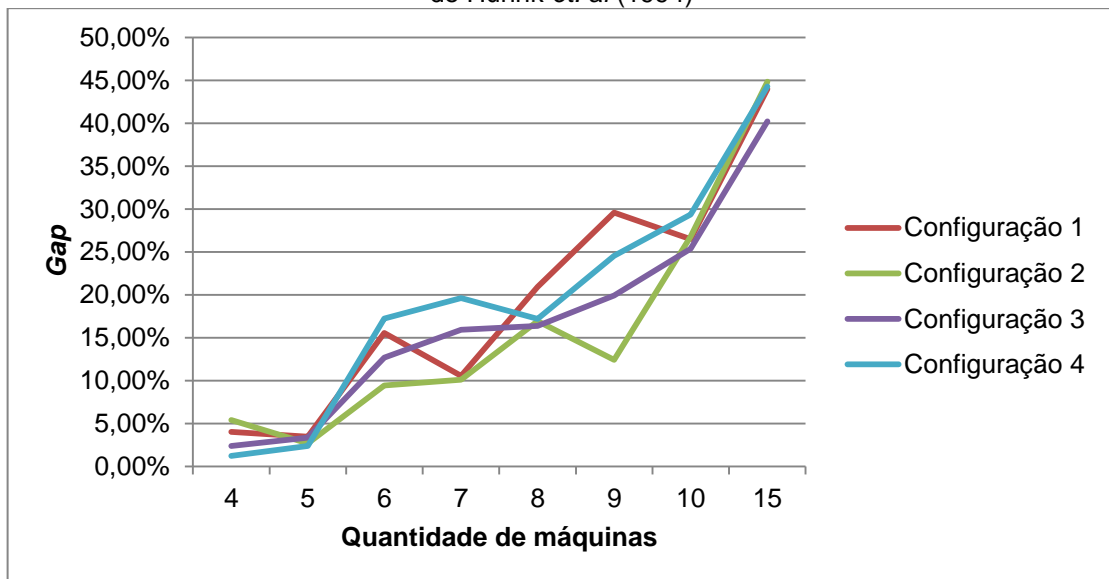
Além disso, gráficos foram plotados a fim de se analisar a influência da quantidade de máquinas, da quantidade de *jobs* e do tamanho das instâncias no *gap* obtido pelo algoritmo com relação aos melhores resultados já encontrados para cada grupo de instâncias utilizados do autor Hurink *et. al* (1994). As Figuras 8, 9 e 10 apresentam os gráficos de comparação entre quantidade de máquinas e *gap* para os grupos de instância *edata*, *rdata* e *vdata*, respectivamente.

Figura 8 - Gráfico comparativo de quantidade de máquinas *versus* gap das instâncias do grupo *edata* de Hurink *et. al* (1994)



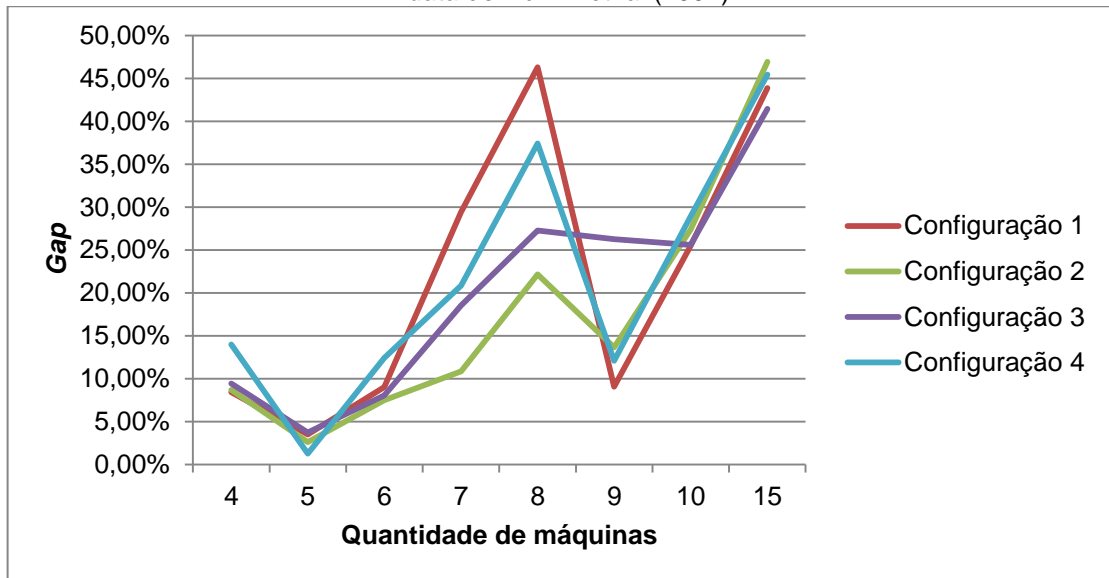
Fonte: Do autor, 2019.

Figura 9 - Gráfico comparativo de quantidade de máquinas *versus* gap das instâncias do grupo *rdata* de Hurink *et. al* (1994)



Fonte: Do autor, 2019.

Figura 10 - Gráfico comparativo de quantidade de máquinas *versus gap* das instâncias do grupo *vdata* de Hurink *et. al* (1994)

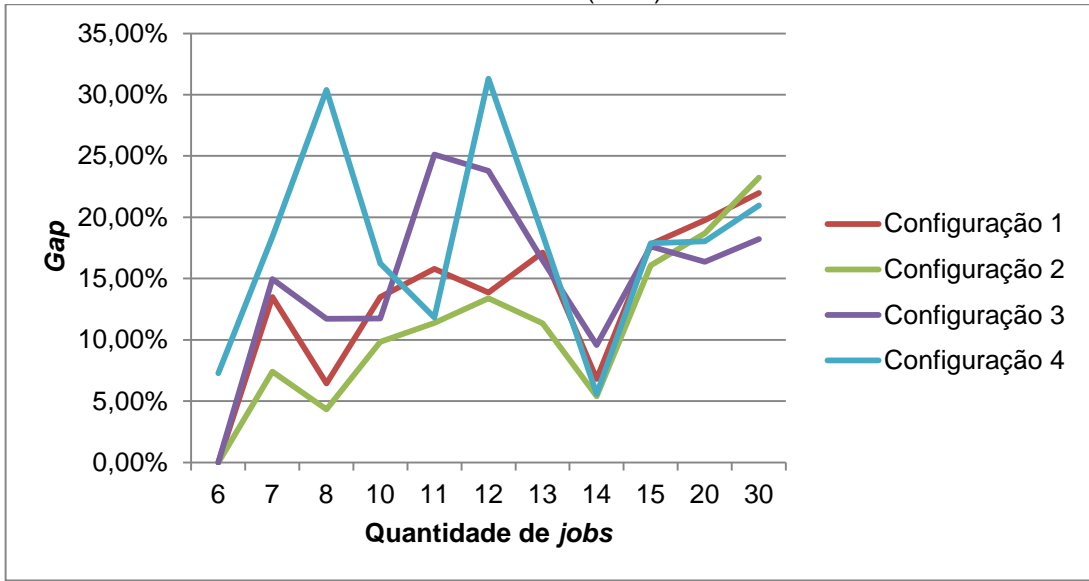


Fonte: Do Autor, 2019.

A Figura 9 evidencia que o aumento do número de máquinas do problema, piora a qualidade da solução, ou seja, o aumento do *gap* conforme maior quantidade de máquinas das instâncias *rdata*. Apesar de alguns pontos discrepantes na configuração quatro da Figura 8 e nas configurações 1 e 4 na Figura 10, indica uma tendência, também, de aumento do *gap* conforme uma maior quantidade de máquinas. Sendo assim, para melhores resultados do algoritmo, o desenvolvimento de novas estruturas de vizinhança para as instâncias de maiores números de máquinas são necessárias.

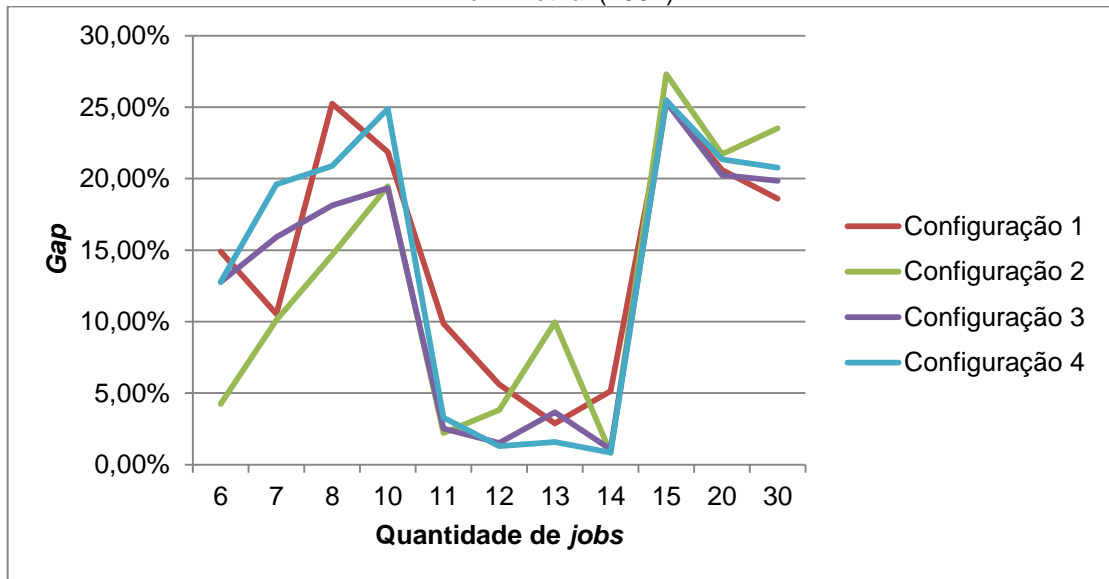
Já com relação ao número de *jobs* de cada instância, conforme gráficos das Figuras 11, 12 e 13, a qualidade da solução não possui relação com o um maior número de *jobs*, ou seja, não foi detectado uma tendência de aumento do *gap* de acordo com o aumento do número de *jobs*.

Figura 11 - Gráfico comparativo de quantidade de *jobs* versus *gap* das instâncias do grupo *edata* de Hurink *et. al* (1994)



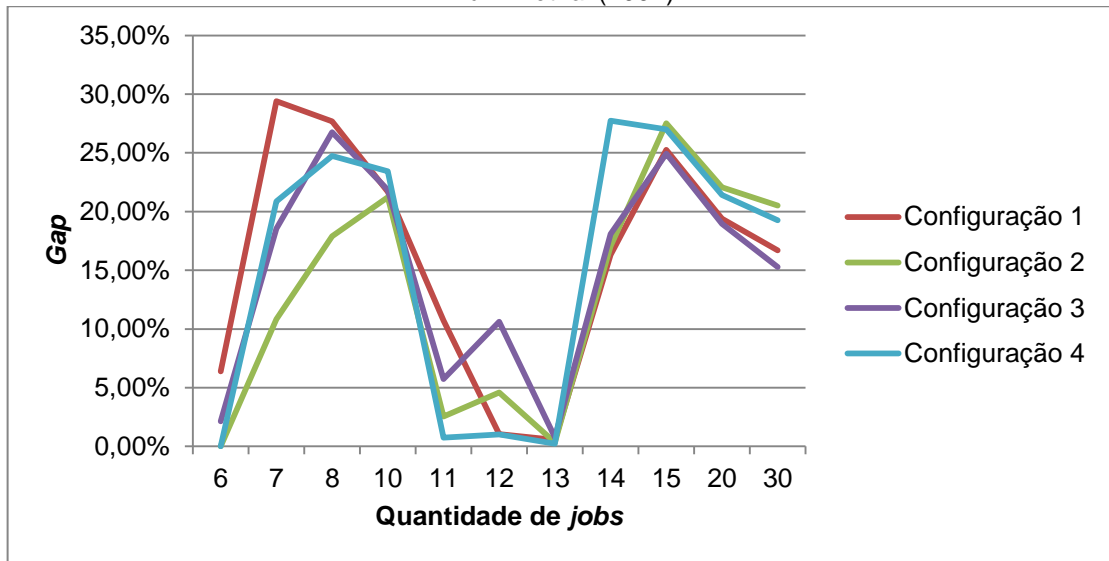
Fonte: Do Autor, 2019.

Figura 12 - Gráfico comparativo de quantidade de *jobs* versus *gap* das instâncias do grupo *rdata* de Hurink *et. al* (1994)



Fonte: Do Autor, 2019.

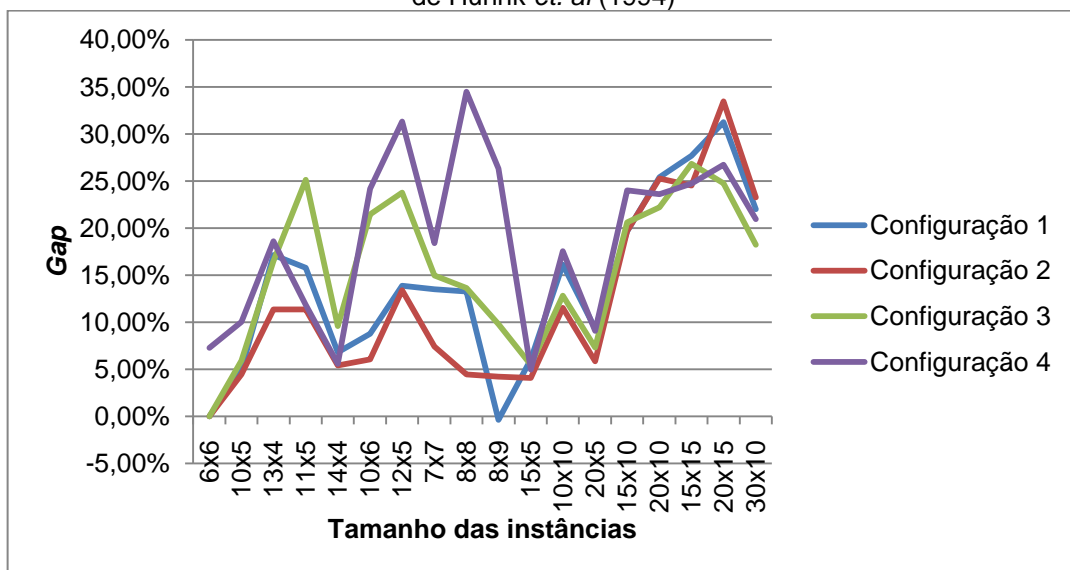
Figura 13 - Gráfico comparativo de quantidade de *jobs* versus *gap* das instâncias do grupo *vdata* de Hurink *et. al* (1994)



Fonte: Do Autor, 2019.

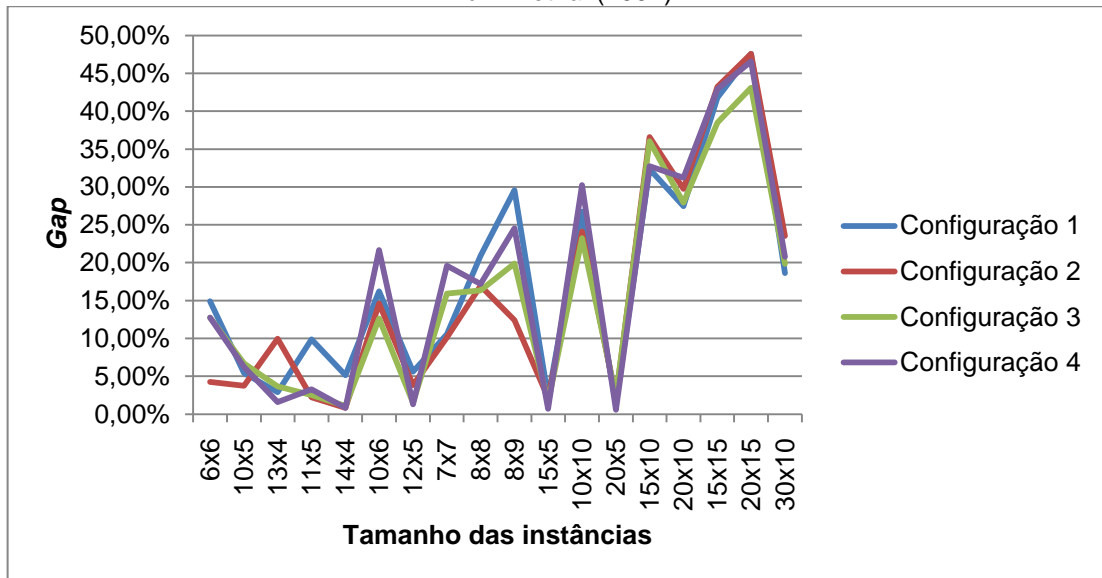
No que diz respeito ao tamanho das instâncias, conforme gráficos das Figuras 14, 15 e 16, também não foi detectado uma tendência de aumento do *gap* de acordo com o aumento do tamanho da instância. Percebe-se, através dos gráficos, que as instâncias de tamanho 8x8, 10x10, 15x10 e 20x15 apresentam os maiores picos, ou seja, apresentam os maiores *gaps* médios para as quatro configurações para os três grupos de instâncias. Isto reforça que instâncias com maiores números de máquinas ocasionam o aumento do *gap*.

Figura 14 - Gráfico comparativo de tamanho da instância versus *gap* das instâncias do grupo *edata* de Hurink *et. al* (1994)



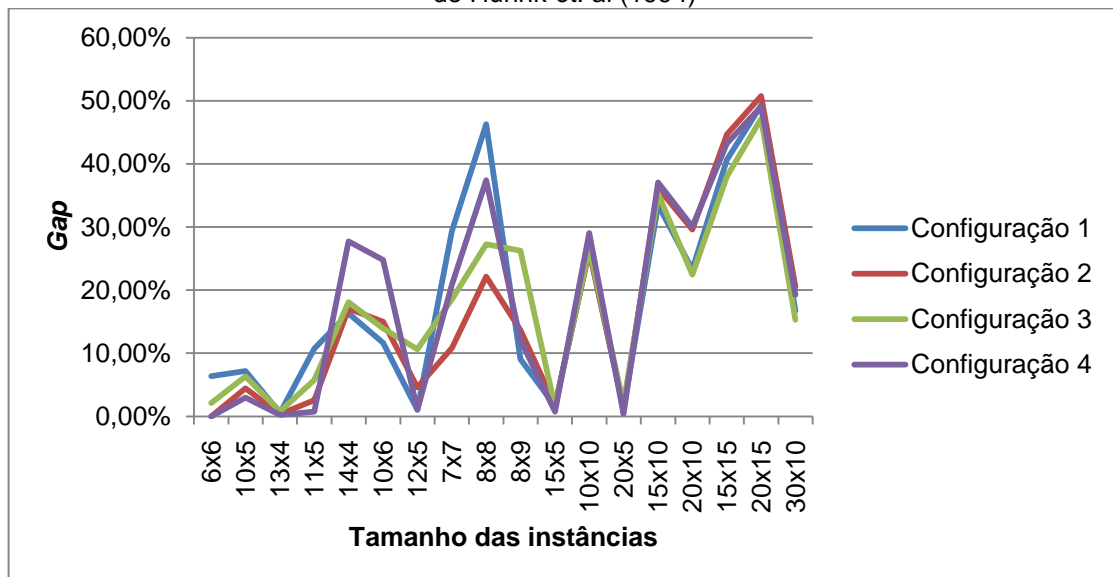
Fonte: Do Autor, 2019

Figura 15 - Gráfico comparativo de tamanho da instância *versus gap* das instâncias do grupo *rdata* de Hurink *et. al* (1994)



Fonte: Do Autor, 2019.

Figura 16 - Gráfico comparativo de tamanho da instância *versus gap* das instâncias do grupo *vdata* de Hurink *et. al* (1994)



Fonte: Do Autor, 2019.

4.4. ANÁLISE DO TEMPO DE EXECUÇÃO DO ALGORITMO

A análise do desempenho do algoritmo contempla, também, a análise do tempo de execução. Para cada configuração e cada grupo de instâncias do autor Hurink *et. al* (1994), calculou-se o tempo médio de execução, em segundos, por

iteração do algoritmo, conforme apresentado na Tabela 11. Assim como os resultados com relação ao *makespan*, os Apêndices D, E, F e G descrevem detalhadamente os resultados obtidos para cada instância do grupo *edata* nas quatro configurações, respectivamente. Os Apêndices H, I, J e K apresentam os resultados obtidos detalhados para cada instância do grupo *rdata* nas quatro configurações, respectivamente, e os Apêndices L, M, N e O demonstram os resultados obtidos detalhados para cada instância do grupo *vdata* nas quatro configurações do algoritmo, respectivamente.

Tabela 11 - Tempo médio (segundos) de execução por iteração do algoritmo

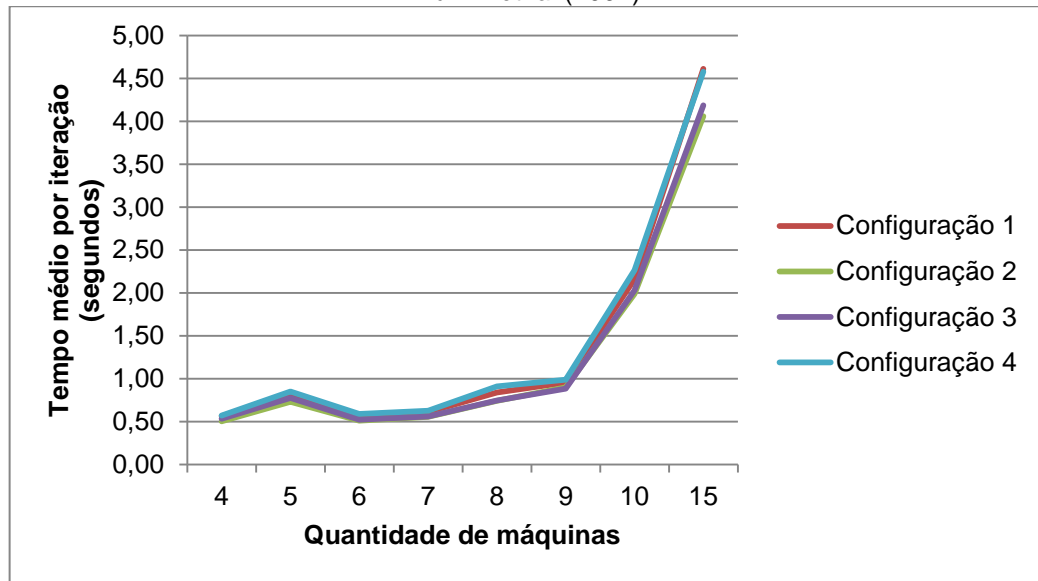
Configurações	1	2	3	4
<i>edata</i>	1,93	1,75	1,80	1,99
<i>rdata</i>	2,01	1,91	1,84	2,12
<i>vdata</i>	2,15	2,05	1,98	2,24

Fonte: Do Autor, 2019

Apesar da similaridade entre as configurações programadas, de acordo com os resultados descritos na Tabela 11, a Configuração 2 apresenta o menor tempo médio de execução por iteração para o conjunto de instâncias *edata*, o que significa que é a configuração mais rápida dentre as quatro configurações para este grupo. Já para os grupos *rdata* e *vdata*, de maior flexibilidade, a Configuração 3 exibe o melhor desempenho com relação à rapidez de execução do algoritmo.

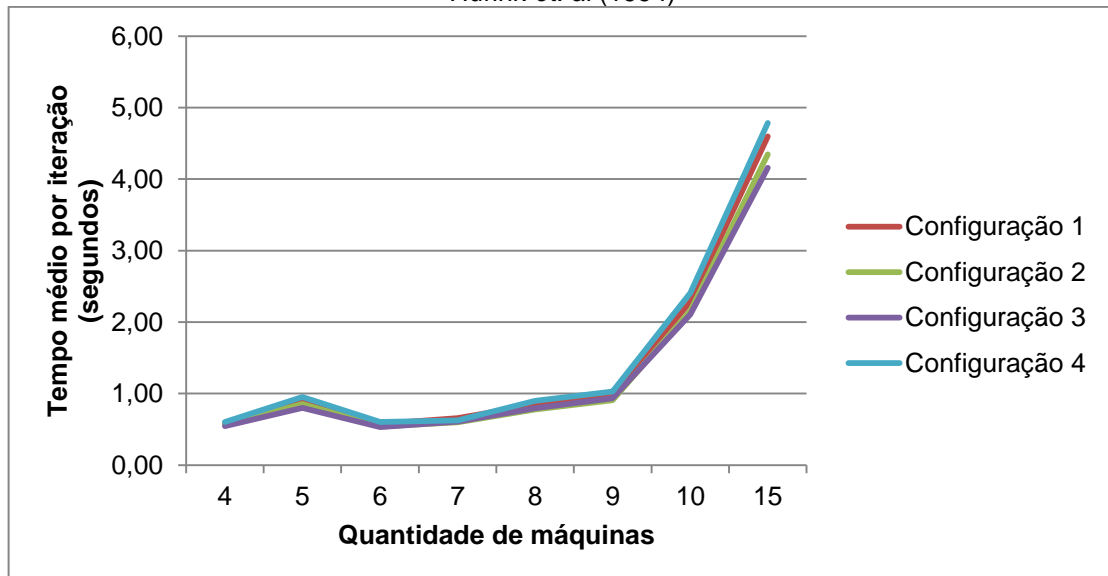
Para a análise do tempo, gráficos também foram plotados a fim de se analisar a relação entre o tempo médio (em segundos) de execução do algoritmo por iteração e as variáveis: quantidade de máquinas, quantidade de *jobs* e quantidade de operações. As Figuras 17, 18 e 19 apresentam a relação entre o número de máquinas e o tempo médio de execução por iteração para as instâncias dos grupos *edata*, *vdata* e *rdata*, respectivamente. As Figuras 20, 21 e 22 apresentam a relação entre o número de *jobs* e o tempo médio de execução por iteração para as instâncias dos grupos *edata*, *vdata* e *rdata*, respectivamente. Já as Figuras 23, 24 e 25 apresentam a relação entre o número de operações e o tempo médio de execução por iteração para as instâncias dos grupos *edata*, *vdata* e *rdata*, respectivamente.

Figura 17 - Gráfico comparativo de quantidade de máquinas *versus* tempo médio do grupo *edata* de Hurink *et. al* (1994)



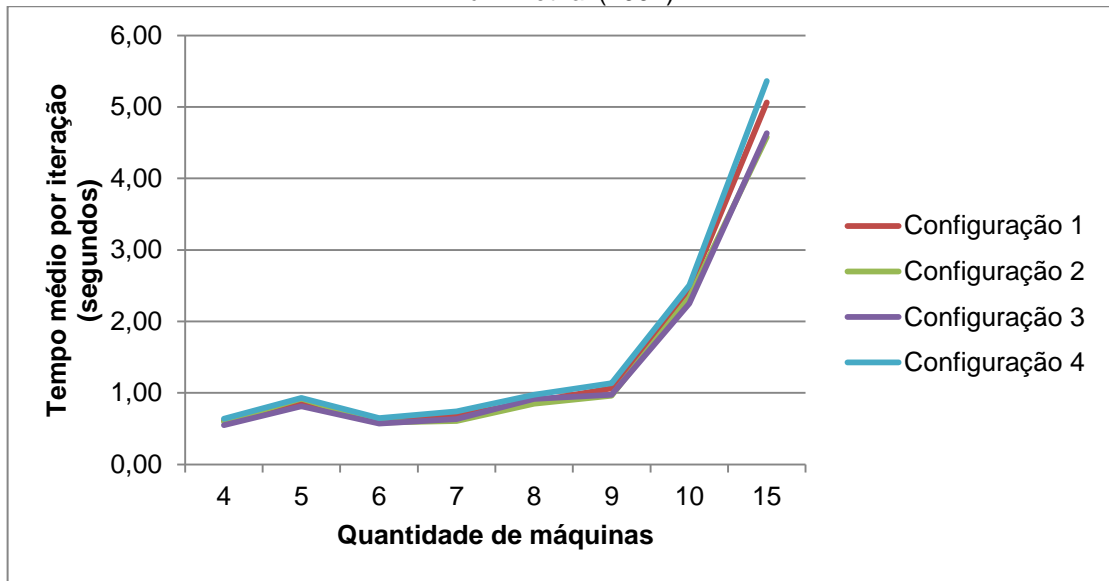
Fonte: Do Autor, 2019.

Figura 18 - Gráfico comparativo de quantidade de máquinas *versus* tempo médio do grupo *rdata* de Hurink *et. al* (1994)



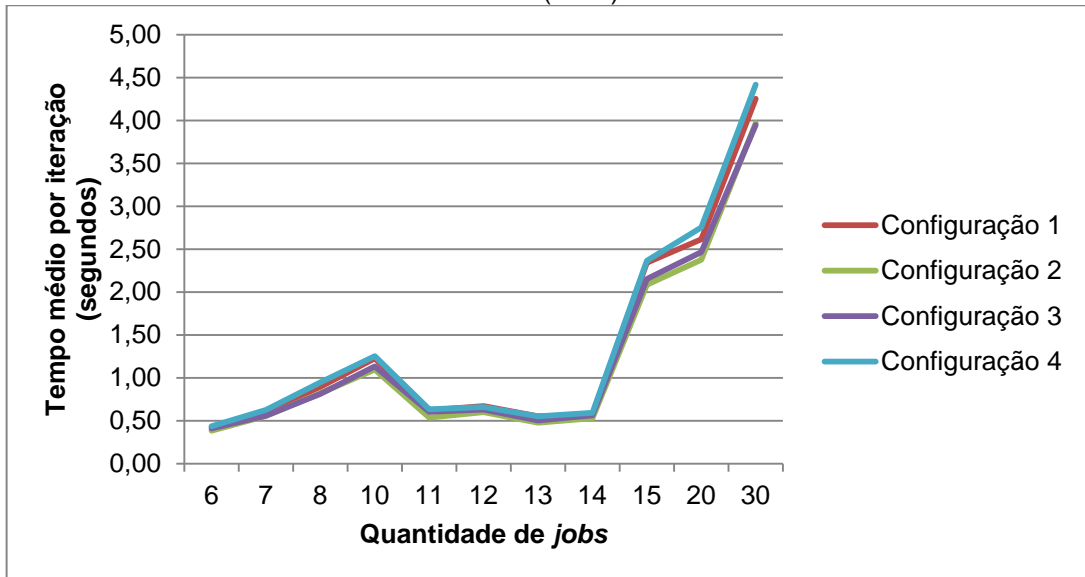
Fonte: Do Autor, 2019.

Figura 19 - Gráfico comparativo de quantidade de máquinas *versus* tempo médio do grupo *vdata* de Hurink *et. al* (1994)



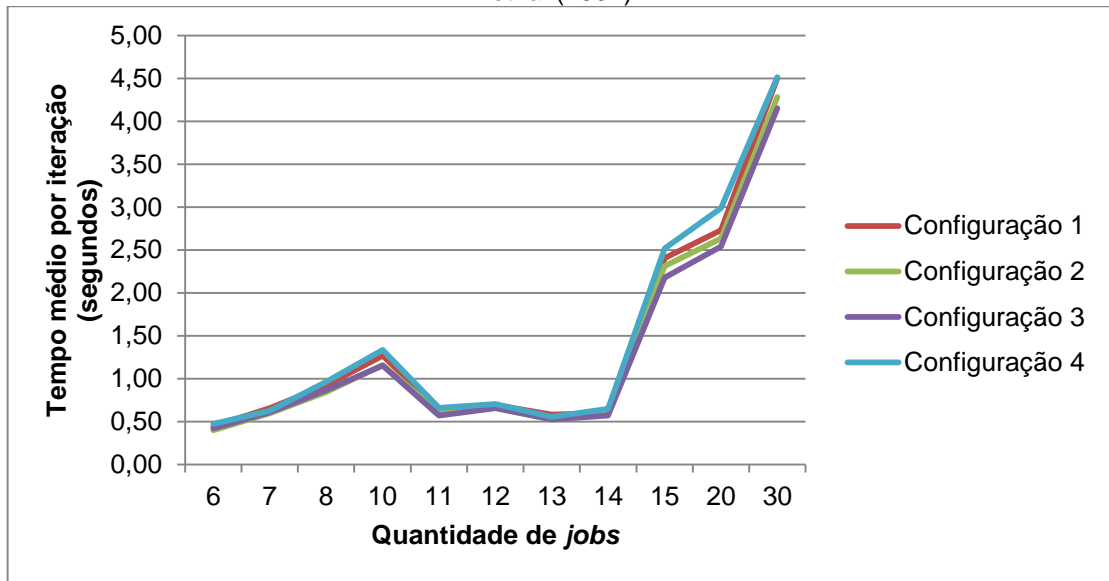
Fonte: Do Autor, 2019.

Figura 20 - Gráfico comparativo de quantidade de *jobs* *versus* tempo médio do grupo *edata* de Hurink *et. al* (1994)



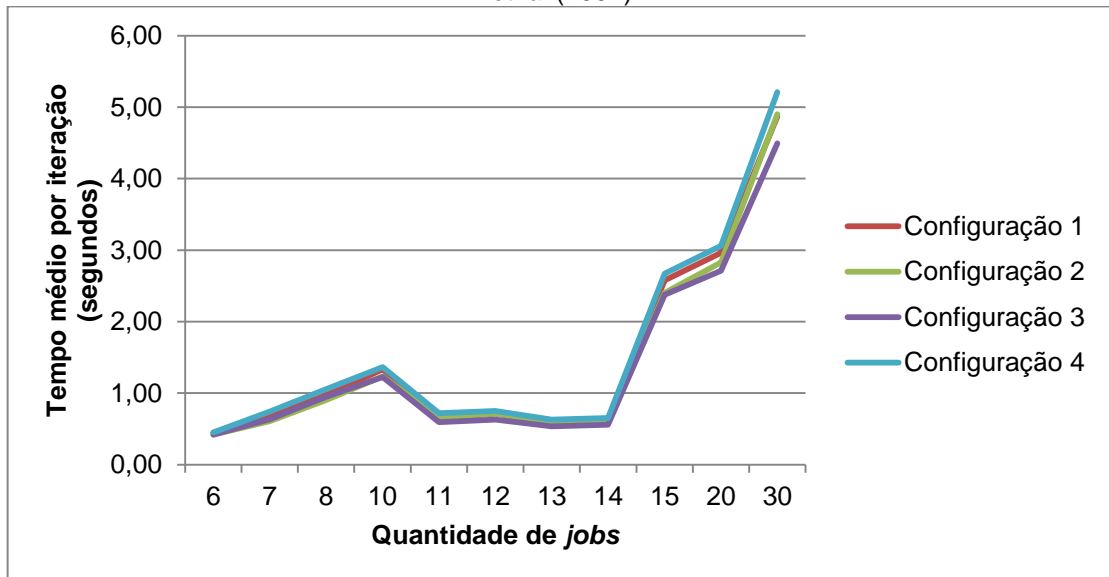
Fonte: Do Autor, 2019.

Figura 21 - Gráfico comparativo de quantidade de *jobs* versus tempo médio do grupo *rdata* de Hurink *et. al* (1994)



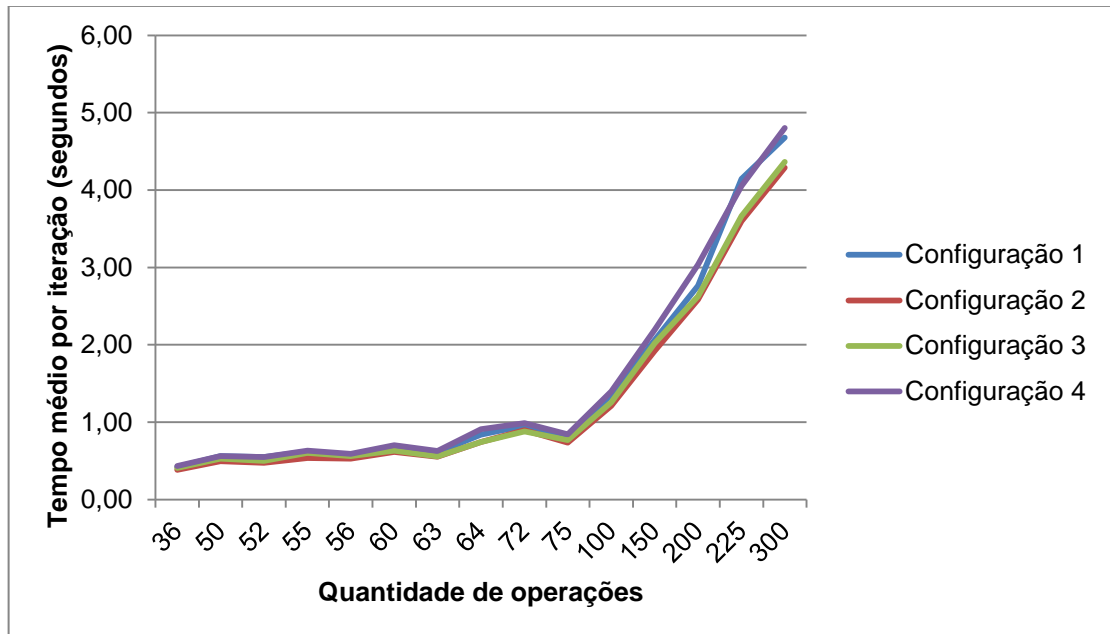
Fonte: Do autor, 2019.

Figura 22 - Gráfico comparativo de quantidade de *de jobs* versus tempo médio do grupo *vdata* de Hurink *et. al* (1994)



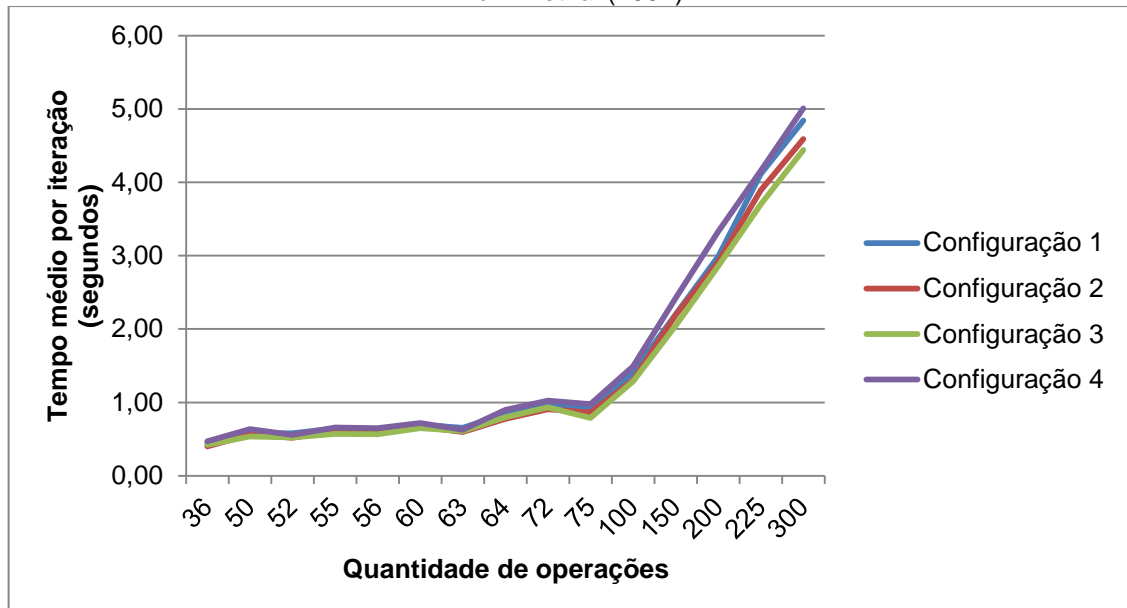
Fonte: Do Autor, 2019.

Figura 23 - Gráfico comparativo de quantidade de operações *versus* tempo médio do grupo *edata* de Hurink *et. al* (1994)



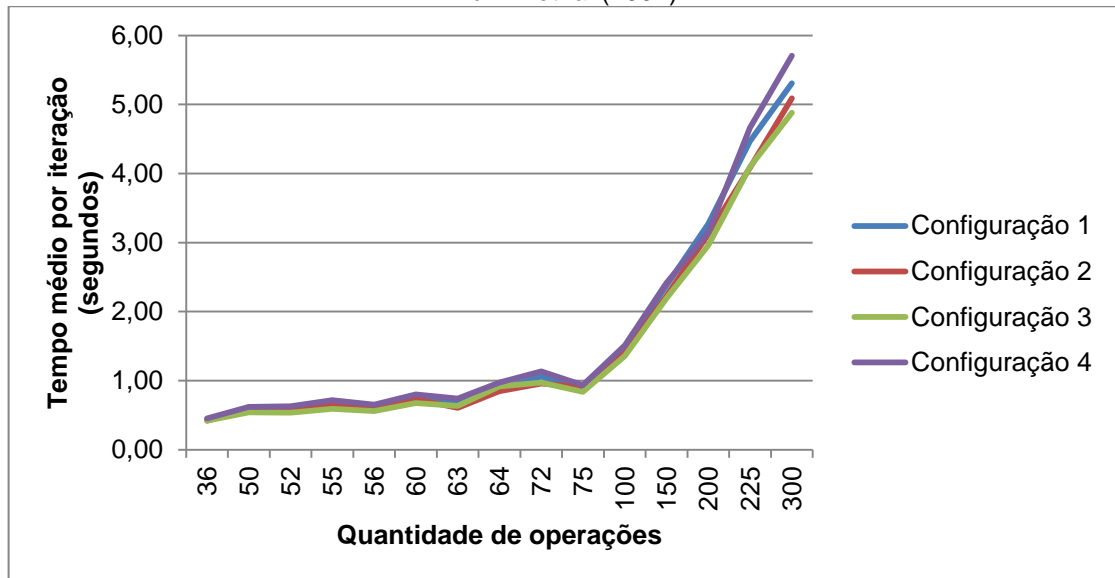
Fonte: Do Autor, 2019.

Figura 24 - Gráfico comparativo de quantidade de operações *versus* tempo médio do grupo *rdata* de Hurink *et. al* (1994)



Fonte: Do Autor, 2019.

Figura 25 - Gráfico comparativo de quantidade de operações *versus* tempo médio do grupo *vdata* de Hurink *et. al* (1994)



Fonte: Do Autor, 2019.

Depreende-se, através dos gráficos contidos nas Figuras de número 17 a 25, que o tempo médio de execução do algoritmo possui relação com a quantidade de máquinas, com a quantidade de *jobs* e com a quantidade de operações, nas quatro configurações, bem como, para os três grupos de instâncias. Ou seja, o aumento dos números de máquinas, *jobs* e operações ocasionam uma tendência de aumento no tempo médio de execução por iteração do algoritmo.

É notável também, através das Figuras 17, 18 e 19, que o tempo médio de execução por iteração do algoritmo eleva-se rapidamente aproximadamente a partir do tempo 1 segundo. Isso é ocasionado devido à explosão combinatória, em que o tamanho do espaço de soluções passa a ser imenso.

5. CASO ILUSTRATIVO

Um estudo a respeito do sistema produtivo de uma metalúrgica foi realizado a fim de se confirmar a viabilidade de aplicação do algoritmo proposto em empresas reais, as quais possuem o sistema produtivo, ou parte dele, caracterizado como *Flexible Jop Shop*.

Na empresa em questão, o layout de chão de fábrica é organizado em células de produção. As principais células do processo produtivo são: o setor de puncionadeiras, que possuem máquinas que realizam tanto processos de corte a laser, como perfuração em forma de estampagem; o setor de dobra, o qual possui diversas máquinas que realizam dobra em chapas e/ou em produtos já semiacabados; e o setor de solda, o qual possui máquinas que realizam o processo de soldagem.

Os roteiros de produção dos itens que passam por esses setores seguem as seguintes possibilidades:

- Punciona, dobra e solda;
- Punciona e dobra;
- Somente punciona;
- Somente dobra;
- Somente solda

Considerando-se os possíveis roteiros de produção para os setores, um exemplo de instância fictícia de tamanho 10x8 foi criado tendo como base as características do processo produtivo em estudo, que consta na Tabela 12.

Tabela 12 - Exemplo de instância com base no processo produtivo em estudo

Job	Operação	Máquinas								Quantidade de máquinas disponíveis
		1	2	3	4	5	6	7	8	
1	1	30	30	25						3
1	2				15		15			2
1	3							10	10	2
2	1	20	19	19						3
3	1		23	20						2
3	2				13					1
4	1	19		16						2

4	2				13				1
4	3					17	17		2
5	1					13	13		2
6	1		22						1
6	2				16				1
7	1	24							1
8	1		32	32					2
8	2				20	19			2
8	3						15	15	2
9	1		40						1
9	2				35				1
9	3						21	21	2
10	1				19	19	19		3

Fonte: Do Autor, 2019

Na instância exemplo criada da Tabela 12, dez *jobs* foram considerados, totalizando 34 operações. As máquinas de número 1, 2 e 3 correspondem às punçoneiras, as máquinas 4, 5 e 6 correspondem às máquinas de dobra e as máquinas 7 e 8 correspondem às máquinas que realizam o processo de solda. Para cada operação, a Tabela 12 mostra a quantidade de máquinas disponíveis para processamento. Os tempos de processamento de cada máquina considerados são em minutos.

A partir da instância criada, uma solução aleatória foi gerada, bem como uma iteração do algoritmo proposto para as quatro configurações foram executadas. A escolha da quantidade de iterações executadas para a geração de solução aleatória foi estipulada com base na simulação de um caso real, já que em empresas que não possuem métodos de sequenciamento, apenas uma solução aleatória é determinada. Para estar equivalente à geração de soluções aleatórias, uma iteração do algoritmo proposto neste trabalho foi executada. Os resultados obtidos constam na Tabela 13.

Tabela 13 - Resultados obtidos através da instância exemplo

Método	Iterações	Resultado Obtido	Tempo de Execução (segundos)
Solução Aleatória	1	148	0,01
Configuração 1	1	111	0,74
Configuração 2	1	128	0,76

Configuração 3	1	128	0,87
Configuração 4	1	128	0,68

Fonte: Do Autor, 2019.

Dessa forma, conforme Tabela 12, depreende-se que o algoritmo proposto é capaz de resolver problemas de sequenciamento de produção reais que possuem características de um processo produtivo do tipo *Job Shop* Flexível, como é o caso da empresa em estudo. É, também, apto a proporcionar resultados melhores do que uma solução gerada aleatoriamente.

Por se tratar de um ambiente bastante flexível, que dispõe de diversas máquinas em paralelo em cada célula, a configuração 3 do algoritmo é a mais adequada para a empresa, em virtude de ter sido a configuração que obteve o melhor desempenho em relação à qualidade da solução e tempo de execução para as instâncias de maior flexibilidade do autor Hurink *et. al* (1994).

Para a aplicação do algoritmo na empresa, algumas ações são necessárias, visto a necessidade de coleta de dados referente aos tempos de processamento de todas as operações realizadas. O correto apontamento de produção, bem como seus registros, considerando tempos de início e fim de processamento das operações de todas as ordens viabilizaria o cálculo de uma estimativa de tempos de processamento das operações nas máquinas, e conseqüentemente, a utilização do algoritmo.

6. CONCLUSÕES

Este trabalho propôs um algoritmo de busca em vizinhança variável, baseado na metaheurística *Variable Neighborhood Search*, para a resolução do problema de sequenciamento de produção em *job shops* flexíveis. Foram implementadas quatro estruturas de vizinhanças, sendo elas: vizinhança baseada em troca de operações adjacentes; vizinhança baseada em retirar uma operação da máquina de maior carga (responsável pela determinação do *makespan*); vizinhança de troca de duas operações; e vizinhança de inversão de subtrecho. Com base nessas estruturas, quatro configurações do algoritmo VNS foram propostas e testadas.

O algoritmo foi testado em três grupos de instâncias do autor Hurink *et. al* (1994) que totalizam 198 instâncias. Apesar da similaridade dos resultados obtidos para as quatro configurações do algoritmo proposto, para o grupo *edata* a Configuração 2 apresentou o melhor desempenho, obtendo-se um *gap* (desvio) médio com relação ao melhor resultado já conhecido na literatura de 13,83%, enquanto que para os grupos *rdata* e *vdata*, a Configuração 3 foi a que obteve os melhores resultados, de 19,71% e 20,45%, respectivamente. Para o grupo *edata* o algoritmo na Configuração 1 apontou um resultado melhor do que o resultado já encontrado para a instância de código *car6†*, com um desvio de -0,39%.

Um caso ilustrativo foi apresentado a fim de se evidenciar que o algoritmo é capaz de resolver instâncias de processos produtivos reais que se caracterizam como FJSP. Uma instância foi criada como exemplo e, a partir dela, uma solução totalmente aleatória foi gerada. Constatou-se que a execução de uma iteração do algoritmo, nas quatro configurações propostas, apresentou melhores resultados quando comparados ao resultado aleatório.

Foi detectado que, para instâncias que possuem maiores quantidades de máquinas, o *gap* obtido é mais elevado, ou seja, para problemas com maior número de máquinas, o algoritmo obtém piores resultados quando comparados aos melhores resultados já conhecidos. Constatou-se, também, que o algoritmo obteve certa dificuldade de escapar de ótimos locais. Sendo assim, para futuros trabalhos, o algoritmo pode ser melhorado através do desenvolvimento de novas estruturas de vizinhança, as quais proporcionem melhores resultados em instâncias com números

de máquinas elevados. Outro fator a ser melhorado é o desenvolvimento de vizinhanças que melhorem o processo de perturbação, a fim de se evitar ótimos locais.

Dessa forma, conclui-se que o algoritmo é capaz de solucionar o problema de sequenciamento de produção em *job shops* flexíveis e de obter bons resultados, os quais se apresentam iguais ou próximos dos melhores resultados já encontrados na literatura.

REFERÊNCIAS

- AMIRI, M., ZANDIEH, M., YAZDANI, M., BAGHERI, A.. **A variable neighbourhood search algorithm for the flexible job-shop scheduling problem.** *International Journal of Production Research*, v. 48, n. 12, p. 5671–5689, 2010.
- BAGHERI, A.; ZANDIEH, M.. **Bi-criteria flexible job-shop scheduling with sequence-dependent setup times – variable neighborhood search approach.** *Journal of Manufacturing Systems*, v. 30, n. 1, p. 8–15, 2011.
- BEHNKE, D.; GEIGER, M. J.. **Test instances for the flexible job shop scheduling problem with work centers.** Helmut-Schmidt-University, Logistics Management Department, Hamburgo, Alemanha. 2012.
- BEKKAR, A.; GUEMRI, O.; BEKRAR, ABDELGHANI; AISSANI, N.; BELDJILALI, B.; TRENTESAUX, DAMIEN. **An Iterative Greedy Insertion Technique For Flexible Job Shop Scheduling.** *IFAC-PapersOnline*, v. 49, n. 12, p. 1956-1961, 2016.
Disponível em:
https://www.academia.edu/30738233/An_Iterative_Greedy_Insertion_Technique_for_Flexible_Job_Shop_Scheduling_Problem?auto=download. Acesso em: 28 abril 2019.
- CHAUDRY, Imran Ali; KHAN, Abid Ali. **A research survey: review of flexible job shop scheduling techniques.** *Internacional Transactions in Operational Research*. v. 23, p. 551–591, 2016. Disponível em:
<https://onlinelibrary.wiley.com/doi/pdf/10.1111/itor.12199>. Acesso em: 09 junho de 2019.
- CORRÊA, Henrique L; CORRÊA, Carlos A. **Administração de produção e operações: manufatura e serviços: uma abordagem estratégica.** 4. ed. 2. Reimpr. São Paulo: Atlas, 2019.
- CORRÊA, Henrique Luiz; GIANESI, Irineu Gustavo Nogueira; CAON, Mauro. **Planejamento, programação e controle da produção: MRP II / ERP: conceitos, uso e implantação: base para SAP, oracle applications e outros softwares integrados de gestão.** 6. ed. São Paulo : Atlas, 2019.
- FATTAHI, P.; MEHRABAD, M. S.; JOLAI, F.. **Mathematical modeling and heuristic approaches to flexible job shop scheduling problems.** *Journal of intelligent manufacturing*, v. 18, p. 331-342, 2007. Disponível em:
<https://link.springer.com/article/10.1007/s10845-007-0026-8>. Acesso em: 28 abril 2019.
- GAO, J.; SUN, L.; GEN, M.. **A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems.** *Computers & Operations Research*, v. 35, p. 2892-2907, 2008. Disponível em
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.158.9239&rep=rep1&type=pdf>. Acesso em: 06 junho 2019.

GLOVER, Fred; KOCHENBER, Gary A.. **Handbook of Metaheuristics**. Springer Science & Business Media, 2003.

HURINK, E., JURISCH, B., THOLE, M.. **Tabu search for the job shop scheduling problem with multi-purpose machines**. Operations Research Spectrum, 15, 205–215, 1994. Disponível em: <https://link.springer.com/article/10.1007/BF01719451>. Acesso em: 03 Setembro 2019.

KUHPFAHL, Jens. **Job Shop Scheduling with Consideration of Due Dates**. Springer Glaber, 2016.

LEI, D.; GUO, X. GUO. **Variable neighbourhood search for dual-resource constrained flexible job shop scheduling**. International Journal of Production Research, v. 52, n. 9, p. 2519-2529, 2014.

LI, X.; PENG, Z.; DU, B.; JUN, G.; WENXIANG, X.; ZHUANG, K.. **Hybrid artificial bee colony algorithm with a rescheduling strategy for solving flexible job shop scheduling problems**. Computers & Industrial Engineering, v. 113, p. 10-26, 2017. Disponível em: https://www.researchgate.net/publication/319610465_Hybrid_artificial_bee_colony_algorithm_with_a_rescheduling_strategy_for_solving_flexible_job_shop_scheduling_problems. Acesso em 09 junho 2019.

MASTROLILLI, M.. **Flexible Job Shop Problem**. <http://people.idsia.ch/~monaldo/fjsp.html>: [s.n.], 2018. Acesso em: 03 setembro 2019.

PEZZELLA, F.; MORGANTI, G.; CIASCETTI, G.. **A genetic algorithm for the flexible job-shop scheduling problem**. Computers & Operations Research, v. 35, n. 10, p. 3202-3212, 2008. Disponível em <https://www.sciencedirect.com/science/article/pii/S0305054807000524>. Acesso em: 09 junho 2019.

PINEDO, Michael L.. **Scheduling: Theory, Algorithms, and Systems**. Springer, 2016.

RIGO, P. M. **Desenvolvimento de um Algoritmo Genético para resolução do problema de sequenciamento de produção em job shops flexíveis**. 2018. 60f. Trabalho de Conclusão de Curso – Universidade Tecnológica Federal do Paraná, Londrina, 2018.

SULE, Dileep R.. **Production Planning and Industrial Scheduling: examples, case studies, and applications**. 2. ed. CRC Press, 2008.

TALBI, E.. **Metaheuristics: from design to implemetation**. John Wiley & Sons, Inc., Hoboken, New Jersey, 2009.

TAMSSAOUET, K.; DAUZÈRE-PÉRÈS, S.; YUGMA, G. C.. **Metaheuristics for the Job-Shop Scheduling Problem with Machine Availability Constraints**. Computers & Industrial Engineering, v.125, p. 1-8, 2018.

TEEKENG, W.; THAMMANO, A.. **Modified Genetic Algorithm for Flexible Job-Shop Scheduling Problems**. *Procedia Computer Science*, v. 12, p. 122-128, 2012. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877050912006321>. Acesso em 09 junho 2019.

TUBINO, Dalvio Ferrari. **Planejamento e controle da produção: teoria e prática**. 3. ed. São Paulo: Atlas, 2017.

WANG, S.; WANG, L.; ZHOU G.; XU, Y. **An Estimation of Distribution Algorithm for the Flexible Job-Shop Scheduling Problem**. Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Automation Tsinghua University, P.R. China, 2012.

YUAN, Y.; XU, H.; YANG, J.. **A hybrid harmony search algorithm for the flexible job shop scheduling problem**. *Applied Soft Computing*, v. 13, p. 3259-3272, 2013. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494613000677>. Acesso em: 09 junho 2019.

ZHANG, G.; GAO, L.; SHI, Y... **An effective genetic algorithm for the flexible job-shop scheduling problem**. *Expert Systems with Applications*, v. 38, p. 3563-3573, 2011. Disponível em: <https://www.sciencedirect.com/science/article/pii/S095741741000953X>. Acesso em: 09 junho 2019.

ZHANG, G.; WANG Y.. **Research on the Flexible Job Shop Scheduling Problem Using Drum-Buffer-Rope Approach**. In: *Proceedings of the 6th International Asia Conference on Industrial Engineering and Management Innovation*. p.911-919. Atlantis Press, Paris, 2016.

ZHANG, G.; ZHANG, L.; SONG, X.; WANG, Y.; ZHOU, C. (2018). **A variable neighborhood search based genetic algorithm for flexible job shop scheduling problem**. *Cluster Computing*, v. 22, p. 11561-11572, 2019.

APÊNDICE A – TABELA DE RESULTADOS DE *MAKESPAN* OBTIDOS NO TESTE DE VIZINHANÇAS COM O GRUPO DE INSTÂNCIAS *RDATA* DO AUTOR HURINK *ET. AL* (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Vizinhanças							
				V_1		V_2		V_3		V_4	
				Resultado Obtido	Gap	Resultado Obtido	Gap	Resultado Obtido	Gap	Resultado Obtido	Gap
mt06†	6x6	36	47	50	6,38%	50	6,38%	50	6,38%	50	6,38%
mt10†	10x10	100	686	938	36,73%	861	25,51%	924	34,69%	923	34,55%
mt20†	20x5	100	1022	1182	14,29%	1174	15,66%	1164	13,89%	1168	14,29%
la01‡	10x5	50	571	659	15,41%	640	12,08%	646	13,13%	674	18,04%
la02‡	10x5	50	530	610	15,09%	613	15,66%	605	14,15%	606	14,34%
la03‡	10x5	50	478	562	17,57%	551	15,27%	556	16,32%	547	14,44%
la04‡	10x5	50	502	595	18,53%	584	16,33%	582	15,94%	584	16,33%
la05‡	10x5	50	457	524	14,66%	522	14,22%	500	9,41%	519	13,57%
la06‡	15x5	75	799	878	9,89%	894	11,89%	899	12,52%	901	12,77%
la07‡	15x5	75	750	824	9,87%	807	7,60%	833	11,07%	852	13,60%
la08‡	15x5	75	765	852	11,37%	869	13,59%	861	12,55%	854	11,63%
la09‡	15x5	75	853	947	11,02%	962	12,78%	938	9,96%	973	14,07%
la10‡	15x5	75	804	911	13,31%	896	11,44%	884	9,95%	918	14,18%
la11‡	20x5	100	1071	1186	10,74%	1173	9,52%	1168	9,06%	1169	9,15%
la12‡	20x5	100	936	1046	11,75%	1016	8,55%	1033	10,36%	1038	10,90%
la13‡	20x5	100	1038	1156	11,37%	1141	9,92%	1103	6,26%	1119	7,80%
la14‡	20x5	100	1070	1173	9,63%	1163	8,69%	1145	7,01%	1200	12,15%
la15‡	20x5	100	1090	1212	11,19%	1187	8,90%	1226	12,48%	1207	10,73%
la16‡	10x10	100	717	959	33,75%	956	33,33%	929	29,57%	946	31,94%
la17‡	10x10	100	646	821	27,09%	858	32,82%	846	30,96%	845	30,80%
la18‡	10x10	100	666	878	31,83%	921	38,29%	887	33,18%	900	35,14%
la19‡	10x10	100	700	987	41,00%	976	39,43%	1003	43,29%	994	42,00%

la20†	10x10	100	756	944	24,87%	998	32,01%	960	26,98%	1003	32,67%
la21†	15x10	150	835	1227	46,95%	1208	44,67%	1224	46,59%	1188	42,28%
la22†	15x10	150	760	1157	52,24%	1141	50,13%	1142	50,26%	1163	53,03%
la23†	15x10	150	842	1261	49,76%	1209	43,59%	1213	44,06%	1195	41,92%
la24†	15x10	150	808	1202	48,76%	1194	47,77%	1182	46,29%	1186	46,78%
la25†	15x10	150	791	1171	48,04%	1186	49,94%	1168	47,66%	1132	43,11%
la26†	20x10	200	1061	1435	35,25%	1523	43,54%	1485	39,96%	1461	37,70%
la27†	20x10	200	1091	1534	40,60%	1549	41,98%	1548	41,89%	1574	44,27%
la28†	20x10	200	1080	1511	39,91%	1568	45,19%	1517	40,46%	1518	40,56%
la29†	20x10	200	998	1437	43,99%	1385	38,78%	1433	43,59%	1438	44,09%
la30†	20x10	200	1078	1527	41,65%	1535	42,39%	1517	40,72%	1546	43,41%
la31†	30x10	300	1521	2009	32,08%	1983	30,37%	2044	34,39%	1976	29,91%
la32†	30x10	300	1659	2161	30,26%	2192	32,13%	2199	32,55%	2084	25,62%
la33†	30x10	300	1499	1998	33,29%	2016	34,49%	1963	30,95%	1920	28,09%
la34†	30x10	300	1536	2011	30,92%	1979	28,84%	2001	30,27%	1927	25,46%
la35†	30x10	300	1550	2061	32,97%	1981	27,81%	1940	25,16%	2053	32,45%
la36†	15x15	225	1030	1547	50,19%	1583	53,69%	1532	48,74%	1565	51,94%
la37†	15x15	225	1077	1652	53,39%	1709	58,68%	1656	53,76%	1718	59,52%
la38†	15x15	225	962	1508	56,76%	1459	51,66%	1482	54,05%	1497	55,61%
la39†	15x15	225	1024	1512	47,66%	1569	53,22%	1516	48,05%	1547	51,07%
la40†	15x15	225	970	1513	55,98%	1509	55,57%	1482	52,78%	1501	54,74%
abz5†	10x10	100	962	1308	35,97%	1338	39,09%	1283	33,37%	1264	31,39%
abz6†	10x10	100	807	1029	27,51%	1039	28,75%	1029	27,51%	1018	26,15%
abz7†	20x15	300	544	811	49,08%	874	60,66%	840	54,41%	790	45,22%
abz8†	20x15	300	555	861	55,14%	905	63,06%	854	53,87%	877	58,02%
abz9†	20x15	300	562	855	52,14%	860	53,02%	866	54,09%	879	56,41%
car1†	11x5	55	5057	5983	18,31%	5801	14,71%	6002	18,69%	5998	18,61%
car2†	13x4	52	5987	6601	10,26%	6433	7,45%	6527	9,02%	6720	12,24%
car3†	12x5	60	5626	6818	21,19%	6599	17,29%	6599	17,29%	6763	20,21%
car4†	14x4	56	6518	7501	15,08%	7253	11,28%	7071	8,48%	7458	14,42%

car5‡	10x6	60	5764	7163	24,27%	7231	25,45%	7244	25,68%	7372	27,90%
car6‡	8x9	72	6147	7946	29,27%	7901	28,53%	8171	32,93%	7837	27,49%
car7‡	7x7	63	4432	5501	24,12%	5416	22,20%	5570	25,68%	5330	20,26%
car8‡	8x8	64	5692	7144	25,51%	6961	22,29%	6966	22,38%	7160	25,79%
orb1††	10x10	100	763	1012	32,63%	1010	32,37%	999	30,93%	1004	31,59%
orb2††	10x10	100	703	928	32,01%	920	30,87%	937	33,29%	939	33,57%
orb3††	10x10	100	720	986	36,94%	1034	43,61%	958	33,06%	1021	41,81%
orb4††	10x10	100	753	1027	36,39%	1016	34,93%	970	28,82%	974	29,35%
orb5††	10x10	100	643	907	41,06%	921	43,23%	899	39,81%	900	39,97%
orb6††	10x10	100	766	1028	34,20%	1035	35,12%	1021	33,29%	1054	37,60%
orb7††	10x10	100	302	405	34,11%	424	40,40%	409	35,43%	422	39,74%
orb8††	10x10	100	651	896	37,63%	880	35,18%	847	30,11%	917	40,86%
orb9††	10x10	100	694	954	37,46%	967	39,34%	950	36,89%	976	40,63%
orb10††	10x10	100	750	1041	38,80%	1045	39,33%	981	30,80%	1023	36,40%

Fonte: Adaptado de Rigo (2018).

APÊNDICE B - TABELA DE RESULTADOS RELACIONADOS À EXECUÇÃO DO *HILL-CLIMBING* OBTIDOS NOS TESTES DAS VIZINHANÇAS V_1 E V_2 COM O GRUPO DE INSTÂNCIAS *RDATA* DO AUTOR HURINK *ET. AL* (1994)

Código Instância	nxm	Quantidade total de operações	Vizinhanças							
			V_1				V_2			
			Iterações Realizadas	Iteração do Resultado	Número de Melhorias até o ótimo local	Tempo de Execução (segundos)	Quantidade Iterações	Iteração do Resultado	Número de Melhorias até o ótimo local	Tempo de Execução (segundos)
mt06†	6x6	36	1960	1947	7	100,03	1026	497	7	100,03
mt10†	10x10	100	620	555	10	100,03	317	281	8	100,05
mt20†	20x5	100	805	794	10	100,05	334	332	5	100,35
la01‡	10x5	50	1526	1062	7	100,05	657	581	9	100,02
la02‡	10x5	50	1553	398	7	100,03	622	237	8	100,10
la03‡	10x5	50	1559	1070	6	100,06	747	604	6	100,14
la04‡	10x5	50	1529	279	9	100,05	671	361	10	100,25
la05‡	10x5	50	1503	1403	8	100,03	670	279	10	100,14
la06‡	15x5	75	1084	120	10	100,03	408	95	5	100,07
la07‡	15x5	75	850	654	10	100,12	436	49	8	100,38
la08‡	15x5	75	1060	1015	8	100,07	408	78	3	100,10
la09‡	15x5	75	1063	107	5	100,04	451	183	5	100,07
la10‡	15x5	75	1056	706	9	100,07	421	378	6	100,18
la11‡	20x5	100	816	606	8	100,10	309	283	11	100,03
la12‡	20x5	100	814	95	7	100,03	276	185	6	100,09
la13‡	20x5	100	804	684	6	100,02	293	147	6	100,03
la14‡	20x5	100	846	608	10	100,09	294	233	5	100,31
la15‡	20x5	100	659	378	5	100,09	355	353	8	100,38
la16‡	10x10	100	625	204	6	100,19	327	248	8	100,01
la17‡	10x10	100	611	573	11	100,05	327	147	10	100,15
la18‡	10x10	100	615	377	8	100,06	283	260	7	100,61

la19‡	10x10	100	646	43	4	100,12	300	9	4	100,25
la20‡	10x10	100	637	524	10	100,10	331	124	8	100,05
la21‡	15x10	150	355	338	7	100,12	204	204	5	100,72
la22‡	15x10	150	391	354	5	100,19	208	14	3	100,10
la23‡	15x10	150	405	182	6	100,08	198	19	5	100,40
la24‡	15x10	150	381	282	5	100,22	122	111	5	100,44
la25‡	15x10	150	414	225	12	100,19	182	53	8	100,37
la26‡	20x10	200	312	196	6	100,22	147	8	4	100,59
la27‡	20x10	200	267	21	3	100,19	148	108	7	100,10
la28‡	20x10	200	328	101	5	100,14	151	40	2	100,43
la29‡	20x10	200	316	221	3	100,01	149	10	3	100,40
la30‡	20x10	200	312	75	6	100,36	134	39	3	100,39
la31‡	30x10	300	224	161	8	100,12	101	90	5	100,61
la32‡	30x10	300	224	124	4	100,03	101	2	2	100,45
la33‡	30x10	300	190	37	2	100,04	71	16	6	101,25
la34‡	30x10	300	209	78	3	100,29	89	58	8	100,73
la35‡	30x10	300	214	25	3	100,20	93	69	5	100,06
la36‡	15x15	225	238	2	2	100,02	118	48	8	100,41
la37‡	15x15	225	233	185	6	100,36	118	45	4	100,45
la38‡	15x15	225	229	111	7	100,15	104	58	5	100,51
la39‡	15x15	225	233	51	4	100,57	109	100	7	100,35
la40‡	15x15	225	214	97	4	100,38	123	93	6	100,40
abz5†	10x10	100	640	266	7	100,13	268	126	3	100,23
abz6†	10x10	100	649	245	4	100,01	264	93	3	100,03
abz7†	20x15	300	142	93	7	100,29	92	92	4	100,21
abz8†	20x15	300	178	143	8	100,43	94	60	5	100,05
abz9†	20x15	300	179	7	3	100,38	89	60	8	100,63
car1‡	11x5	55	1331	1006	9	100,02	524	292	9	100,09
car2‡	13x4	52	1547	831	5	100,01	557	69	6	100,21
car3‡	12x5	60	1330	12	4	100,12	533	62	6	100,10

car4‡	14x4	56	1500	88	5	100,03	578	399	10	100,06
car5‡	10x6	60	1129	651	10	100,07	573	335	8	100,18
car6‡	8x9	72	909	140	8	100,05	467	160	6	100,11
car7‡	7x7	63	1389	1203	13	100,03	732	538	8	100,04
car8‡	8x8	64	1030	202	7	100,02	519	511	5	100,01
orb1††	10x10	100	625	336	7	100,11	320	16	5	100,11
orb2††	10x10	100	560	473	6	100,09	322	133	6	100,25
orb3††	10x10	100	622	513	8	100,14	320	119	12	100,05
orb4††	10x10	100	524	447	8	100,20	327	211	4	100,22
orb5††	10x10	100	638	15	6	100,08	304	124	4	100,26
orb6††	10x10	100	597	200	7	100,06	337	202	9	100,03
orb7††	10x10	100	627	246	5	100,09	291	60	4	100,27
orb8††	10x10	100	587	410	9	100,04	310	282	7	100,03
orb9††	10x10	100	618	251	6	100,11	316	235	8	100,12
orb10††	10x10	100	626	145	4	100,14	316	120	4	100,22

Fonte: Adaptado de Rigo (2018).

APÊNDICE C - TABELA DE RESULTADOS RELACIONADOS À EXECUÇÃO DO *HILL-CLIMBING* OBTIDOS NOS TESTES DAS VIZINHANÇAS V_3 E V_4 COM O GRUPO DE INSTÂNCIAS *RDATA* DO AUTOR HURINK *ET. AL* (1994)

Código Instância	nxm	Quantidade total de operações	Vizinhanças							
			V_3				V_4			
			Iterações Realizadas	Iteração do Resultado	Número de Melhorias até o ótimo local	Tempo de Execução (segundos)	Iterações Realizadas	Iteração do Resultado	Número de Melhorias até o ótimo local	Tempo de Execução (segundos)
mt06†	6x6	36	1109	6	4	100,10	1881	118	6	100,03
mt10†	10x10	100	253	20	6	101	542	461	6	100
mt20†	20x5	100	324	155	8	100	748	171	4	100
la01‡	10x5	50	690	374	7	100	1378	695	8	100
la02‡	10x5	50	751	268	5	100	1413	672	8	100
la03‡	10x5	50	732	171	3	100	1454	165	9	100
la04‡	10x5	50	709	409	9	100	584	717	7	100
la05‡	10x5	50	746	578	7	100	1277	116	8	100
la06‡	15x5	75	431	198	4	100	984	912	6	100,04
la07‡	15x5	75	446	33	4	100	987	309	4	100,04
la08‡	15x5	75	478	154	6	100	971	670	8	100,02
la09‡	15x5	75	448	194	7	100	991	134	9	100,03
la10‡	15x5	75	442	208	4	100	965	50	5	100,09
la11‡	20x5	100	324	216	9	100	773	19	5	100,05
la12‡	20x5	100	341	201	7	100	746	711	9	100,08
la13‡	20x5	100	322	117	7	100	661	199	9	100,01
la14‡	20x5	100	349	201	8	100	718	526	6	100,12
la15‡	20x5	100	343	199	10	100	770	681	5	100,06
la16‡	10x10	100	263	262	7	101	565	397	7	100,03
la17‡	10x10	100	261	44	4	100	541	322	14	100,21
la18‡	10x10	100	265	93	4	100	535	37	6	100,18

la19‡	10x10	100	259	151	6	100,23	535	257	5	100,23
la20‡	10x10	100	258	41	6	100	556	320	4	100,13
la21‡	15x10	150	171	105	5	100,32	380	333	9	100,19
la22‡	15x10	150	174	20	3	100,28	360	80	5	100,11
la23‡	15x10	150	143	2	2	100	349	163	6	100,26
la24‡	15x10	150	206	143	4	100	372	148	9	100,06
la25‡	15x10	150	142	122	11	100	357	125	5	100,05
la26‡	20x10	200	124	73	6	100	294	103	6	100
la27‡	20x10	200	103	25	6	101	268	102	6	100,33
la28‡	20x10	200	138	103	9	100	313	179	5	100,16
la29‡	20x10	200	142	142	5	100	294	189	6	100,15
la30‡	20x10	200	135	13	4	100	307	213	7	100,16
la31‡	30x10	300	77	35	6	101	178	38	7	100,38
la32‡	30x10	300	93	1	1	101	210	94	4	100,28
la33‡	30x10	300	110	32	5	101	212	19	5	100,18
la34‡	30x10	300	108	99	3	100	208	178	13	100,13
la35‡	30x10	300	102	76	5	100	196	38	6	100,62
la36‡	15x15	225	84	17	2	102	200	152	3	100,17
la37‡	15x15	225	82	57	7	100	187	17	4	100,18
la38‡	15x15	225	94	4	4	102	186	119	7	100,26
la39‡	15x15	225	97	52	9	101	200	31	5	100,54
la40‡	15x15	225	102	33	6	103	211	95	7	100,27
abz5†	10x10	100	265	63	6	100	565	563	6	100,04
abz6†	10x10	100	273	205	7	100	554	7	4	100,01
abz7†	20x15	300	84	895	2	100	175	90	6	100,52
abz8†	20x15	300	66	38	7	101	165	94	9	100,3
abz9†	20x15	300	74	37	5	102	167	134	9	100,09
car1‡	11x5	55	640	25	9	100	1295	915	11	100,05
car2‡	13x4	52	734	41	3	100	1368	1028	9	100
car3‡	12x5	60	576	511	10	100	1239	480	7	100,07

car4‡	14x4	56	622	194	8	100	1406	586	9	100,08
car5‡	10x6	60	495	443	6	100	1091	14	3	100,01
car6‡	8x9	72	401	4	2	100	802	339	9	100,06
car7‡	7x7	63	732	633	5	100	1347	831	7	100,06
car8‡	8x8	64	427	181	8	100	980	164	6	100,08
orb1††	10x10	100	247	109	4	100	554	150	6	100,11
orb2††	10x10	100	247	245	4	100	422	332	5	100,02
orb3††	10x10	100	178	172	6	100	496	87	4	100,06
orb4††	10x10	100	207	109	5	100	544	269	8	100,06
orb5††	10x10	100	250	338	7	101	507	86	5	100,07
orb6††	10x10	100	220	6	3	100	524	345	6	100,03
orb7††	10x10	100	274	126	4	101	511	365	6	100,12
orb8††	10x10	100	237	120	8	100	514	73	5	100,04
orb9††	10x10	100	263	76	7	100	526	62	5	100,1
orb10††	10x10	100	223	54	3	100,23	522	197	10	100,08

Fonte: Adaptado de Rigo (2018).

APÊNDICE D - TABELA DE RESULTADOS DA CONFIGURAÇÃO 1 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS EDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado Aleatório	Configuração 1					
					Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	55	57	55	0,00%	-3,51%	1422	600,2	0,42
mt10†	10x10	100	871	1180	960	10,22%	-18,64%	421	600,92	1,43
mt20†	20x5	100	1088	1622	1293	18,84%	-20,28%	537	600,22	1,12
la01‡	10x5	50	609	753	621	1,97%	-17,53%	1041	600,03	0,58
la02‡	10x5	50	655	823	655	0,00%	-20,41%	1071	600,09	0,56
la03‡	10x5	50	550	699	581	5,64%	-16,88%	1089	600,38	0,55
la04‡	10x5	50	568	724	591	4,05%	-18,37%	1088	600,16	0,55
la05‡	10x5	50	503	653	569	13,12%	-12,86%	1067	600,52	0,56
la06‡	15x5	75	833	1041	833	0,00%	-19,98%	743	600,21	0,81
la07‡	15x5	75	762	1031	821	7,74%	-20,37%	767	600,3	0,78
la08‡	15x5	75	845	1042	944	11,72%	-9,40%	758	601,09	0,79
la09‡	15x5	75	878	1088	900	2,51%	-17,28%	749	600,18	0,80
la10‡	15x5	75	866	1029	937	8,20%	-8,94%	682	600,54	0,88
la11‡	20x5	100	1103	1360	1191	7,98%	-12,43%	555	600,07	1,08
la12‡	20x5	100	960	1192	1004	4,58%	-15,77%	529	600,37	1,13
la13‡	20x5	100	1053	1259	1123	6,65%	-10,80%	556	600,85	1,08
la14‡	20x5	100	1123	1349	1199	6,77%	-11,12%	539	600,03	1,11
la15‡	20x5	100	1111	1447	1234	11,07%	-14,72%	565	600,34	1,06
la16‡	10x10	100	892	1224	1080	21,08%	-11,76%	408	600,46	1,47
la17‡	10x10	100	707	970	878	24,19%	-9,48%	403	600,29	1,49
la18‡	10x10	100	842	1119	948	12,59%	-15,28%	449	600,52	1,34

la19‡	10x10	100	796	1135	911	14,45%	-19,74%	437	600,97	1,38
la20‡	10x10	100	857	1204	1024	19,49%	-14,95%	416	601,23	1,45
la21‡	15x10	150	1017	1520	1155	13,57%	-24,01%	295	601,77	2,04
la22‡	15x10	150	882	1401	1075	21,88%	-23,27%	284	600,38	2,11
la23‡	15x10	150	950	1465	1123	18,21%	-23,34%	289	601,93	2,08
la24‡	15x10	150	909	1423	1105	21,56%	-22,35%	292	602,11	2,06
la25‡	15x10	150	941	1516	1158	23,06%	-23,61%	298	601,47	2,02
la26‡	20x10	200	1125	1778	1336	18,76%	-24,86%	202	601,54	2,98
la27‡	20x10	200	1186	1845	1578	33,05%	-14,47%	227	600,49	2,65
la28‡	20x10	200	1149	1784	1407	22,45%	-21,13%	223	601,39	2,70
la29‡	20x10	200	1118	1809	1379	23,35%	-23,77%	217	600,08	2,77
la30‡	20x10	200	1204	1966	1557	29,32%	-20,80%	220	602,3	2,74
la31‡	30x10	300	1539	2415	1922	24,89%	-20,41%	146	603,7	4,13
la32‡	30x10	300	1698	2600	2103	23,85%	-19,12%	142	600,76	4,23
la33‡	30x10	300	1547	2482	1915	23,79%	-22,84%	137	600,87	4,39
la34‡	30x10	300	1604	2525	1834	14,34%	-27,37%	138	603,85	4,38
la35‡	30x10	300	1736	2633	2137	23,10%	-18,84%	146	602,86	4,13
la36‡	15x15	225	1162	1905	1461	25,73%	-23,31%	151	600,35	3,98
la37‡	15x15	225	1397	2103	1704	21,98%	-18,97%	150	601,64	4,01
la38‡	15x15	225	1144	1858	1471	28,58%	-20,83%	153	600,65	3,93
la39‡	15x15	225	1184	1865	1586	33,95%	-14,96%	137	601,3	4,39
la40‡	15x15	225	1150	1893	1474	28,17%	-22,13%	136	600,96	4,42
abz5†	10x10	100	1176	1613	1254	6,63%	-22,26%	412	600,89	1,46
abz6†	10x10	100	925	1226	1142	23,46%	-6,85%	424	600,7	1,42
abz7†	20x15	300	638	1085	851	33,39%	-21,57%	105	601,7	5,73
abz8†	20x15	300	654	1113	856	30,89%	-23,09%	112	600,23	5,36
abz9†	20x15	300	668	1037	865	29,49%	-16,59%	118	600,32	5,09
car1‡	11x5	55	6176	10231	7151	15,79%	-30,10%	966	600,1	0,62
car2‡	13x4	52	6455	9296	7561	17,13%	-18,66%	1092	600,48	0,55
car3‡	12x5	60	6856	10913	7806	13,86%	-28,47%	898	600,26	0,67

car4‡	14x4	56	7789	10381	8319	6,80%	-19,86%	1052	600,34	0,57
car5‡	10x6	60	7229	10565	7863	8,77%	-25,58%	841	600,71	0,71
car6‡	8x9	72	8478	12156	8445	-0,39%	-30,53%	630	600,7	0,95
car7‡	7x7	63	6123	8042	6950	13,51%	-13,58%	983	600,16	0,61
car8‡	8x8	64	7689	11154	8708	13,25%	-21,93%	717	600,57	0,84
orb1††	10x10	100	988	1421	1162	17,61%	-18,23%	410	600,09	1,46
orb2††	10x10	100	870	1222	1059	21,72%	-13,34%	420	601,63	1,43
orb3††	10x10	100	960	1495	1154	20,21%	-22,81%	403	600,09	1,49
orb4††	10x10	100	1016	1357	1148	12,99%	-15,40%	433	600,18	1,39
orb5††	10x10	100	865	1298	925	6,94%	-28,74%	390	600,66	1,54
orb6††	10x10	100	1004	1395	1154	14,94%	-17,28%	433	601,08	1,39
orb7††	10x10	100	387	528	449	16,02%	-14,96%	422	601,24	1,42
orb8††	10x10	100	894	1297	1056	18,12%	-18,58%	407	601,93	1,48
orb9††	10x10	100	933	1254	1028	10,18%	-18,02%	433	600,77	1,39
orb10††	10x10	100	937	1371	1119	19,42%	-18,38%	404	601,23	1,49

Fonte: Adaptado de Rigo (2018).

APÊNDICE E - TABELA DE RESULTADOS DA CONFIGURAÇÃO 2 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS EDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado Aleatório	Configuração 2					
					Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	55	57	55	0,00%	-3,51%	1564	600,04	0,38
mt10†	10x10	100	871	1180	1024	17,57%	-13,22%	465	600,04	1,29
mt20†	20x5	100	1088	1622	1247	14,61%	-23,12%	617	600,9	0,97
la01‡	10x5	50	609	753	609	0,00%	-19,12%	1186	600,48	0,51
la02‡	10x5	50	655	823	682	4,12%	-17,13%	1182	600,4	0,51
la03‡	10x5	50	550	699	606	10,18%	-13,30%	1207	600,31	0,50
la04‡	10x5	50	568	724	602	5,99%	-16,85%	1229	600,02	0,49
la05‡	10x5	50	503	653	513	1,99%	-21,44%	1232	600,33	0,49
la06‡	15x5	75	833	1041	855	2,64%	-17,87%	776	600,29	0,77
la07‡	15x5	75	762	1031	826	8,40%	-19,88%	842	600,45	0,71
la08‡	15x5	75	845	1042	871	3,08%	-16,41%	851	600,63	0,71
la09‡	15x5	75	878	1088	912	3,87%	-16,18%	814	600,2	0,74
la10‡	15x5	75	866	1029	886	2,31%	-13,90%	807	600,2	0,74
la11‡	20x5	100	1103	1360	1143	3,63%	-15,96%	604	600,27	0,99
la12‡	20x5	100	960	1192	982	2,29%	-17,62%	609	600,48	0,99
la13‡	20x5	100	1053	1259	1074	1,99%	-14,69%	607	600,62	0,99
la14‡	20x5	100	1123	1349	1163	3,56%	-13,79%	622	600,36	0,97
la15‡	20x5	100	1111	1447	1210	8,91%	-16,38%	648	600,32	0,93
la16‡	10x10	100	892	1224	968	8,52%	-20,92%	475	600,98	1,27
la17‡	10x10	100	707	970	795	12,45%	-18,04%	495	600,84	1,21
la18‡	10x10	100	842	1119	916	8,79%	-18,14%	470	600,66	1,28

la19†	10x10	100	796	1135	895	12,44%	-21,15%	448	600,27	1,34
la20†	10x10	100	857	1204	893	4,20%	-25,83%	494	601,29	1,22
la21†	15x10	150	1017	1520	1228	20,75%	-19,21%	323	600,61	1,86
la22†	15x10	150	882	1401	1045	18,48%	-25,41%	316	601,03	1,90
la23†	15x10	150	950	1465	1113	17,16%	-24,03%	289	601,33	2,08
la24†	15x10	150	909	1423	1106	21,67%	-22,28%	315	601,82	1,91
la25†	15x10	150	941	1516	1132	20,30%	-25,33%	324	601,54	1,86
la26†	20x10	200	1125	1778	1378	22,49%	-22,50%	227	600,4	2,64
la27†	20x10	200	1186	1845	1485	25,21%	-19,51%	226	601,94	2,66
la28†	20x10	200	1149	1784	1475	28,37%	-17,32%	235	601	2,56
la29†	20x10	200	1118	1809	1428	27,73%	-21,06%	240	602,17	2,51
la30†	20x10	200	1204	1966	1476	22,59%	-24,92%	235	600,41	2,55
la31†	30x10	300	1539	2415	1995	29,63%	-17,39%	151	601,88	3,99
la32†	30x10	300	1698	2600	2050	20,73%	-21,15%	155	603,36	3,89
la33†	30x10	300	1547	2482	1916	23,85%	-22,80%	147	602,76	4,10
la34†	30x10	300	1604	2525	1969	22,76%	-22,02%	150	601,62	4,01
la35†	30x10	300	1736	2633	2070	19,24%	-21,38%	156	601,83	3,86
la36†	15x15	225	1162	1905	1430	23,06%	-24,93%	176	601,53	3,42
la37†	15x15	225	1397	2103	1650	18,11%	-21,54%	161	600,88	3,73
la38†	15x15	225	1144	1858	1430	25,00%	-23,04%	162	602,29	3,72
la39†	15x15	225	1184	1865	1551	31,00%	-16,84%	170	602,19	3,54
la40†	15x15	225	1150	1893	1443	25,48%	-23,77%	168	602,34	3,59
abz5†	10x10	100	1176	1613	1285	9,27%	-20,33%	471	600,76	1,28
abz6†	10x10	100	925	1226	1018	10,05%	-16,97%	445	601,78	1,35
abz7†	20x15	300	638	1085	852	33,54%	-21,47%	123	603,9	4,91
abz8†	20x15	300	654	1113	870	33,03%	-21,83%	126	602,94	4,79
abz9†	20x15	300	668	1037	894	33,83%	-13,79%	126	602,58	4,78
car1†	11x5	55	6176	10231	6878	11,37%	-32,77%	1118	600,46	0,54
car2†	13x4	52	6455	9296	7187	11,34%	-22,69%	1265	600,47	0,47
car3†	12x5	60	6856	10913	7774	13,39%	-28,76%	1001	600,41	0,60

car4‡	14x4	56	7789	10381	8209	5,39%	-20,92%	1137	600,38	0,53
car5‡	10x6	60	7229	10565	7665	6,03%	-27,45%	946	600,3	0,63
car6‡	8x9	72	8478	12156	8834	4,20%	-27,33%	668	600,21	0,90
car7‡	7x7	63	6123	8042	6577	7,41%	-18,22%	1086	600,66	0,55
car8‡	8x8	64	7689	11154	8032	4,46%	-27,99%	809	600,56	0,74
orb1††	10x10	100	988	1421	1118	13,16%	-21,32%	477	601,11	1,26
orb2††	10x10	100	870	1222	978	12,41%	-19,97%	461	601,08	1,30
orb3††	10x10	100	960	1495	1063	10,73%	-28,90%	485	601,16	1,24
orb4††	10x10	100	1016	1357	1124	10,63%	-17,17%	459	600,04	1,31
orb5††	10x10	100	865	1298	971	12,25%	-25,19%	477	601,11	1,26
orb6††	10x10	100	1004	1395	1104	9,96%	-20,86%	432	601,5	1,39
orb7††	10x10	100	387	528	440	13,70%	-16,67%	475	600,43	1,26
orb8††	10x10	100	894	1297	1023	14,43%	-21,13%	471	601,59	1,28
orb9††	10x10	100	933	1254	1028	10,18%	-18,02%	462	600,37	1,30
orb10††	10x10	100	937	1371	1096	16,97%	-20,06%	441	600,7	1,36

Fonte: Adaptado de Rigo (2018).

APÊNDICE F - TABELA DE RESULTADOS DA CONFIGURAÇÃO 3 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS EDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado Aleatório	Configuração 3					
					Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	55	57	55	0,00%	-3,51%	1474	600,33	0,41
mt10†	10x10	100	871	1180	975	11,94%	-17,37%	450	600,99	1,34
mt20†	20x5	100	1088	1622	1283	17,92%	-20,90%	579	600,89	1,04
la01‡	10x5	50	609	753	631	3,61%	-16,20%	1140	600,17	0,53
la02‡	10x5	50	655	823	676	3,21%	-17,86%	1081	600,35	0,56
la03‡	10x5	50	550	699	614	11,64%	-12,16%	1161	600,07	0,52
la04‡	10x5	50	568	724	607	6,87%	-16,16%	1161	600,34	0,52
la05‡	10x5	50	503	653	524	4,17%	-19,75%	1131	600,06	0,53
la06‡	15x5	75	833	1041	833	0,00%	-19,98%	761	600,16	0,79
la07‡	15x5	75	762	1031	831	9,06%	-19,40%	811	600,13	0,74
la08‡	15x5	75	845	1042	882	4,38%	-15,36%	802	600,47	0,75
la09‡	15x5	75	878	1088	928	5,69%	-14,71%	773	600,6	0,78
la10‡	15x5	75	866	1029	937	8,20%	-8,94%	761	600,27	0,79
la11‡	20x5	100	1103	1360	1129	2,36%	-16,99%	548	600,24	1,10
la12‡	20x5	100	960	1192	982	2,29%	-17,62%	601	600,39	1,00
la13‡	20x5	100	1053	1259	1092	3,70%	-13,26%	578	600,74	1,04
la14‡	20x5	100	1123	1349	1195	6,41%	-11,42%	595	600,68	1,01
la15‡	20x5	100	1111	1447	1235	11,16%	-14,65%	581	600,6	1,03
la16‡	10x10	100	892	1224	1057	18,50%	-13,64%	468	601,15	1,28
la17‡	10x10	100	707	970	839	18,67%	-13,51%	431	601,02	1,39
la18‡	10x10	100	842	1119	861	2,26%	-23,06%	428	600,79	1,40

la19‡	10x10	100	796	1135	845	6,16%	-25,55%	417	600,91	1,44
la20‡	10x10	100	857	1204	909	6,07%	-24,50%	444	600,51	1,35
la21‡	15x10	150	1017	1520	1166	14,65%	-23,29%	291	601,17	2,07
la22‡	15x10	150	882	1401	1050	19,05%	-25,05%	298	601,42	2,02
la23‡	15x10	150	950	1465	1115	17,37%	-23,89%	295	602	2,04
la24‡	15x10	150	909	1423	1054	15,95%	-25,93%	296	601,27	2,03
la25‡	15x10	150	941	1516	1279	35,92%	-15,63%	308	601,2	1,95
la26‡	20x10	200	1125	1778	1300	15,56%	-26,88%	228	601,42	2,64
la27‡	20x10	200	1186	1845	1403	18,30%	-23,96%	230	600,97	2,61
la28‡	20x10	200	1149	1784	1342	16,80%	-24,78%	222	601,04	2,71
la29‡	20x10	200	1118	1809	1517	35,69%	-16,14%	230	600,51	2,61
la30‡	20x10	200	1204	1966	1501	24,67%	-23,65%	235	602,34	2,56
la31‡	30x10	300	1539	2415	1903	23,65%	-21,20%	152	602,75	3,97
la32‡	30x10	300	1698	2600	1981	16,67%	-23,81%	145	601,23	4,15
la33‡	30x10	300	1547	2482	1814	17,26%	-26,91%	160	600,14	3,75
la34‡	30x10	300	1604	2525	1869	16,52%	-25,98%	148	602,51	4,07
la35‡	30x10	300	1736	2633	2031	16,99%	-22,86%	159	605,28	3,81
la36‡	15x15	225	1162	1905	1434	23,41%	-24,72%	169	600,69	3,55
la37‡	15x15	225	1397	2103	1650	18,11%	-21,54%	163	603,31	3,70
la38‡	15x15	225	1144	1858	1465	28,06%	-21,15%	169	602,56	3,57
la39‡	15x15	225	1184	1865	1602	35,30%	-14,10%	163	600,58	3,68
la40‡	15x15	225	1150	1893	1486	29,22%	-21,50%	157	601,02	3,83
abz5†	10x10	100	1176	1613	1287	9,44%	-20,21%	471	600,52	1,27
abz6†	10x10	100	925	1226	1086	17,41%	-11,42%	448	601,05	1,34
abz7†	20x15	300	638	1085	831	30,25%	-23,41%	110	600,47	5,46
abz8†	20x15	300	654	1113	782	19,57%	-29,74%	122	601,12	4,93
abz9†	20x15	300	668	1037	831	24,40%	-19,86%	127	606,29	4,77
car1‡	11x5	55	6176	10231	7728	25,13%	-24,46%	995	600,09	0,60
car2‡	13x4	52	6455	9296	7520	16,50%	-19,10%	1197	600,55	0,50
car3‡	12x5	60	6856	10913	8486	23,77%	-22,24%	956	600,31	0,63

car4‡	14x4	56	7789	10381	8535	9,58%	-17,78%	1069	600,39	0,56
car5‡	10x6	60	7229	10565	8779	21,44%	-16,90%	942	600,41	0,64
car6‡	8x9	72	8478	12156	9309	9,80%	-23,42%	681	600,75	0,88
car7‡	7x7	63	6123	8042	7039	14,96%	-12,47%	1077	600,37	0,56
car8‡	8x8	64	7689	11154	8737	13,63%	-21,67%	808	600,29	0,74
orb1††	10x10	100	988	1421	1150	16,40%	-19,07%	470	601,19	1,28
orb2††	10x10	100	870	1222	1002	15,17%	-18,00%	471	600,52	1,27
orb3††	10x10	100	960	1495	1091	13,65%	-27,02%	467	601,07	1,29
orb4††	10x10	100	1016	1357	1117	9,94%	-17,69%	465	600,02	1,29
orb5††	10x10	100	865	1298	959	10,87%	-26,12%	482	601,06	1,25
orb6††	10x10	100	1004	1395	1107	10,26%	-20,65%	460	600,48	1,31
orb7††	10x10	100	387	528	437	12,92%	-17,23%	429	601,08	1,40
orb8††	10x10	100	894	1297	1104	23,49%	-14,88%	441	600,77	1,36
orb9††	10x10	100	933	1254	1022	9,54%	-18,50%	434	600,22	1,38
orb10††	10x10	100	937	1371	1108	18,25%	-19,18%	471	600,98	1,28

Fonte: Adaptado de Rigo (2018).

APÊNDICE G - TABELA DE RESULTADOS DA CONFIGURAÇÃO 4 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS EDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado Aleatório	Configuração 4					
					Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	55	57	59	7,27%	3,51%	1385	600,27	0,43
mt10†	10x10	100	871	1180	1025	17,68%	-13,14%	423	600,36	1,42
mt20†	20x5	100	1088	1622	1283	17,92%	-20,90%	532	600,99	1,13
la01‡	10x5	50	609	753	640	5,09%	-15,01%	1094	600,06	0,55
la02‡	10x5	50	655	823	781	19,24%	-5,10%	1052	600,24	0,57
la03‡	10x5	50	550	699	606	10,18%	-13,30%	1075	600,64	0,56
la04‡	10x5	50	568	724	635	11,80%	-12,29%	1065	600,48	0,56
la05‡	10x5	50	503	653	522	3,78%	-20,06%	1048	600,32	0,57
la06‡	15x5	75	833	1041	875	5,04%	-15,95%	689	600,47	0,87
la07‡	15x5	75	762	1031	817	7,22%	-20,76%	728	600,02	0,82
la08‡	15x5	75	845	1042	912	7,93%	-12,48%	735	600,54	0,82
la09‡	15x5	75	878	1088	918	4,56%	-15,63%	727	600,59	0,83
la10‡	15x5	75	866	1029	866	0,00%	-15,84%	688	600,26	0,87
la11‡	20x5	100	1103	1360	1127	2,18%	-17,13%	512	600,75	1,17
la12‡	20x5	100	960	1192	1032	7,50%	-13,42%	556	601,35	1,08
la13‡	20x5	100	1053	1259	1161	10,26%	-7,78%	478	601,41	1,26
la14‡	20x5	100	1123	1349	1163	3,56%	-13,79%	499	600,26	1,20
la15‡	20x5	100	1111	1447	1256	13,05%	-13,20%	520	600,14	1,15
la16‡	10x10	100	892	1224	1031	15,58%	-15,77%	338	600,19	1,78
la17‡	10x10	100	707	970	845	19,52%	-12,89%	389	601,31	1,55
la18‡	10x10	100	842	1119	978	16,15%	-12,60%	434	601,16	1,39

la19†	10x10	100	796	1135	938	17,84%	-17,36%	425	601	1,41
la20†	10x10	100	857	1204	1056	23,22%	-12,29%	390	600,02	1,54
la21†	15x10	150	1017	1520	1248	22,71%	-17,89%	260	601,49	2,31
la22†	15x10	150	882	1401	1203	36,39%	-14,13%	270	600,14	2,22
la23†	15x10	150	950	1465	1168	22,95%	-20,27%	278	600,5	2,16
la24†	15x10	150	909	1423	1103	21,34%	-22,49%	282	602,24	2,14
la25†	15x10	150	941	1516	1097	16,58%	-27,64%	279	601,41	2,16
la26†	20x10	200	1125	1778	1342	19,29%	-24,52%	208	600,92	2,89
la27†	20x10	200	1186	1845	1577	32,97%	-14,53%	195	601,12	3,08
la28†	20x10	200	1149	1784	1353	17,75%	-24,16%	176	603,31	3,43
la29†	20x10	200	1118	1809	1404	25,58%	-22,39%	202	603,56	2,99
la30†	20x10	200	1204	1966	1473	22,34%	-25,08%	216	602,21	2,79
la31†	30x10	300	1539	2415	1871	21,57%	-22,53%	142	602,2	4,24
la32†	30x10	300	1698	2600	2101	23,73%	-19,19%	134	602,45	4,50
la33†	30x10	300	1547	2482	1882	21,65%	-24,17%	126	603,27	4,79
la34†	30x10	300	1604	2525	1930	20,32%	-23,56%	145	603,79	4,16
la35†	30x10	300	1736	2633	2040	17,51%	-22,52%	137	601,39	4,39
la36†	15x15	225	1162	1905	1392	19,79%	-26,93%	128	600,65	4,69
la37†	15x15	225	1397	2103	1758	25,84%	-16,41%	146	604,05	4,14
la38†	15x15	225	1144	1858	1466	28,15%	-21,10%	157	601,22	3,83
la39†	15x15	225	1184	1865	1462	23,48%	-21,61%	161	603,07	3,75
la40†	15x15	225	1150	1893	1452	26,26%	-23,30%	156	601,48	3,86
abz5†	10x10	100	1176	1613	1343	14,20%	-16,74%	417	601,26	1,44
abz6†	10x10	100	925	1226	1026	10,92%	-16,31%	401	600,29	1,50
abz7†	20x15	300	638	1085	845	32,45%	-22,12%	108	606,32	5,61
abz8†	20x15	300	654	1113	801	22,48%	-28,03%	113	600,15	5,31
abz9†	20x15	300	668	1037	837	25,30%	-19,29%	111	601,34	5,42
car1†	11x5	55	6176	10231	6906	11,82%	-32,50%	949	600,33	0,63
car2†	13x4	52	6455	9296	7656	18,61%	-17,64%	1092	600,36	0,55
car3†	12x5	60	6856	10913	9003	31,32%	-17,50%	912	600,72	0,66

car4‡	14x4	56	7789	10381	8224	5,58%	-20,78%	1017	600,23	0,59
car5‡	10x6	60	7229	10565	8977	24,18%	-15,03%	806	600,49	0,75
car6‡	8x9	72	8478	12156	10709	26,32%	-11,90%	608	600,3	0,99
car7‡	7x7	63	6123	8042	7250	18,41%	-9,85%	959	600,44	0,63
car8‡	8x8	64	7689	11154	10341	34,49%	-7,29%	662	600,88	0,91
orb1††	10x10	100	988	1421	1192	20,65%	-16,12%	410	600,18	1,46
orb2††	10x10	100	870	1222	1053	21,03%	-13,83%	376	600,89	1,60
orb3††	10x10	100	960	1495	1205	25,52%	-19,40%	426	600,87	1,41
orb4††	10x10	100	1016	1357	1162	14,37%	-14,37%	424	600,51	1,42
orb5††	10x10	100	865	1298	1062	22,77%	-18,18%	433	601,2	1,39
orb6††	10x10	100	1004	1395	1182	17,73%	-15,27%	394	601,54	1,53
orb7††	10x10	100	387	528	455	17,57%	-13,83%	435	600,49	1,38
orb8††	10x10	100	894	1297	949	6,15%	-26,83%	417	600,59	1,44
orb9††	10x10	100	933	1254	1036	11,04%	-17,38%	404	600,38	1,49
orb10††	10x10	100	937	1371	1159	23,69%	-15,46%	433	600,36	1,39

Fonte: Adaptado de Rigo (2018).

APÊNDICE H - TABELA DE RESULTADOS DA CONFIGURAÇÃO 1 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS *RDATA* DO AUTOR HURINK *ET. AL* (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado <i>Hill-Climbing</i>	Melhor Resultado Aleatório	Configuração 1						
						Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap <i>Hill-Climbing</i>	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	50	59	54	14,89%	8,00%	-8,47%	1349	600,2	0,44
mt10†	10x10	100	686	861	1143	894	30,32%	3,83%	-21,78%	392	600,99	1,53
mt20†	20x5	100	1022	1164	1397	1062	3,91%	-8,76%	-23,98%	495	600,35	1,21
la01‡	10x5	50	571	640	761	595	4,20%	-7,03%	-21,81%	1022	600,37	0,59
la02‡	10x5	50	530	605	744	552	4,15%	-8,76%	-25,81%	1046	600,19	0,57
la03‡	10x5	50	478	547	638	492	2,93%	-10,05%	-22,88%	1045	600,16	0,57
la04‡	10x5	50	502	582	694	551	9,76%	-5,33%	-20,61%	1055	600,44	0,57
la05‡	10x5	50	457	500	623	485	6,13%	-3,00%	-22,15%	1015	600,24	0,59
la06‡	15x5	75	799	878	997	812	1,63%	-7,52%	-18,56%	694	600,09	0,86
la07‡	15x5	75	750	807	983	784	4,53%	-2,85%	-20,24%	640	600,11	0,94
la08‡	15x5	75	765	852	1007	777	1,57%	-8,80%	-22,84%	663	600,79	0,91
la09‡	15x5	75	853	938	1097	859	0,70%	-8,42%	-21,70%	675	600,3	0,89
la10‡	15x5	75	804	884	1077	826	2,74%	-6,56%	-23,31%	647	600,76	0,93
la11‡	20x5	100	1071	1168	1353	1095	2,24%	-6,25%	-19,07%	525	600,71	1,14
la12‡	20x5	100	936	1016	1207	945	0,96%	-6,99%	-21,71%	513	600,7	1,17
la13‡	20x5	100	1038	1103	1328	1042	0,39%	-5,53%	-21,54%	506	600,95	1,19
la14‡	20x5	100	1070	1145	1356	1076	0,56%	-6,03%	-20,65%	533	600,02	1,13
la15‡	20x5	100	1090	1187	1380	1095	0,46%	-7,75%	-20,65%	503	600,6	1,19
la16‡	10x10	100	717	929	1147	847	18,13%	-8,83%	-26,16%	401	600,37	1,50
la17‡	10x10	100	646	821	968	788	21,98%	-4,02%	-18,60%	414	600,42	1,45
la18‡	10x10	100	666	878	1069	913	37,09%	3,99%	-14,59%	399	600,75	1,51

la19‡	10x10	100	700	976	1112	873	24,71%	-10,55%	-21,49%	413	601,27	1,46
la20‡	10x10	100	756	944	1182	874	15,61%	-7,42%	-26,06%	396	600,27	1,52
la21‡	15x10	150	835	1188	1509	1095	31,14%	-7,83%	-27,44%	275	601,47	2,19
la22‡	15x10	150	760	1141	1413	1118	47,11%	-2,02%	-20,88%	276	600,98	2,18
la23‡	15x10	150	842	1195	1561	1061	26,01%	-11,21%	-32,03%	279	600,94	2,15
la24‡	15x10	150	808	1182	1448	1064	31,68%	-9,98%	-26,52%	268	600,75	2,24
la25‡	15x10	150	791	1132	1351	996	25,92%	-12,01%	-26,28%	268	600,29	2,24
la26‡	20x10	200	1061	1435	1840	1321	24,51%	-7,94%	-28,21%	186	600,22	3,23
la27‡	20x10	200	1091	1534	1789	1388	27,22%	-9,52%	-22,41%	214	600,09	2,80
la28‡	20x10	200	1080	1511	1780	1413	30,83%	-6,49%	-20,62%	210	600,03	2,86
la29‡	20x10	200	998	1385	1823	1247	24,95%	-9,96%	-31,60%	195	601,61	3,09
la30‡	20x10	200	1078	1517	1917	1398	29,68%	-7,84%	-27,07%	203	602,14	2,97
la31‡	30x10	300	1521	1976	2501	1754	15,32%	-11,23%	-29,87%	130	612,46	4,71
la32‡	30x10	300	1659	2084	2665	2003	20,74%	-3,89%	-24,84%	143	602,09	4,21
la33‡	30x10	300	1499	1920	2484	1761	17,48%	-8,28%	-29,11%	134	601,65	4,49
la34‡	30x10	300	1536	1927	2413	1841	19,86%	-4,46%	-23,70%	129	600,57	4,66
la35‡	30x10	300	1550	1940	2332	1854	19,61%	-4,43%	-20,50%	136	604,06	4,44
la36‡	15x15	225	1030	1532	1897	1479	43,59%	-3,46%	-22,03%	152	601,55	3,96
la37‡	15x15	225	1077	1652	2006	1524	41,50%	-7,75%	-24,03%	142	601,91	4,24
la38‡	15x15	225	962	1459	1879	1343	39,60%	-7,95%	-28,53%	155	603,02	3,89
la39‡	15x15	225	1024	1512	1880	1355	32,32%	-10,38%	-27,93%	148	600,53	4,06
la40‡	15x15	225	970	1482	1785	1474	51,96%	-0,54%	-17,42%	136	602,15	4,43
abz5†	10x10	100	962	1264	1620	1282	33,26%	1,42%	-20,86%	389	600,15	1,54
abz6†	10x10	100	807	1018	1210	924	14,50%	-9,23%	-23,64%	413	600,94	1,46
abz7†	20x15	300	544	790	1039	772	41,91%	-2,28%	-25,70%	109	605,8	5,56
abz8†	20x15	300	555	854	1045	837	50,81%	-1,99%	-19,90%	114	602	5,28
abz9†	20x15	300	562	855	1059	843	50,00%	-1,40%	-20,40%	112	602,53	5,38
car1‡	11x5	55	5057	5801	7281	5556	9,87%	-4,22%	-23,69%	933	600,3	0,64
car2‡	13x4	52	5987	6433	7095	6160	2,89%	-4,24%	-13,18%	1037	600,12	0,58
car3‡	12x5	60	5626	6599	8045	5941	5,60%	-9,97%	-26,15%	865	600,08	0,69

car4‡	14x4	56	6518	7071	8378	6853	5,14%	-3,08%	-18,20%	999	600,25	0,60
car5‡	10x6	60	5764	7163	8960	6698	16,20%	-6,49%	-25,25%	852	600,63	0,70
car6‡	8x9	72	6147	7837	9407	7965	29,58%	1,63%	-15,33%	601	600,31	1,00
car7‡	7x7	63	4432	5330	6807	4899	10,54%	-8,09%	-28,03%	918	600,54	0,65
car8‡	8x8	64	5692	6961	8878	6883	20,92%	-1,12%	-22,47%	713	600,73	0,84
orb1††	10x10	100	763	999	1280	1030	34,99%	3,10%	-19,53%	393	601,12	1,53
orb2††	10x10	100	703	920	1156	840	19,49%	-8,70%	-27,34%	380	600,43	1,58
orb3††	10x10	100	720	958	1207	892	23,89%	-6,89%	-26,10%	408	600,37	1,47
orb4††	10x10	100	753	970	1200	1017	35,06%	4,85%	-15,25%	378	600,26	1,59
orb5††	10x10	100	643	899	1114	831	29,24%	-7,56%	-25,40%	415	600,52	1,45
orb6††	10x10	100	766	1021	1210	947	23,63%	-7,25%	-21,74%	413	600,95	1,46
orb7††	10x10	100	302	405	503	389	28,81%	-3,95%	-22,66%	413	601,27	1,46
orb8††	10x10	100	651	847	1076	836	28,42%	-1,30%	-22,30%	424	600,84	1,42
orb9††	10x10	100	694	950	1138	890	28,24%	-6,32%	-21,79%	403	601,07	1,49
orb10††	10x10	100	750	981	1223	1007	34,27%	2,65%	-17,66%	423	600,33	1,42

Fonte: Adaptado de Rigo (2018).

APÊNDICE I - TABELA DE RESULTADOS DA CONFIGURAÇÃO 2 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS RDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado <i>Hill-Climbing</i>	Melhor Resultado Aleatório	Configuração 2						
						Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap <i>Hill-Climbing</i>	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	50	59	49	4,26%	-2,00%	-16,95%	1505	600,1	0,40
mt10†	10x10	100	686	861	1143	878	27,99%	1,97%	-23,18%	443	600,32	1,36
mt20†	20x5	100	1022	1164	1397	1074	5,09%	-7,73%	-23,12%	552	600,74	1,09
la01‡	10x5	50	571	640	761	594	4,03%	-7,19%	-21,94%	1077	600,34	0,56
la02‡	10x5	50	530	605	744	543	2,45%	-10,25%	-27,02%	1091	600,72	0,55
la03‡	10x5	50	478	547	638	503	5,23%	-8,04%	-21,16%	1082	600,44	0,55
la04‡	10x5	50	502	582	694	529	5,38%	-9,11%	-23,78%	1089	600,47	0,55
la05‡	10x5	50	457	500	623	464	1,53%	-7,20%	-25,52%	1027	600,6	0,58
la06‡	15x5	75	799	878	997	817	2,25%	-6,95%	-18,05%	703	601,01	0,85
la07‡	15x5	75	750	807	983	774	3,20%	-4,09%	-21,26%	667	600,85	0,90
la08‡	15x5	75	765	852	1007	780	1,96%	-8,45%	-22,54%	723	600,59	0,83
la09‡	15x5	75	853	938	1097	872	2,23%	-7,04%	-20,51%	691	600,28	0,87
la10‡	15x5	75	804	884	1077	813	1,12%	-8,03%	-24,51%	702	600,21	0,86
la11‡	20x5	100	1071	1168	1353	1080	0,84%	-7,53%	-20,18%	501	600,42	1,20
la12‡	20x5	100	936	1016	1207	952	1,71%	-6,30%	-21,13%	466	600,8	1,29
la13‡	20x5	100	1038	1103	1328	1045	0,67%	-5,26%	-21,31%	521	600,3	1,15
la14‡	20x5	100	1070	1145	1356	1088	1,68%	-4,98%	-19,76%	527	601,02	1,14
la15‡	20x5	100	1090	1187	1380	1117	2,48%	-5,90%	-19,06%	545	600,41	1,10
la16‡	10x10	100	717	929	1147	862	20,22%	-7,21%	-24,85%	439	601,27	1,37
la17‡	10x10	100	646	821	968	737	14,09%	-10,23%	-23,86%	427	601,61	1,41
la18‡	10x10	100	666	878	1069	874	31,23%	-0,46%	-18,24%	456	601,16	1,32

la19†	10x10	100	700	976	1112	874	24,86%	-10,45%	-21,40%	468	600,79	1,28
la20†	10x10	100	756	944	1182	907	19,97%	-3,92%	-23,27%	462	600,08	1,30
la21†	15x10	150	835	1188	1509	1164	39,40%	-2,02%	-22,86%	276	601,09	2,18
la22†	15x10	150	760	1141	1413	1004	32,11%	-12,01%	-28,95%	270	601,06	2,23
la23†	15x10	150	842	1195	1561	1138	35,15%	-4,77%	-27,10%	299	600,18	2,01
la24†	15x10	150	808	1182	1448	1144	41,58%	-3,21%	-20,99%	249	602,05	2,42
la25†	15x10	150	791	1132	1351	1065	34,64%	-5,92%	-21,17%	285	600,63	2,11
la26†	20x10	200	1061	1435	1840	1359	28,09%	-5,30%	-26,14%	214	601,83	2,81
la27†	20x10	200	1091	1534	1789	1403	28,60%	-8,54%	-21,58%	203	600,91	2,96
la28†	20x10	200	1080	1511	1780	1430	32,41%	-5,36%	-19,66%	195	601,59	3,09
la29†	20x10	200	998	1385	1823	1290	29,26%	-6,86%	-29,24%	219	602,83	2,75
la30†	20x10	200	1078	1517	1917	1404	30,24%	-7,45%	-26,76%	203	600,22	2,96
la31†	30x10	300	1521	1976	2501	1850	21,63%	-6,38%	-26,03%	139	601,14	4,32
la32†	30x10	300	1659	2084	2665	2064	24,41%	-0,96%	-22,55%	140	603,16	4,31
la33†	30x10	300	1499	1920	2484	1912	27,55%	-0,42%	-23,03%	151	600,74	3,98
la34†	30x10	300	1536	1927	2413	1894	23,31%	-1,71%	-21,51%	133	600,47	4,51
la35†	30x10	300	1550	1940	2332	1871	20,71%	-3,56%	-19,77%	141	603,03	4,28
la36†	15x15	225	1030	1532	1897	1511	46,70%	-1,37%	-20,35%	160	601,47	3,76
la37†	15x15	225	1077	1652	2006	1556	44,48%	-5,81%	-22,43%	147	600,87	4,09
la38†	15x15	225	962	1459	1879	1368	42,20%	-6,24%	-27,20%	162	601,83	3,72
la39†	15x15	225	1024	1512	1880	1442	40,82%	-4,63%	-23,30%	161	600,98	3,73
la40†	15x15	225	970	1482	1785	1376	41,86%	-7,15%	-22,91%	145	601,8	4,15
abz5†	10x10	100	962	1264	1620	1181	22,77%	-6,57%	-27,10%	444	601,77	1,36
abz6†	10x10	100	807	1018	1210	924	14,50%	-9,23%	-23,64%	460	600,09	1,30
abz7†	20x15	300	544	790	1039	827	52,02%	4,68%	-20,40%	116	600,6	5,18
abz8†	20x15	300	555	854	1045	793	42,88%	-7,14%	-24,11%	121	601,98	4,98
abz9†	20x15	300	562	855	1059	831	47,86%	-2,81%	-21,53%	117	603,35	5,16
car1†	11x5	55	5057	5801	7281	5169	2,21%	-10,89%	-29,01%	987	600,14	0,61
car2†	13x4	52	5987	6433	7095	6583	9,95%	2,33%	-7,22%	1156	600,2	0,52
car3†	12x5	60	5626	6599	8045	5841	3,82%	-11,49%	-27,40%	904	600,51	0,66

car4‡	14x4	56	6518	7071	8378	6571	0,81%	-7,07%	-21,57%	973	600,06	0,62
car5‡	10x6	60	5764	7163	8960	6607	14,63%	-7,76%	-26,26%	881	600,34	0,68
car6‡	8x9	72	6147	7837	9407	6909	12,40%	-11,84%	-26,55%	663	600,52	0,91
car7‡	7x7	63	4432	5330	6807	4880	10,11%	-8,44%	-28,31%	1007	600,36	0,60
car8‡	8x8	64	5692	6961	8878	6655	16,92%	-4,40%	-25,04%	772	600,31	0,78
orb1††	10x10	100	763	999	1280	992	30,01%	-0,70%	-22,50%	452	600,41	1,33
orb2††	10x10	100	703	920	1156	861	22,48%	-6,41%	-25,52%	441	600,71	1,36
orb3††	10x10	100	720	958	1207	893	24,03%	-6,78%	-26,01%	421	600,48	1,43
orb4††	10x10	100	753	970	1200	944	25,37%	-2,68%	-21,33%	455	600,6	1,32
orb5††	10x10	100	643	899	1114	872	35,61%	-3,00%	-21,72%	456	600,71	1,32
orb6††	10x10	100	766	1021	1210	1007	31,46%	-1,37%	-16,78%	446	601,12	1,35
orb7††	10x10	100	302	405	503	378	25,17%	-6,67%	-24,85%	433	600,66	1,39
orb8††	10x10	100	651	847	1076	789	21,20%	-6,85%	-26,67%	440	600,53	1,36
orb9††	10x10	100	694	950	1138	862	24,21%	-9,26%	-24,25%	419	600,9	1,43
orb10††	10x10	100	750	981	1223	893	19,07%	-8,97%	-26,98%	457	600,83	1,31

Fonte: Adaptado de Rigo (2018).

APÊNDICE J - TABELA DE RESULTADOS DA CONFIGURAÇÃO 3 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS *RDATA* DO AUTOR HURINK *ET. AL* (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado <i>Hill-Climbing</i>	Melhor Resultado Aleatório	Configuração 3						
						Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap <i>Hill-Climbing</i>	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	50	59	53	12,77%	6,00%	-10,17%	1427	600,36	0,42
mt10†	10x10	100	686	861	1143	846	23,32%	-1,74%	-25,98%	442	601,59	1,36
mt20†	20x5	100	1022	1164	1397	1090	6,65%	-6,36%	-21,98%	561	601,23	1,07
la01‡	10x5	50	571	640	761	583	2,10%	-8,91%	-23,39%	1138	600,51	0,53
la02‡	10x5	50	530	605	744	545	2,83%	-9,92%	-26,75%	1134	600,09	0,53
la03‡	10x5	50	478	547	638	557	16,53%	1,83%	-12,70%	1119	600,3	0,54
la04‡	10x5	50	502	582	694	533	6,18%	-8,42%	-23,20%	1098	600,51	0,55
la05‡	10x5	50	457	500	623	484	5,91%	-3,20%	-22,31%	1134	600,25	0,53
la06‡	15x5	75	799	878	997	806	0,88%	-8,20%	-19,16%	764	600,12	0,79
la07‡	15x5	75	750	807	983	770	2,67%	-4,58%	-21,67%	747	600,39	0,80
la08‡	15x5	75	765	852	1007	787	2,88%	-7,63%	-21,85%	750	600,14	0,80
la09‡	15x5	75	853	938	1097	857	0,47%	-8,64%	-21,88%	758	600,26	0,79
la10‡	15x5	75	804	884	1077	811	0,87%	-8,26%	-24,70%	776	600,12	0,77
la11‡	20x5	100	1071	1168	1353	1095	2,24%	-6,25%	-19,07%	552	600,94	1,09
la12‡	20x5	100	936	1016	1207	943	0,75%	-7,19%	-21,87%	528	600,33	1,14
la13‡	20x5	100	1038	1103	1328	1051	1,25%	-4,71%	-20,86%	557	601,01	1,08
la14‡	20x5	100	1070	1145	1356	1075	0,47%	-6,11%	-20,72%	563	600,77	1,07
la15‡	20x5	100	1090	1187	1380	1126	3,30%	-5,14%	-18,41%	551	600,35	1,09
la16‡	10x10	100	717	929	1147	906	26,36%	-2,48%	-21,01%	463	600,66	1,30
la17‡	10x10	100	646	821	968	714	10,53%	-13,03%	-26,24%	437	601,33	1,38
la18‡	10x10	100	666	878	1069	865	29,88%	-1,48%	-19,08%	384	600,34	1,56

la19‡	10x10	100	700	976	1112	859	22,71%	-11,99%	-22,75%	438	600,27	1,37
la20‡	10x10	100	756	944	1182	877	16,01%	-7,10%	-25,80%	456	600,37	1,32
la21‡	15x10	150	835	1188	1509	1162	39,16%	-2,19%	-23,00%	314	600,35	1,91
la22‡	15x10	150	760	1141	1413	1045	37,50%	-8,41%	-26,04%	265	600,81	2,27
la23‡	15x10	150	842	1195	1561	1042	23,75%	-12,80%	-33,25%	303	601,57	1,99
la24‡	15x10	150	808	1182	1448	1139	40,97%	-3,64%	-21,34%	301	601,55	2,00
la25‡	15x10	150	791	1132	1351	1098	38,81%	-3,00%	-18,73%	292	600,86	2,06
la26‡	20x10	200	1061	1435	1840	1320	24,41%	-8,01%	-28,26%	223	600,03	2,69
la27‡	20x10	200	1091	1534	1789	1336	22,46%	-12,91%	-25,32%	212	601,01	2,83
la28‡	20x10	200	1080	1511	1780	1398	29,44%	-7,48%	-21,46%	207	600,31	2,90
la29‡	20x10	200	998	1385	1823	1279	28,16%	-7,65%	-29,84%	215	602,01	2,80
la30‡	20x10	200	1078	1517	1917	1456	35,06%	-4,02%	-24,05%	197	600,68	3,05
la31‡	30x10	300	1521	1976	2501	1843	21,17%	-6,73%	-26,31%	145	602,72	4,16
la32‡	30x10	300	1659	2084	2665	1951	17,60%	-6,38%	-26,79%	141	601,9	4,27
la33‡	30x10	300	1499	1920	2484	1895	26,42%	-1,30%	-23,71%	147	603,3	4,10
la34‡	30x10	300	1536	1927	2413	1788	16,41%	-7,21%	-25,90%	150	602,3	4,02
la35‡	30x10	300	1550	1940	2332	1823	17,61%	-6,03%	-21,83%	143	601,49	4,21
la36‡	15x15	225	1030	1532	1897	1365	32,52%	-10,90%	-28,04%	156	604,03	3,87
la37‡	15x15	225	1077	1652	2006	1519	41,04%	-8,05%	-24,28%	151	602,89	3,99
la38‡	15x15	225	962	1459	1879	1419	47,51%	-2,74%	-24,48%	166	602,89	3,63
la39‡	15x15	225	1024	1512	1880	1396	36,33%	-7,67%	-25,74%	172	600,24	3,49
la40‡	15x15	225	970	1482	1785	1310	35,05%	-11,61%	-26,61%	172	601,76	3,50
abz5†	10x10	100	962	1264	1620	1195	24,22%	-5,46%	-26,23%	443	600,9	1,36
abz6†	10x10	100	807	1018	1210	973	20,57%	-4,42%	-19,59%	424	601,11	1,42
abz7†	20x15	300	544	790	1039	729	34,01%	-7,72%	-29,84%	128	600,71	4,69
abz8†	20x15	300	555	854	1045	818	47,39%	-4,22%	-21,72%	120	601,86	5,02
abz9†	20x15	300	562	855	1059	831	47,86%	-2,81%	-21,53%	119	601,93	5,06
car1‡	11x5	55	5057	5801	7281	5185	2,53%	-10,62%	-28,79%	1052	600,38	0,57
car2‡	13x4	52	5987	6433	7095	6206	3,66%	-3,53%	-12,53%	1145	600,52	0,52
car3‡	12x5	60	5626	6599	8045	5711	1,51%	-13,46%	-29,01%	918	600,38	0,65

car4‡	14x4	56	6518	7071	8378	6587	1,06%	-6,84%	-21,38%	1057	600,1	0,57
car5‡	10x6	60	5764	7163	8960	6491	12,61%	-9,38%	-27,56%	934	600,63	0,64
car6‡	8x9	72	6147	7837	9407	7370	19,90%	-5,96%	-21,65%	643	600,46	0,93
car7‡	7x7	63	4432	5330	6807	5137	15,91%	-3,62%	-24,53%	987	600,31	0,61
car8‡	8x8	64	5692	6961	8878	6623	16,36%	-4,86%	-25,40%	755	600,65	0,80
orb1††	10x10	100	763	999	1280	951	24,64%	-4,80%	-25,70%	470	600,81	1,28
orb2††	10x10	100	703	920	1156	852	21,19%	-7,39%	-26,30%	423	600,11	1,42
orb3††	10x10	100	720	958	1207	843	17,08%	-12,00%	-30,16%	443	600,09	1,35
orb4††	10x10	100	753	970	1200	932	23,77%	-3,92%	-22,33%	456	600,88	1,32
orb5††	10x10	100	643	899	1114	780	21,31%	-13,24%	-29,98%	462	600,77	1,30
orb6††	10x10	100	766	1021	1210	940	22,72%	-7,93%	-22,31%	441	600,6	1,36
orb7††	10x10	100	302	405	503	389	28,81%	-3,95%	-22,66%	467	601,06	1,29
orb8††	10x10	100	651	847	1076	891	36,87%	5,19%	-17,19%	442	600,66	1,36
orb9††	10x10	100	694	950	1138	925	33,29%	-2,63%	-18,72%	465	600,28	1,29
orb10††	10x10	100	750	981	1223	860	14,67%	-12,33%	-29,68%	463	600,44	1,30

Fonte: Adaptado de Rigo (2018).

APÊNDICE K - TABELA DE RESULTADOS DA CONFIGURAÇÃO 4 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS *RDATA* DO AUTOR HURINK *ET. AL* (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado <i>Hill-Climbing</i>	Melhor Resultado Aleatório	Configuração 4						
						Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap <i>Hill-Climbing</i>	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	50	59	53	12,77%	6,00%	-10,17%	1272	600,42	0,47
mt10†	10x10	100	686	861	1143	846	23,32%	-1,74%	-25,98%	364	601,14	1,65
mt20†	20x5	100	1022	1164	1397	1031	0,88%	-11,43%	-26,20%	496	600,69	1,21
la01‡	10x5	50	571	640	761	593	3,85%	-7,34%	-22,08%	950	600,56	0,63
la02‡	10x5	50	530	605	744	546	3,02%	-9,75%	-26,61%	949	600,44	0,63
la03‡	10x5	50	478	547	638	488	2,09%	-10,79%	-23,51%	897	600,62	0,67
la04‡	10x5	50	502	582	694	596	18,73%	2,41%	-14,12%	996	600,35	0,60
la05‡	10x5	50	457	500	623	473	3,50%	-5,40%	-24,08%	927	600,74	0,65
la06‡	15x5	75	799	878	997	805	0,75%	-8,31%	-19,26%	626	600,17	0,96
la07‡	15x5	75	750	807	983	756	0,80%	-6,32%	-23,09%	632	600,34	0,95
la08‡	15x5	75	765	852	1007	772	0,92%	-9,39%	-23,34%	618	600,28	0,97
la09‡	15x5	75	853	938	1097	854	0,12%	-8,96%	-22,15%	598	600,61	1,00
la10‡	15x5	75	804	884	1077	812	1,00%	-8,14%	-24,61%	602	600,63	1,00
la11‡	20x5	100	1071	1168	1353	1073	0,19%	-8,13%	-20,69%	460	600,14	1,30
la12‡	20x5	100	936	1016	1207	938	0,21%	-7,68%	-22,29%	464	600,64	1,29
la13‡	20x5	100	1038	1103	1328	1043	0,48%	-5,44%	-21,46%	456	600,59	1,32
la14‡	20x5	100	1070	1145	1356	1073	0,28%	-6,29%	-20,87%	456	600,45	1,32
la15‡	20x5	100	1090	1187	1380	1104	1,28%	-6,99%	-20,00%	478	600,34	1,26
la16‡	10x10	100	717	929	1147	967	34,87%	4,09%	-15,69%	369	600,04	1,63
la17‡	10x10	100	646	821	968	864	33,75%	5,24%	-10,74%	395	601	1,52
la18‡	10x10	100	666	878	1069	811	21,77%	-7,63%	-24,13%	368	601,02	1,63

la19‡	10x10	100	700	976	1112	880	25,71%	-9,84%	-20,86%	359	601,18	1,67
la20‡	10x10	100	756	944	1182	865	14,42%	-8,37%	-26,82%	351	600,57	1,71
la21‡	15x10	150	835	1188	1509	1073	28,50%	-9,68%	-28,89%	237	602,11	2,54
la22‡	15x10	150	760	1141	1413	1002	31,84%	-12,18%	-29,09%	250	600,11	2,40
la23‡	15x10	150	842	1195	1561	1124	33,49%	-5,94%	-27,99%	261	601,42	2,30
la24‡	15x10	150	808	1182	1448	1024	26,73%	-13,37%	-29,28%	241	601,84	2,50
la25‡	15x10	150	791	1132	1351	1132	43,11%	0,00%	-16,21%	255	600,06	2,35
la26‡	20x10	200	1061	1435	1840	1390	31,01%	-3,14%	-24,46%	192	600,38	3,13
la27‡	20x10	200	1091	1534	1789	1367	25,30%	-10,89%	-23,59%	188	600,54	3,19
la28‡	20x10	200	1080	1511	1780	1410	30,56%	-6,68%	-20,79%	179	600,1	3,35
la29‡	20x10	200	998	1385	1823	1393	39,58%	0,58%	-23,59%	174	602,96	3,47
la30‡	20x10	200	1078	1517	1917	1397	29,59%	-7,91%	-27,13%	172	601,37	3,50
la31‡	30x10	300	1521	1976	2501	1928	26,76%	-2,43%	-22,91%	137	601,71	4,39
la32‡	30x10	300	1659	2084	2665	2038	22,85%	-2,21%	-23,53%	136	600,09	4,41
la33‡	30x10	300	1499	1920	2484	1748	16,61%	-8,96%	-29,63%	130	603,02	4,64
la34‡	30x10	300	1536	1927	2413	1889	22,98%	-1,97%	-21,72%	126	601,11	4,77
la35‡	30x10	300	1550	1940	2332	1777	14,65%	-8,40%	-23,80%	138	601,33	4,36
la36‡	15x15	225	1030	1532	1897	1438	39,61%	-6,14%	-24,20%	144	601,56	4,18
la37‡	15x15	225	1077	1652	2006	1545	43,45%	-6,48%	-22,98%	149	602,66	4,04
la38‡	15x15	225	962	1459	1879	1444	50,10%	-1,03%	-23,15%	134	600,37	4,48
la39‡	15x15	225	1024	1512	1880	1467	43,26%	-2,98%	-21,97%	144	601,99	4,18
la40‡	15x15	225	970	1482	1785	1341	38,25%	-9,51%	-24,87%	154	601,14	3,90
abz5†	10x10	100	962	1264	1620	1259	30,87%	-0,40%	-22,28%	365	600,08	1,64
abz6†	10x10	100	807	1018	1210	947	17,35%	-6,97%	-21,74%	414	600,69	1,45
abz7†	20x15	300	544	790	1039	761	39,89%	-3,67%	-26,76%	112	604,46	5,40
abz8†	20x15	300	555	854	1045	850	53,15%	-0,47%	-18,66%	100	601,87	6,02
abz9†	20x15	300	562	855	1059	824	46,62%	-3,63%	-22,19%	99	601,48	6,08
car1‡	11x5	55	5057	5801	7281	5223	3,28%	-9,96%	-28,27%	913	600,47	0,66
car2‡	13x4	52	5987	6433	7095	6082	1,59%	-5,46%	-14,28%	1080	600,13	0,56
car3‡	12x5	60	5626	6599	8045	5699	1,30%	-13,64%	-29,16%	851	600,23	0,71

car4‡	14x4	56	6518	7071	8378	6573	0,84%	-7,04%	-21,54%	927	600,64	0,65
car5‡	10x6	60	5764	7163	8960	7013	21,67%	-2,09%	-21,73%	823	600,55	0,73
car6‡	8x9	72	6147	7837	9407	7656	24,55%	-2,31%	-18,61%	585	600,27	1,03
car7‡	7x7	63	4432	5330	6807	5301	19,61%	-0,54%	-22,12%	960	600,73	0,63
car8‡	8x8	64	5692	6961	8878	6671	17,20%	-4,17%	-24,86%	668	600,75	0,90
orb1††	10x10	100	763	999	1280	1053	38,01%	5,41%	-17,73%	389	600,62	1,54
orb2††	10x10	100	703	920	1156	913	29,87%	-0,76%	-21,02%	359	601,09	1,67
orb3††	10x10	100	720	958	1207	1102	53,06%	15,03%	-8,70%	428	600,96	1,40
orb4††	10x10	100	753	970	1200	932	23,77%	-3,92%	-22,33%	414	601,08	1,45
orb5††	10x10	100	643	899	1114	866	34,68%	-3,67%	-22,26%	424	600,29	1,42
orb6††	10x10	100	766	1021	1210	1025	33,81%	0,39%	-15,29%	388	600,61	1,55
orb7††	10x10	100	302	405	503	376	24,50%	-7,16%	-25,25%	386	600,3	1,56
orb8††	10x10	100	651	847	1076	897	37,79%	5,90%	-16,64%	407	600,92	1,48
orb9††	10x10	100	694	950	1138	899	29,54%	-5,37%	-21,00%	389	600,28	1,54
orb10††	10x10	100	750	981	1223	1031	37,47%	5,10%	-15,70%	378	600,06	1,59

Fonte: Adaptado de Rigo (2018).

APÊNDICE L - TABELA DE RESULTADOS DA CONFIGURAÇÃO 1 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS VDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado Aleatório	Configuração 1					
					Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	56	50	6,38%	-10,71%	1349	600,29	0,44
mt10†	10x10	100	655	1098	753	14,96%	-31,42%	378	601,24	1,59
mt20†	20x5	100	1022	1321	1035	1,27%	-21,65%	507	600,39	1,18
la01‡	10x5	50	570	758	591	3,68%	-22,03%	964	600,41	0,62
la02‡	10x5	50	529	715	555	4,91%	-22,38%	1028	600,2	0,58
la03‡	10x5	50	477	638	555	16,35%	-13,01%	1026	600,45	0,59
la04‡	10x5	50	502	667	518	3,19%	-22,34%	997	600,31	0,60
la05‡	10x5	50	457	629	492	7,66%	-21,78%	1073	600,23	0,56
la06‡	15x5	75	799	1002	801	0,25%	-20,06%	719	600,18	0,83
la07‡	15x5	75	749	1029	765	2,14%	-25,66%	656	600,2	0,91
la08‡	15x5	75	765	1008	782	2,22%	-22,42%	672	600,36	0,89
la09‡	15x5	75	853	1079	875	2,58%	-18,91%	712	601,02	0,84
la10‡	15x5	75	804	1038	813	1,12%	-21,68%	662	600,41	0,91
la11‡	20x5	100	1071	1356	1079	0,75%	-20,43%	491	601,73	1,23
la12‡	20x5	100	936	1156	960	2,56%	-16,96%	538	600,41	1,12
la13‡	20x5	100	1038	1327	1056	1,73%	-20,42%	541	600,46	1,11
la14‡	20x5	100	1070	1394	1074	0,37%	-22,96%	514	600,02	1,17
la15‡	20x5	100	1089	1324	1094	0,46%	-17,37%	513	601,62	1,17
la16‡	10x10	100	717	1119	794	10,74%	-29,04%	408	600,83	1,47
la17‡	10x10	100	646	965	752	16,41%	-22,07%	380	601,27	1,58
la18‡	10x10	100	663	1078	795	19,91%	-26,25%	381	601,05	1,58
la19‡	10x10	100	617	1087	796	29,01%	-26,77%	417	601,32	1,44

la20‡	10x10	100	756	1128	1002	32,54%	-11,17%	377	601,04	1,59
la21‡	15x10	150	806	1454	1009	25,19%	-30,61%	237	601,58	2,54
la22‡	15x10	150	739	1360	1016	37,48%	-25,29%	262	600,84	2,29
la23‡	15x10	150	815	1501	1126	38,16%	-24,98%	274	601,27	2,19
la24‡	15x10	150	777	1395	1045	34,49%	-25,09%	271	600,55	2,22
la25‡	15x10	150	756	1450	999	32,14%	-31,10%	227	600,83	2,65
la26‡	20x10	200	1054	1790	1303	23,62%	-27,21%	199	600,97	3,02
la27‡	20x10	200	1085	1908	1320	21,66%	-30,82%	179	602,08	3,36
la28‡	20x10	200	1070	1904	1245	16,36%	-34,61%	168	602,51	3,59
la29‡	20x10	200	994	1764	1326	33,40%	-24,83%	192	603,31	3,14
la30‡	20x10	200	1069	1917	1297	21,33%	-32,34%	185	602,35	3,26
la31‡	30x10	300	1520	2449	1804	18,68%	-26,34%	114	601,36	5,28
la32‡	30x10	300	1658	2644	1920	15,80%	-27,38%	132	603,7	4,57
la33‡	30x10	300	1497	2440	1748	16,77%	-28,36%	126	601,41	4,77
la34‡	30x10	300	1535	2462	1799	17,20%	-26,93%	131	603,5	4,61
la35‡	30x10	300	1549	2535	1782	15,04%	-29,70%	118	603,63	5,12
la36‡	15x15	225	948	1855	1285	35,55%	-30,73%	135	603,69	4,47
la37‡	15x15	225	986	2022	1476	49,70%	-27,00%	124	601,98	4,85
la38‡	15x15	225	943	1851	1267	34,36%	-31,55%	146	603,41	4,13
la39‡	15x15	225	922	1822	1306	41,65%	-28,32%	135	602,13	4,46
la40‡	15x15	225	955	1870	1356	41,99%	-27,49%	136	602,38	4,43
abz5†	10x10	100	860	1610	1151	33,84%	-28,51%	385	600,67	1,56
abz6†	10x10	100	742	1252	928	25,07%	-25,88%	406	600,7	1,48
abz7†	20x15	300	495	1035	731	47,68%	-29,37%	103	602,63	5,85
abz8†	20x15	300	509	1105	749	47,15%	-32,22%	96	602,47	6,28
abz9†	20x15	300	500	1064	765	53,00%	-28,10%	100	600,81	6,01
car1‡	11x5	55	5013	6601	5551	10,73%	-15,91%	931	600,48	0,64
car2‡	13x4	52	5930	7308	5963	0,56%	-18,40%	1011	600,12	0,59
car3‡	12x5	60	5600	7692	5658	1,04%	-26,44%	839	600,27	0,72
car4‡	14x4	56	5617	8032	6533	16,31%	-18,66%	952	600,2	0,63

car5‡	10x6	60	4932	7873	5505	11,62%	-30,08%	796	600,17	0,75
car6‡	8x9	72	5486	8649	5983	9,06%	-30,82%	565	600,39	1,06
car7‡	7x7	63	4281	5936	5540	29,41%	-6,67%	882	600,44	0,68
car8‡	8x8	64	4613	7700	6749	46,30%	-12,35%	682	600,19	0,88
orb1††	10x10	100	695	1130	789	13,53%	-30,18%	386	600,77	1,56
orb2††	10x10	100	620	1117	817	31,77%	-26,86%	381	601,05	1,58
orb3††	10x10	100	648	1127	824	27,16%	-26,89%	411	600,99	1,46
orb4††	10x10	100	753	1191	876	16,33%	-26,45%	370	600,23	1,62
orb5††	10x10	100	584	1023	779	33,39%	-23,85%	376	600,34	1,60
orb6††	10x10	100	715	1093	932	30,35%	-14,73%	350	600,98	1,72
orb7††	10x10	100	275	472	416	51,27%	-11,86%	398	601,7	1,51
orb8††	10x10	100	573	982	674	17,63%	-31,36%	352	600,98	1,71
orb9††	10x10	100	659	1066	883	33,99%	-17,17%	373	600,13	1,61
orb10††	10x10	100	681	1184	919	34,95%	-22,38%	372	600,02	1,61

Fonte: Adaptado de Rigo (2018).

APÊNDICE M - TABELA DE RESULTADOS DA CONFIGURAÇÃO 2 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS VDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor Makespan Conhecido	Melhor Resultado Aleatório	Configuração 2					
					Resultado Obtido	Gap Melhor Makespan Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	56	47	0,00%	-16,07%	1434	600,12	0,42
mt10†	10x10	100	655	1098	858	30,99%	-21,86%	432	600,77	1,39
mt20†	20x5	100	1022	1321	1039	1,66%	-21,35%	458	601,09	1,31
la01‡	10x5	50	570	758	598	4,91%	-21,11%	1026	600,52	0,59
la02‡	10x5	50	529	715	537	1,51%	-24,90%	978	600,27	0,61
la03‡	10x5	50	477	638	502	5,24%	-21,32%	1018	600,2	0,59
la04‡	10x5	50	502	667	539	7,37%	-19,19%	1052	600,01	0,57
la05‡	10x5	50	457	629	472	3,28%	-24,96%	1036	600,28	0,58
la06‡	15x5	75	799	1002	818	2,38%	-18,36%	684	600,93	0,88
la07‡	15x5	75	749	1029	768	2,54%	-25,36%	673	600,94	0,89
la08‡	15x5	75	765	1008	784	2,48%	-22,22%	648	600,73	0,93
la09‡	15x5	75	853	1079	859	0,70%	-20,39%	657	601,25	0,92
la10‡	15x5	75	804	1038	809	0,62%	-22,06%	647	600,59	0,93
la11‡	20x5	100	1071	1356	1098	2,52%	-19,03%	490	600,32	1,23
la12‡	20x5	100	936	1156	953	1,82%	-17,56%	444	600,44	1,35
la13‡	20x5	100	1038	1327	1049	1,06%	-20,95%	476	602,66	1,27
la14‡	20x5	100	1070	1394	1077	0,65%	-22,74%	481	600,94	1,25
la15‡	20x5	100	1089	1324	1100	1,01%	-16,92%	450	601,3	1,34
la16‡	10x10	100	717	1119	834	16,32%	-25,47%	408	600,49	1,47
la17‡	10x10	100	646	965	719	11,30%	-25,49%	406	601,02	1,48
la18‡	10x10	100	663	1078	819	23,53%	-24,03%	406	601,22	1,48
la19‡	10x10	100	617	1087	804	30,31%	-26,03%	399	600,49	1,50

la20‡	10x10	100	756	1128	884	16,93%	-21,63%	418	600,35	1,44
la21‡	15x10	150	806	1454	1073	33,13%	-26,20%	273	600,88	2,20
la22‡	15x10	150	739	1360	1042	41,00%	-23,38%	290	600,5	2,07
la23‡	15x10	150	815	1501	1069	31,17%	-28,78%	265	600,59	2,27
la24‡	15x10	150	777	1395	1088	40,03%	-22,01%	269	601,8	2,24
la25‡	15x10	150	756	1450	1025	35,58%	-29,31%	257	600,28	2,34
la26‡	20x10	200	1054	1790	1454	37,95%	-18,77%	201	601,25	2,99
la27‡	20x10	200	1085	1908	1336	23,13%	-29,98%	190	603,61	3,18
la28‡	20x10	200	1070	1904	1403	31,12%	-26,31%	192	600,28	3,13
la29‡	20x10	200	994	1764	1276	28,37%	-27,66%	190	603,78	3,18
la30‡	20x10	200	1069	1917	1363	27,50%	-28,90%	191	601,39	3,15
la31‡	30x10	300	1520	2449	1827	20,20%	-25,40%	122	604,53	4,96
la32‡	30x10	300	1658	2644	2060	24,25%	-22,09%	124	603,32	4,87
la33‡	30x10	300	1497	2440	1767	18,04%	-27,58%	123	600,22	4,88
la34‡	30x10	300	1535	2462	1848	20,39%	-24,94%	125	602,83	4,82
la35‡	30x10	300	1549	2535	1854	19,69%	-26,86%	121	604,29	4,99
la36‡	15x15	225	948	1855	1358	43,25%	-26,79%	143	600,89	4,20
la37‡	15x15	225	986	2022	1409	42,90%	-30,32%	142	600,52	4,23
la38‡	15x15	225	943	1851	1339	41,99%	-27,66%	149	601,6	4,04
la39‡	15x15	225	922	1822	1363	47,83%	-25,19%	156	601,61	3,86
la40‡	15x15	225	955	1870	1406	47,23%	-24,81%	147	601,4	4,09
abz5†	10x10	100	860	1610	1194	38,84%	-25,84%	443	601,4	1,36
abz6†	10x10	100	742	1252	887	19,54%	-29,15%	419	600,67	1,43
abz7†	20x15	300	495	1035	756	52,73%	-26,96%	114	605,48	5,31
abz8†	20x15	300	509	1105	767	50,69%	-30,59%	117	601,88	5,14
abz9†	20x15	300	500	1064	744	48,80%	-30,08%	105	603,42	5,75
car1‡	11x5	55	5013	6601	5141	2,55%	-22,12%	908	600,04	0,66
car2‡	13x4	52	5930	7308	5950	0,34%	-18,58%	1000	600,16	0,60
car3‡	12x5	60	5600	7692	5857	4,59%	-23,86%	838	600,25	0,72
car4‡	14x4	56	5617	8032	6575	17,06%	-18,14%	1011	600,01	0,59

car5†	10x6	60	4932	7873	5672	15,00%	-27,96%	804	600,26	0,75
car6†	8x9	72	5486	8649	6235	13,65%	-27,91%	625	600,25	0,96
car7†	7x7	63	4281	5936	4745	10,84%	-20,06%	988	600,23	0,61
car8†	8x8	64	4613	7700	5635	22,15%	-26,82%	709	600,52	0,85
orb1††	10x10	100	695	1130	862	24,03%	-23,72%	435	601,06	1,38
orb2††	10x10	100	620	1117	846	36,45%	-24,26%	425	601,23	1,41
orb3††	10x10	100	648	1127	870	34,26%	-22,80%	418	600,34	1,44
orb4††	10x10	100	753	1191	839	11,42%	-29,55%	405	600,72	1,48
orb5††	10x10	100	584	1023	769	31,68%	-24,83%	416	601,46	1,45
orb6††	10x10	100	715	1093	886	23,92%	-18,94%	422	600,16	1,42
orb7††	10x10	100	275	472	381	38,55%	-19,28%	424	600,16	1,42
orb8††	10x10	100	573	982	710	23,91%	-27,70%	426	601,2	1,41
orb9††	10x10	100	659	1066	852	29,29%	-20,08%	421	601,32	1,43
orb10††	10x10	100	681	1184	892	30,98%	-24,66%	400	600,74	1,50

Fonte: Adaptado de Rigo (2018).

APÊNDICE N - TABELA DE RESULTADOS DA CONFIGURAÇÃO 3 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS VDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor Makespan Conhecido	Melhor Resultado Aleatório	Configuração 3					
					Resultado Obtido	Gap Melhor Makespan Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	56	48	2,13%	-14,29%	1426	600,2	0,42
mt10†	10x10	100	655	1098	789	20,46%	-28,14%	432	600,125	1,39
mt20†	20x5	100	1022	1321	1054	3,13%	-20,21%	542	601,09	1,11
la01‡	10x5	50	570	758	601	5,44%	-20,71%	1140	600,45	0,53
la02‡	10x5	50	529	715	564	6,62%	-21,12%	1108	600,58	0,54
la03‡	10x5	50	477	638	504	5,66%	-21,00%	1100	600,05	0,55
la04‡	10x5	50	502	667	527	4,98%	-20,99%	1130	600,84	0,53
la05‡	10x5	50	457	629	498	8,97%	-20,83%	1093	600,34	0,55
la06‡	15x5	75	799	1002	816	2,13%	-18,56%	734	600,38	0,82
la07‡	15x5	75	749	1029	757	1,07%	-26,43%	682	601,3	0,88
la08‡	15x5	75	765	1008	772	0,92%	-23,41%	696	600,15	0,86
la09‡	15x5	75	853	1079	877	2,81%	-18,72%	751	600,05	0,80
la10‡	15x5	75	804	1038	807	0,37%	-22,25%	719	600,47	0,84
la11‡	20x5	100	1071	1356	1078	0,65%	-20,50%	538	600,14	1,12
la12‡	20x5	100	936	1156	948	1,28%	-17,99%	526	600,8	1,14
la13‡	20x5	100	1038	1327	1067	2,79%	-19,59%	576	601,125	1,04
la14‡	20x5	100	1070	1394	1095	2,34%	-21,45%	566	601,21	1,06
la15‡	20x5	100	1089	1324	1106	1,56%	-16,47%	563	600,91	1,07
la16‡	10x10	100	717	1119	887	23,71%	-20,73%	435	600,89	1,38
la17‡	10x10	100	646	965	774	19,81%	-19,79%	438	601,05	1,37
la18‡	10x10	100	663	1078	839	26,55%	-22,17%	425	600,79	1,41
la19‡	10x10	100	617	1087	895	45,06%	-17,66%	446	600,53	1,35

la20‡	10x10	100	756	1128	965	27,65%	-14,45%	394	601,28	1,53
la21‡	15x10	150	806	1454	1142	41,69%	-21,46%	268	601,69	2,25
la22‡	15x10	150	739	1360	977	32,21%	-28,16%	258	600,93	2,33
la23‡	15x10	150	815	1501	1097	34,60%	-26,92%	299	600,17	2,01
la24‡	15x10	150	777	1395	1070	37,71%	-23,30%	283	600,36	2,12
la25‡	15x10	150	756	1450	984	30,16%	-32,14%	267	601,4	2,25
la26‡	20x10	200	1054	1790	1274	20,87%	-28,83%	199	601,43	3,02
la27‡	20x10	200	1085	1908	1328	22,40%	-30,40%	205	600,86	2,93
la28‡	20x10	200	1070	1904	1265	18,22%	-33,56%	184	601,3	3,27
la29‡	20x10	200	994	1764	1227	23,44%	-30,44%	214	600,41	2,81
la30‡	20x10	200	1069	1917	1361	27,32%	-29,00%	216	602,09	2,79
la31‡	30x10	300	1520	2449	1791	17,83%	-26,87%	130	603,4	4,64
la32‡	30x10	300	1658	2644	1943	17,19%	-26,51%	139	602,66	4,34
la33‡	30x10	300	1497	2440	1660	10,89%	-31,97%	132	602,48	4,56
la34‡	30x10	300	1535	2462	1786	16,35%	-27,46%	138	600,96	4,35
la35‡	30x10	300	1549	2535	1768	14,14%	-30,26%	132	602,84	4,57
la36‡	15x15	225	948	1855	1395	47,15%	-24,80%	156	602,84	3,86
la37‡	15x15	225	986	2022	1353	37,22%	-33,09%	146	602,69	4,13
la38‡	15x15	225	943	1851	1230	30,43%	-33,55%	139	601,58	4,33
la39‡	15x15	225	922	1822	1285	39,37%	-29,47%	147	601,87	4,09
la40‡	15x15	225	955	1870	1299	36,02%	-30,53%	148	602,22	4,07
abz5†	10x10	100	860	1610	1181	37,33%	-26,65%	387	601,33	1,55
abz6†	10x10	100	742	1252	885	19,27%	-29,31%	415	601,66	1,45
abz7†	20x15	300	495	1035	710	43,43%	-31,40%	104	605,36	5,82
abz8†	20x15	300	509	1105	777	52,65%	-29,68%	112	600,97	5,37
abz9†	20x15	300	500	1064	727	45,40%	-31,67%	112	604,46	5,40
car1‡	11x5	55	5013	6601	5301	5,75%	-19,69%	1014	600,46	0,59
car2‡	13x4	52	5930	7308	5976	0,78%	-18,23%	1121	600,16	0,54
car3‡	12x5	60	5600	7692	6195	10,63%	-19,46%	956	600,49	0,63
car4‡	14x4	56	5617	8032	6633	18,09%	-17,42%	1074	600,14	0,56

car5‡	10x6	60	4932	7873	5619	13,93%	-28,63%	835	600,55	0,72
car6‡	8x9	72	5486	8649	6927	26,27%	-19,91%	616	600,12	0,97
car7‡	7x7	63	4281	5936	5075	18,55%	-14,50%	945	600,01	0,63
car8‡	8x8	64	4613	7700	5871	27,27%	-23,75%	655	600,23	0,92
orb1††	10x10	100	695	1130	856	23,17%	-24,25%	389	601,02	1,55
orb2††	10x10	100	620	1117	762	22,90%	-31,78%	386	601,22	1,56
orb3††	10x10	100	648	1127	892	37,65%	-20,85%	420	600,23	1,43
orb4††	10x10	100	753	1191	887	17,80%	-25,52%	428	601,27	1,40
orb5††	10x10	100	584	1023	730	25,00%	-28,64%	420	600,89	1,43
orb6††	10x10	100	715	1093	905	26,57%	-17,20%	433	600,24	1,39
orb7††	10x10	100	275	472	361	31,27%	-23,52%	427	600,06	1,41
orb8††	10x10	100	573	982	776	35,43%	-20,98%	391	601,08	1,54
orb9††	10x10	100	659	1066	792	20,18%	-25,70%	425	600,77	1,41
orb10††	10x10	100	681	1184	811	19,09%	-31,50%	411	600,7	1,46

Fonte: Adaptado de Rigo (2018).

APÊNDICE O - TABELA DE RESULTADOS DA CONFIGURAÇÃO 4 DO ALGORITMO PROPOSTO COM O GRUPO DE INSTÂNCIAS VDATA DO AUTOR HURINK ET. AL (1994)

Código Instância	nxm	Quantidade total de operações	Melhor <i>Makespan</i> Conhecido	Melhor Resultado Aleatório	Configuração 4					
					Resultado Obtido	Gap Melhor <i>Makespan</i> Conhecido	Gap Melhor Resultado Aleatório	Iterações Realizadas	Tempo Execução (segundos)	Tempo por Iteração (segundos)
mt06†	6x6	36	47	56	47	0,00%	-16,07%	1327	600,05	0,45
mt10†	10x10	100	655	1098	835	27,48%	-23,95%	386	601,76	1,56
mt20†	20x5	100	1022	1321	1029	0,68%	-22,10%	480	600,61	1,25
la01‡	10x5	50	570	758	580	1,75%	-23,48%	954	600,42	0,63
la02‡	10x5	50	529	715	539	1,89%	-24,62%	980	600,41	0,61
la03‡	10x5	50	477	638	500	4,82%	-21,63%	961	600,35	0,62
la04‡	10x5	50	502	667	519	3,39%	-22,19%	988	600,55	0,61
la05‡	10x5	50	457	629	471	3,06%	-25,12%	948	600,62	0,63
la06‡	15x5	75	799	1002	808	1,13%	-19,36%	655	600,75	0,92
la07‡	15x5	75	749	1029	757	1,07%	-26,43%	652	600,91	0,92
la08‡	15x5	75	765	1008	769	0,52%	-23,71%	645	600,19	0,93
la09‡	15x5	75	853	1079	859	0,70%	-20,39%	649	600,76	0,93
la10‡	15x5	75	804	1038	806	0,25%	-22,35%	606	600,22	0,99
la11‡	20x5	100	1071	1356	1074	0,28%	-20,80%	472	600,22	1,27
la12‡	20x5	100	936	1156	940	0,43%	-18,69%	473	601,03	1,27
la13‡	20x5	100	1038	1327	1043	0,48%	-21,40%	487	601,17	1,23
la14‡	20x5	100	1070	1394	1073	0,28%	-23,03%	486	600,56	1,24
la15‡	20x5	100	1089	1324	1090	0,09%	-17,67%	483	600,39	1,24
la16‡	10x10	100	717	1119	820	14,37%	-26,72%	378	600,5	1,59
la17‡	10x10	100	646	965	814	26,01%	-15,65%	339	600,42	1,77
la18‡	10x10	100	663	1078	863	30,17%	-19,94%	387	601,43	1,55
la19‡	10x10	100	617	1087	909	47,33%	-16,38%	331	600,05	1,81

la20‡	10x10	100	756	1128	954	26,19%	-15,43%	380	601,84	1,58
la21‡	15x10	150	806	1454	1089	35,11%	-25,10%	240	602,57	2,51
la22‡	15x10	150	739	1360	1109	50,07%	-18,46%	250	600,4	2,40
la23‡	15x10	150	815	1501	1131	38,77%	-24,65%	260	600,55	2,31
la24‡	15x10	150	777	1395	979	26,00%	-29,82%	241	600,66	2,49
la25‡	15x10	150	756	1450	1023	35,32%	-29,45%	256	601,26	2,35
la26‡	20x10	200	1054	1790	1370	29,98%	-23,46%	189	600,74	3,18
la27‡	20x10	200	1085	1908	1403	29,31%	-26,47%	194	600,3	3,09
la28‡	20x10	200	1070	1904	1466	37,01%	-23,00%	193	600,59	3,11
la29‡	20x10	200	994	1764	1291	29,88%	-26,81%	195	601,92	3,09
la30‡	20x10	200	1069	1917	1326	24,04%	-30,83%	185	602,7	3,26
la31‡	30x10	300	1520	2449	1816	19,47%	-25,85%	117	605,5	5,18
la32‡	30x10	300	1658	2644	2050	23,64%	-22,47%	124	603,68	4,87
la33‡	30x10	300	1497	2440	1822	21,71%	-25,33%	114	601,78	5,28
la34‡	30x10	300	1535	2462	1739	13,29%	-29,37%	109	605,34	5,55
la35‡	30x10	300	1549	2535	1830	18,14%	-27,81%	117	605,35	5,17
la36‡	15x15	225	948	1855	1408	48,52%	-24,10%	129	604,9	4,69
la37‡	15x15	225	986	2022	1397	41,68%	-30,91%	122	602,09	4,94
la38‡	15x15	225	943	1851	1274	35,10%	-31,17%	133	603,15	4,53
la39‡	15x15	225	922	1822	1344	45,77%	-26,23%	133	600,18	4,51
la40‡	15x15	225	955	1870	1386	45,13%	-25,88%	130	604,07	4,65
abz5†	10x10	100	860	1610	1142	32,79%	-29,07%	344	601,23	1,75
abz6†	10x10	100	742	1252	886	19,41%	-29,23%	325	601,44	1,85
abz7†	20x15	300	495	1035	728	47,07%	-29,66%	88	602,54	6,85
abz8†	20x15	300	509	1105	765	50,29%	-30,77%	95	606,38	6,38
abz9†	20x15	300	500	1064	749	49,80%	-29,61%	95	604,91	6,37
car1‡	11x5	55	5013	6601	5051	0,76%	-23,48%	836	600,57	0,72
car2‡	13x4	52	5930	7308	5944	0,24%	-18,66%	953	600,47	0,63
car3‡	12x5	60	5600	7692	5657	1,02%	-26,46%	798	600,05	0,75
car4‡	14x4	56	5617	8032	7175	27,74%	-10,67%	923	600,27	0,65

car5†	10x6	60	4932	7873	6155	24,80%	-21,82%	709	600,06	0,85
car6†	8x9	72	5486	8649	6149	12,09%	-28,91%	529	600,82	1,14
car7†	7x7	63	4281	5936	5174	20,86%	-12,84%	810	600,12	0,74
car8†	8x8	64	4613	7700	6339	37,42%	-17,68%	617	600,33	0,97
orb1††	10x10	100	695	1130	926	33,24%	-18,05%	370	300,09	0,81
orb2††	10x10	100	620	1117	889	43,39%	-20,41%	358	601,84	1,68
orb3††	10x10	100	648	1127	816	25,93%	-27,60%	365	600,79	1,65
orb4††	10x10	100	753	1191	852	13,15%	-28,46%	352	600,96	1,71
orb5††	10x10	100	584	1023	783	34,08%	-23,46%	364	600,41	1,65
orb6††	10x10	100	715	1093	859	20,14%	-21,41%	341	600,3	1,76
orb7††	10x10	100	275	472	378	37,45%	-19,92%	398	601,2	1,51
orb8††	10x10	100	573	982	770	34,38%	-21,59%	393	601,39	1,53
orb9††	10x10	100	659	1066	820	24,43%	-23,08%	392	601,55	1,53
orb10††	10x10	100	681	1184	902	32,45%	-23,82%	403	600,35	1,49

Fonte: Adaptado de Rigo (2018).