

**UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA**

FELIPE KESROUANI LEMOS

**SEQUENCIAMENTO DE CENTROS PRODUTIVOS COM FORMAÇÃO DE LOTES
CONSIDERANDO OCUPAÇÃO E ATENDIMENTO DE DATAS**

**São Paulo
2009**

FELIPE KESROUANI LEMOS

**SEQUENCIAMENTO DE CENTROS PRODUTIVOS COM FORMAÇÃO DE LOTE
CONSIDERANDO OCUPAÇÃO E ATENDIMENTO DE DATAS**

**Trabalho de formatura apresentado à
Escola Politécnica da Universidade de
São Paulo para a obtenção do diploma
de Engenheiro de Produção.**

São Paulo

2009

FELIPE KESROUANI LEMOS

**SEQUENCIAMENTO DE CENTROS PRODUTIVOS COM FORMAÇÃO DE LOTE
CONSIDERANDO OCUPAÇÃO E ATENDIMENTO DE DATAS**

**Trabalho de formatura apresentado à
Escola Politécnica da Universidade de
São Paulo para a obtenção do diploma
de Engenheiro de Produção.**

Orientador: Prof. Dr. Miguel Cezar Santoro

São Paulo

2009

**AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTES
TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO,
PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.**

FICHA CATALOGRÁFICA

Lemos, Felipe Kesrouani

**Sequenciamento de centros produtivos com formação de
lote considerando ocupação e atendimento de datas / F.K.**

Lemos. -- São Paulo, 2009.

224 p.

**Trabalho de Formatura - Escola Politécnica da Universidade
de São Paulo. Departamento de Engenharia de Produção.**

**1. Sequenciação da produção I. Universidade de São Paulo.
Escola Politécnica. Departamento de Engenharia de Produção
II. t.**

DEDICATÓRIA

A tudo e todos que fizeram esta jornada ser como é.
E Àquele que fez com que tudo fosse exatamente perfeito como foi.

AGRADECIMENTOS

Ao Prof. Dr. Miguel Cezar Santoro, por extrapolar qualquer definição estrita de “orientador” e ser um verdadeiro guia na e ao longo da realização deste trabalho. Agradeço pela paciência, pelo tratamento humano, além das valiosas lições de Engenharia de Produção.

À minha família que me apoiou a vida toda e, em particular nas batalhas que cercaram este estudo. Ao meu pai pelo exemplo de engenheiro e pelo amor que sempre demonstrou; à minha mãe pela proteção, carinho e estímulo, sobretudo para me motivar a continuar; à minha irmã, em cuja competência e vibração eu me espelhei para atingir este intento; e ao meu irmão, companheiro eterno das alegrias e tristezas deste e de todos os desafios.

Aos meus amigos da vida inteira, Lviz por compartilhar as dores de um trabalho final; Teus, pela torcida intensa e sincera; Glê, referência em minha caminhada, que me confortou nos momentos de dúvidas e incertezas; e Alê, minha amiga desde sempre e para sempre, que nem a distância fez diminuir a gratidão por me indicar tudo que eu era capaz.

Aos amigos da Poli, pelo apoio nessa jornada tortuosa, mas prazerosa. Em especial, à Julia, pela revisão do trabalho e pelo socorro nas tempestades que o cercaram.

Ao amigo Osni e à amiga Cris, meus co-orientadores, não só neste TF, mas principalmente na minha vida na Produção. Agradeço as conversas e abraços, que foram de grande valia para terminar este estudo e o serão em toda minha vida.

Aos companheiros da Embraer, não só pelo apoio na realização do trabalho, nem só pelos agradáveis dias de estágio, mas por terem sido a contraparte prática na minha formação de Engenheiro. Em especial, agradeço aos meus companheiros de equipe, Bruno, Carlinhos, Cléber, Corá, Dewey, Jéssica, Marilda, Masseli, Miyoshi, Paulinho, Rodrigo e Ana, por tudo em que me apoiaram e pelo tanto que me ensinaram.

E, finalmente, agradeço a Ana, minha namorada e amiga, minha parceira e companheira, meu exemplo e meu suporte; que me viu sofrer por cada linha de código, vibrar com cada etapa concluída, torcer por cada experimento bem sucedido e chorar pelas incertezas do que estava pela frente neste trabalho. Quem fez cada um dos 40 mil km viajados terem um sentido justo e maravilhoso. Por seu carinho, seu companheirismo, sua alegria, sua motivação, sua ajuda e, sobretudo, por seu amor: conquista maior de minha vida e que faz todo o resto – inclusive este trabalho – ser apenas detalhes.

EPIGRAFE

“Golden are the corners; silvery are the sides; and strawy are the voids.”

Provérbio chinês

(Tradução livre: De ouro são os cantos, de prata são os lados e de palha são os espaços vazios)

RESUMO

LEMOS, F. K. Sequenciamento de centros produtivos com formação de lotes considerando ocupação e atendimento de datas. 2009. 100 f. Trabalho de formatura (Graduação) – Escola Politécnica, Universidade de São Paulo, São Paulo, 2009.

Este trabalho se propõe a analisar a questão do sequenciamento de ordens de produção no contexto de centro produtivos em que as operações apresentam formação de lotes. Trata-se de auxiliar o processo de decisão associado ao *trade-off* entre atendimento de datas e utilização de matérias-primas. Embora seja a intersecção de dois temas bastante clássicos na literatura (corte e empacotamento, e sequenciamento), verificou-se que não há riqueza de abordagens conjuntamente. Heurísticas construtivas foram escolhidas como modelo de solução por sua agilidade de processamento, consonante com as demandas do contexto estudado. Foram abordados 3 diferentes problemas (unidimensional, bidimensional guilhotinado e bidimensional não-guilhotinado), para cada um dos quais 10 variantes de heurísticas foram testadas. Destaca-se o uso de heurísticas construtivas probabilísticas como forma de mitigar a deficiência da baixa exploração do espaço de soluções viáveis da versão determinística. Experimentos computacionais foram realizados para comparar os resultados propostos por tais heurísticas sob diferentes condições de ponderação dos critérios, folga dinâmica e dimensões dos itens. Foi feita uma aplicação prática da heurística tida como a mais adequada para o caso, com resultados satisfatórios do ponto de vista da ocupação dos objetos e englobando o atendimento de datas, antes não levado de maneira padronizada nas operações de tais centros.

Palavras-chave: sequenciamento; corte e empacotamento; heurísticas construtivas.

ABSTRACT

LEMOS, F. K. Work station sequencing with batch operations considering occupation and date accuracy. 2009. 100 f. Final Project (Graduation) – Escola Politécnica, Universidade de São Paulo, São Paulo, 2009.

This work proposes to analyze the issue of sequencing of production orders on the context of work stations with batch operations. It deals with supporting the decision process associated to the trade-off between date accuracy and raw material utilization. Although it's the intersection of two classic themes on literature (cutting and packing, and sequencing), it has been seen that there's a lack of approaches simultaneously. Single path heuristics were chosen as solution model for its processing agility, in consonance with the demands of the studied context. Three different problems were approached (one-dimensional, guillotine two-dimensional and non-guillotine two-dimensional), for which 10 variants of heuristics have been tested with each one. The use of probabilistic single path heuristics is stressed as a way to mitigate the deficiency of low exploration of the viable solutions space of the deterministic version. Computational experiments have been made to compare the results among these heuristics under different conditions of weights of criteria, dynamic slack and items size. A practical implementation of the most adequate heuristic for the case has been done, with satisfactorily results from the point of view of objects occupation, and also taking into account the date accuracy, no taken into consideration on operations in a standard manner before that.

Keywords: sequencing; cutting and packing; single path heuristics.

LISTA DE FIGURAS

Figura 2.1 – Primeiras ramificações da hierarquia de modelos de <i>scheduling</i> para <i>jobshop</i>	36
Figura 2.2 – Ilustração das entidades de um problema de corte e empacotamento	37
Figura 2.3 – Hierarquização dos problemas de corte e empacotamento	40
Figura 2.4 – Exemplos de padrões de corte com restrição de guilhotinagem	42
Figura 2.5 – Exemplos de padrões de corte não-guilhotinados de primeira ordem	42
Figura 3.1 – Ilustração do modelo lógico atual de formação dos lotes	56
Figura 3.2 – Componentes que caracterizam as heurísticas propostas	58
Figura 3.2 – Ilustração do modelo lógico proposto	62
Figura 3.3 – Ilustração do procedimento unidimensional de corte	68
Figura 3.4 – Ilustração do procedimento bidimensional guilhotinado de corte	70
Figura 3.5 – Ilustração do procedimento bidimensional não-guilhotinado de corte	72
Figura 5.1 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema unidimensional	92
Figura 5.2 – Resultado do teste de Kruskal-Wallis para o critério de atendimento de datas do problema unidimensional	93
Figura 5.3 – Resultado do teste de Kruskal-Wallis para o critério ponderado do problema unidimensional	93
Figura C.1 – Teste de homocedasticidade para critério de ocupação	143
Figura C.2 – Teste de homocedasticidade para critério de atendimento de datas	143
Figura C.3 – Teste de homocedasticidade para critério ponderado	143

Figura D.1 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema bidimensional guilhotinado	144
Figura D.2 – Resultado do teste de Kruskal-Wallis para o critério de atendimento de datas do problema bidimensional guilhotinado	144
Figura D.3 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema bidimensional guilhotinado	144
Figura D.4 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema bidimensional não-guilhotinado	145
Figura D.5 – Resultado do teste de Kruskal-Wallis para o critério de atendimento de datas do problema bidimensional não-guilhotinado	145
Figura D.6 – Resultado do teste de Kruskal-Wallis para o critério ponderado do problema bidimensional não-guilhotinado	145

LISTA DE GRÁFICOS

Gráfico 3.1 – Probabilidade de sorteio de uma ordem em cada tipo de viés de escolha	64
Gráfico 3.2 – Desempenho da heurística probabilística em diferentes cenários do número máximo de iterações	65
Gráfico 3.3 – Desempenho da heurística probabilística em diferentes cenários do número máximo de iterações sem ganho	66
Gráfico 3.4 – Desempenho da heurística probabilística em diferentes cenários do número máximo de iterações	66
Gráfico 3.5 – Pesos pela posição do item no padrão de corte em ordem decrescente de folga dinâmica	74
Gráfico 4.1 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 (1D)	95
Gráfico 4.2 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 (1D)	95
Gráfico 4.3 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 (1D)	95
Gráfico 4.4 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 (1D)	95
Gráfico 4.5 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 (1D)	95
Gráfico 4.6 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 (1D)	95
Gráfico 4.7 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 por peso de ocupação (1D)	96

Gráfico 4.8 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 por peso de ocupação (1D)	96
Gráfico 4.9 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 por peso de ocupação (1D)	96
Gráfico 4.10 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 por peso de ocupação (1D)	96
Gráfico 4.11 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 por peso de ocupação (1D)	96
Gráfico 4.12 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 por peso de ocupação (1D)	96
Gráfico 4.13 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)	97
Gráfico 4.14 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)	97
Gráfico 4.15 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)	97
Gráfico 4.16 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)	98
Gráfico 4.17 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)	98
Gráfico 4.18 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)	98
Gráfico 4.19 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)	99
Gráfico 4.20 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)	99

Gráfico 4.21 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)	99
Gráfico 4.22 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)	100
Gráfico 4.23 – Variação percentual média em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)	100
Gráfico 4.24 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)	100
Gráfico 5.1 – Avaliações de atendimento de datas e ocupação dos padrões de corte gerados para o corte de placas (caso unidimensional)	106
Gráfico 5.2 – Avaliações de atendimento de datas e ocupação dos padrões de corte gerados para o corte de placas (caso bidimensional guilhotinado)	107
Gráfico 5.3 – Detalhe dos primeiros padrões de corte gerados para o corte de placas	108
Gráfico 5.4 – Avaliações de atendimento de datas e ocupação dos padrões de corte gerados para o <i>nesting</i> (caso bidimensional não-guilhotinado)	109
Gráfico 5.5 – Detalhe dos primeiros padrões de corte gerados para o <i>nesting</i>	109
Gráfico E.1 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 (2DG)	146
Gráfico E.2 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 (2DG)	146
Gráfico E.3 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 (2DG)	146
Gráfico E.4 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 (2DG)	146

Gráfico E.5 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 (2DG)	146
Gráfico E.6 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 (2DG)	146
Gráfico E.7 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 por peso de ocupação (2DG)	147
Gráfico E.8 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 por peso de ocupação (2DG)	147
Gráfico E.9 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 por peso de ocupação (2DG)	147
Gráfico E.10 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 por peso de ocupação (2DG)	147
Gráfico E.11 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 por peso de ocupação (2DG)	147
Gráfico E.12 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 por peso de ocupação (2DG)	147
Gráfico E.13 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)	148
Gráfico E.14 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)	148
Gráfico E.15 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)	148
Gráfico E.16 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)	149
Gráfico E.17 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)	149

Gráfico E.18 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)	149
Gráfico E.19 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)	150
Gráfico E.20 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)	150
Gráfico E.21 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)	150
Gráfico E.22 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)	151
Gráfico E.23 – Variação percentual média em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)	151
Gráfico E.24 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)	151
Gráfico E.25 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 (2DNG)	152
Gráfico E.26 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 (2DNG)	152
Gráfico E.27 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 (2DNG)	152
Gráfico E.28 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 (2DNG)	152
Gráfico E.29 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 (2DNG)	152
Gráfico E.30 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 (2DNG)	152

Gráfico E.31 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 por peso de ocupação (2DNG)	153
Gráfico E.32 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 por peso de ocupação (2DNG)	153
Gráfico E.33 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 por peso de ocupação (2DNG)	153
Gráfico E.34 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 por peso de ocupação (2DNG)	153
Gráfico E.35 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 por peso de ocupação (2DNG)	153
Gráfico E.36 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 por peso de ocupação (2DNG)	153
Gráfico E.37 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)	154
Gráfico E.38 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)	154
Gráfico E.39 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)	154
Gráfico E.40 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)	155
Gráfico E.41 – Variação percentual média do <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)	155
Gráfico E.42 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)	155
Gráfico E.43 – Variação percentual média da <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)	156

Gráfico E.44 – Porcentagem de sucesso em <u>ocupação</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)	156
Gráfico E.45 – Variação percentual média de <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)	156
Gráfico E.46 – Porcentagem de sucesso em <u>atendimento de datas</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)	157
Gráfico E.47 – Variação percentual média em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)	157
Gráfico E.48 – Porcentagem de sucesso em <u>critério ponderado</u> em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)	157

LISTA DE ILUSTRAÇÕES

Quadro 2.1 – Variantes dos problemas de corte e empacotamento de maximização dos <i>outputs</i>	41
Quadro 2.2 – Variantes dos problemas de corte e empacotamento de minimização dos <i>inputs</i>	41

LISTA DE TABELAS

Tabela 1.1 - Classificação dos centros de trabalho e tratamento do problema de sequenciamento	30
Tabela 1.2 – Classes de problemas e quantificação	33
Tabela 3.1 – Comparação entre os modelos de solução segundo critérios	55
Tabela 4.1 – Níveis do fator “Dimensão média dos itens”	77
Tabela 4.2 – Níveis do fator “Variabilidade das dimensões dos itens”	78
Tabela 4.3 – Níveis do fator “Folga dinâmica média dos itens”	78
Tabela 4.4 – Níveis do fator “Variabilidade da folga dinâmica dos itens”	79
Tabela 4.5 – Níveis do fator “Peso dos critérios”	79
Tabela 4.6 – Níveis do fator “Métodos heurísticos”	80
Tabela 4.7 – Situações experimentais para cada método heurístico	81
Tabela 4.8 – Resultados do critério de ocupação para o problema unidimensional com peso de ocupação de 20% e peso de atendimento de datas de 80%	83
Tabela 4.9 – Resultados do critério de ocupação para o problema unidimensional com peso de ocupação de 50% e peso de atendimento de datas de 50%	84
Tabela 4.10 – Resultados do critério de ocupação para o problema unidimensional com peso de ocupação de 80% e peso de atendimento de datas de 20%	85
Tabela 4.11 – Resultados do critério de atendimento de datas para o problema unidimensional com peso de ocupação de 20% e peso de atendimento de datas de 80%	86

Tabela 4.12 – Resultados do critério de atendimento de datas para o problema unidimensional com peso de ocupação de 50% e peso de atendimento de datas de 50%	87
Tabela 4.13 – Resultados do critério de atendimento de datas para o problema unidimensional com peso de ocupação de 80% e peso de atendimento de datas de 20%	88
Tabela 4.14 – Resultados do critério ponderado para o problema unidimensional com peso de ocupação de 20% e peso de atendimento de datas de 80%	89
TABELA 4.15 – Resultados do critério ponderado para o problema unidimensional com peso de ocupação de 50% e peso de atendimento de datas de 50%	90
TABELA 4.16 – Resultados do critério ponderado para o problema unidimensional com peso de ocupação de 80% e peso de atendimento de datas de 20%	91
TABELA B.1 – Resultados do critério de ocupação para o problema bidimensional guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80%	125
TABELA B.2 – Resultados do critério de ocupação para o problema bidimensional guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%	126
TABELA B.3 – Resultados do critério de ocupação para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%	127
TABELA B.4 – Resultados do critério de atendimento de datas para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%	128
TABELA B.5 – Resultados do critério de atendimento de datas para o problema	129

bidimensional guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

TABELA B.6 – Resultados do critério de atendimento de datas para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20% 130

TABELA B.7 – Resultados do critério ponderado para o problema bidimensional guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80% 131

TABELA B.8 – Resultados do critério ponderado para o problema bidimensional guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50% 132

TABELA B.9 – Resultados do critério ponderado para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20% 133

TABELA B.10 – Resultados do critério de ocupação para o problema bidimensional não- guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80% 134

TABELA B.11 – Resultados do critério de ocupação para o problema bidimensional não-guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50% 135

TABELA B.12 – Resultados do critério de ocupação para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20% 136

TABELA B.13 – Resultados do critério de atendimento de datas para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20% 137

TABELA B.14 – Resultados do critério de atendimento de datas para o problema 138

bidimensional não-guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

TABELA B.15 – Resultados do critério de atendimento de datas para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20% 139

TABELA B.16 – Resultados do critério ponderado para o problema bidimensional não-guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80% 140

TABELA B.17 – Resultados do critério ponderado para o problema bidimensional não-guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50% 141

TABELA B.18 – Resultados do critério ponderado para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20% 142

LISTA DE ABREVIATURAS E SIGLAS

CT: Centro de trabalho

C&E: Corte e empacotamento

MORP: *Minimization of order spread problem*

MOSP: *Minimization of open stack problem*

OP: Ordem de produção

P&D: Pesquisa e Desenvolvimento

PCP: Programação e Controle da Produção

PL: Programação Linear

PLI: Programação Linear Inteira

SA: Sociedade Anônima

TQM: *Total Quality Management*

SUMÁRIO

1	INTRODUÇÃO	27
1.1	A EMPRESA	27
1.2	PROBLEMA.....	28
1.3	ESCOPO	31
2	REVISÃO BIBLIOGRÁFICA.....	35
2.1	SCHEDULING	35
2.2	FORMAÇÃO DE LOTES: CORTE E EMPACOTAMENTO	37
2.2.1	TIPOLOGIA.....	38
2.2.2	ABORDAGENS PARA O PROBLEMA DE OCUPAÇÃO	43
2.2.3	ABORDAGENS ENVOLVENDO SEQUENCIAMENTO	49
3	MATERIAIS E MÉTODOS	53
3.1	DEFINIÇÕES	53
3.2	DETERMINAÇÃO DO TIPO DE ABORDAGEM.....	54
3.3	MODELO LÓGICO ATUAL E PROPOSTO.....	56
3.3.1	CARACTERIZAÇÃO DAS ENTIDADES ENVOLVIDAS	59
3.3.2	CRITÉRIOS DE ORDENAÇÃO INICIAL	61
3.3.3	HEURÍSTICAS DE FORMAÇÃO DE PADRÕES DE CORTE	62
3.3.3.1	VIÉS DA ESCOLHA.....	63
3.3.3.2	CRITÉRIOS DE PARADA	64
3.3.3.3	PROCEDIMENTO DE CORTE E EMPACOTAMENTO	66
3.3.3.3.1	UNIDIMENSIONAL.....	67
3.3.3.3.2	BIDIMENSIONAL GUILHOTINADO	68
3.3.3.3.3	BIDIMENSIONAL NÃO-GUILHOTINADO	71
3.3.4	PROCEDIMENTO DE AVALIAÇÃO DE PADRÕES	73

3.3.4.1	PONDERAÇÃO E NORMALIZAÇÃO DAS AVALIAÇÕES	74
3.4	RESUMO.....	75
4	RESULTADOS EXPERIMENTAIS	77
4.1	CENÁRIOS PROJETADOS	77
4.2	RESULTADOS	80
4.3	ANÁLISE DOS RESULTADOS	92
4.4	DISCUSSÃO DOS EXPERIMENTOS	94
4.4.1	VIÉS DA HEURÍSTICA.....	101
4.4.2	ORDENAÇÃO INICIAL	102
4.4.3	PESO DOS CRITÉRIOS.....	103
4.4.4	DISTRIBUIÇÕES DAS DIMENSÕES E FOLGAS DINÂMICAS.....	103
5	APLICAÇÃO PRÁTICA.....	105
5.1.	RESULTADOS NO CORTE DE SERRA FITA	106
5.2.	RESULTADOS NO CORTE DE PLACAS	107
5.3.	RESULTADOS NO <i>NESTING</i>	108
5.4	DISCUSSÃO DOS RESULTADOS PRÁTICOS	110
6	CONCLUSÕES.....	113
	REFERÊNCIAS	115
	APÊNDICES	123

1 INTRODUÇÃO

1.1 A EMPRESA

O objeto de estudo do presente trabalho de formatura é a Embraer, Empresa Brasileira de Aeronáutica S.A. Trata-se de uma empresa do setor aeronáutico, mais especificamente na produção de aeronaves, que completa em 2009 seu 40º ano de existência, já tendo acumulado uma produção de pouco mais de 5.000 aviões nesse período.

Figura entre as maiores empresas do setor no mundo, liderando o nicho de aeronaves comerciais regionais de médio porte, presente atualmente em 78 países com produtos em operação. Sua produção a colocou no patamar de maior exportadora brasileira entre 1999 e 2001 e segunda maior entre 2002 e 2004, possuindo uma força de trabalho de pouco mais de 17.000 funcionários.

Sua carteira de pedidos firmes – parâmetro muito utilizado no setor para dimensionamento do porte da empresa – totaliza hoje US\$ 19,7 bilhões, tendo apresentado no primeiro trimestre de 2009 uma receita líquida de R\$ 2,7 bilhões e um lucro líquido de R\$ 38 milhões (uma queda significativa em desempenho em relação ao ano anterior, devido à crise econômico-financeira). Apresentou em 2008 um investimento total de R\$ 482 milhões em imobilizado e R\$ 480 milhões em Pesquisa e Desenvolvimento (P&D).

Sua linha de produtos pode ser vista como três grandes famílias, a título de entendimento de divisões de negócio:

- a) **Aviação comercial:** inclui a família ERJ (135, 140, 145 e 145 XR) de jatos até 50 assentos e a família EMBRAER (170, 175, 190 e 195) de jatos de 70 a 122 assentos. Representa a grande fatia de receita da empresa, totalizando 75,3%;
- b) **Aviação executiva:** entrou no mercado em 2000 com os jatos da família Legacy (450, 500 e 600) e hoje conta em seu portfólio com a família Phenom (100 e 300) e o Lineage 1000. É responsável por 6,1% das receitas da empresa;
- c) **Defesa e governo:** fornece mais de 50% da frota aérea brasileira e ainda para cerca de 20 forças aéreas estrangeiras. Apresenta como principais produtos o Super Tucano (projeto desenvolvido com a Força Aérea Brasileira, FAB), o EMB 145

AEW&C, EMB 145 RS/AGS e o P-99. São responsáveis por 5,9% do faturamento da empresa.

Além desses três segmentos, a empresa ainda apresenta outras receitas (12,7%) provenientes de outros serviços aeronáuticos, reposições, *leasings* etc.

O estudo será realizado na unidade de Botucatu (internamente denominada “BOT1”), que é uma das 5 plantas de produção situadas no Brasil. As outras quatro estão em Eugênio de Melo, Gavião Peixoto e São José dos Campos (duas plantas).

A Indústria Aeronáutica Neiva, razão social da Embraer – Unidade Botucatu, passou a integrar o grupo em meados de 2006, sendo anteriormente uma fábrica de aviões agrícolas – o Ipanema – que hoje integra o portfólio de produtos da empresa. Após a incorporação, passou a ser uma espécie de “fornecedor interno” do grupo, produzindo, sobretudo, peças primárias, perfis, revestimentos e submontagens (painéis e fuselagens, dependendo do modelo). Hoje apresenta quase 48.000 m² de área construída e conta com pouco mais de 2.000 colaboradores.

O vínculo que viabilizou a realização do trabalho de formatura nessa empresa é um módulo de estágio em um projeto de consultoria através da Fundação Carlos Alberto Vanzolini. O projeto desenvolvido Unidade Botucatu da Embraer tem como objetivo amplo a melhora da programação e controle da produção em diversos departamentos da unidade (fabricação, montagem, compras e subcontrato), no intuito de reduzir atrasos, estoques intermediários e promover a utilização eficiente dos recursos produtivos. Além disso, objetiva promover o crescimento conceitual em planejamento e programação da produção na planta, sobretudo no próprio departamento de PCP.

1.2 PROBLEMA

O estudo se insere, neste contexto, como uma das frentes do projeto, o **sequenciamento da produção**. Lenstra e Kan (1984) definem um problema de sequenciamento como aquele que busca a alocação ótima das tarefas (*jobs*, que neste trabalho serão chamadas de ordens de produção ou OPs) no tempo, tendo em vista os recursos escassos na forma de máquinas, com a restrição básica de que nenhuma máquina processa mais de uma OP e nenhuma OP é processada por mais de uma máquina, dado um instante temporal. Em

termos gerais, o estudo do sequenciamento é importante por gerar impactos positivos na redução de custos, melhoria de eficiência da operação e aumento da produtividade.

No caso específico da planta estudada, o estudo do sequenciamento também é particularmente relevante: na verdade, trata-se de uma decorrência da extrema importância que o atendimento de datas toma no universo desta fábrica. Sendo uma produtora de peças primárias e submontagens (poderia-se classificá-la como uma fábrica de “aeropeças”), apresenta clientes internos e externos ao site (embora todos dentro da empresa), cuja inserção de seus *outputs* se dá no contexto de grandes montagens. Em outras palavras, o atraso com relação a prazos acarreta paradas nas linhas de montagem intermediárias e finais, almejando-se sempre uma boa sincronização. Por outro lado, o adiantamento da produção gera custos de estoque, tanto financeiros quanto de manutenção física.

Das características do sistema produtivo conclui-se que o objetivo maior do sequenciamento a ser estudado é a minimização do desvio do prazo, porém sem deixar de levar em conta aspectos de utilização. O sequenciamento toma uma dimensão tal que, por um lado garante o nível de atendimento essencial às operações fabris da empresa e, por outro, pode levar a uma utilização e ocupação melhorada dos recursos, de forma a possibilitar uma flexibilidade para futuras expansões (ou retomada da cadência anterior à crise econômico-financeira) ou ainda, de forma mais imediata, a diminuição dos tempos de ciclo nos centros de trabalho.

Atualmente, o sequenciamento da produção é feito nos centros de trabalho através de uma regra de liberação, a folga dinâmica. Por folga dinâmica da uma ordem de produção i (OP_i), entende-se:

$$FDin_i(t) = d_i - \sum_m p_{im} - t \quad (1.1)$$

Onde,

d_i é a data de necessidade (*due date*) da OP_i

$\sum_m p_{im}$ é o tempo de ciclo (tempo de processamento somado ao tempo esperado de fila) restante nos demais m centros de trabalho do roteiro da OP_i

t é o instante no qual se calcula o índice de prioridade.

Em termos de gestão da produção, a folga dinâmica pode ser entendida como um parâmetro que indica quanto tempo a ordem de produção pode atrasar sem prejudicar os tempos das demais etapas de fabricação. Em uma segunda visão (que leva ao mesmo conceito matemático), trata-se da diferença entre a data de necessidade do item e a data de fim previsto deste.

Trata-se de uma regra satisfatória para processos de agrupamento operacional unitário e sem setup significativo (ou mesmo quando é independente de sequência). Jayamohan e Rajendran (2004) mencionam que esta regra de liberação é usada para minimizar o máximo atraso, além de ter a propriedade de reduzir a variância do atraso das OPs. Entretanto, alguns processos apresentam agrupamento em lotes para a operação, *setup* dependente da sequência ou ambos, de forma que há necessidade de um estudo maior acurado do sequenciamento, com regras que otimizem a eficiência de ocupação e/ou eficiência de utilização dessas máquinas, não perdendo de vista o atendimento de datas que é o maior objetivo.

Baseando-se na observação do processo e na interação com as equipes de programação, controle e fabricação, foi possível construir de maneira dialógica uma classificação dos centros de trabalho (CTs) para dividir o problema do sequenciamento em grupos mais homogêneos, levando em conta, sobretudo as características técnicas de processamento. Speranza e Woerlee (1991) ressaltam esta natureza do entendimento do problema de *scheduling* a partir do nível operacional. A tabela 1.1 mostra essa sistematização, com base em uma análise em duas dimensões: característica do *setup* e agrupamento operacional.

Tabela 1.1 - Classificação dos centros de trabalho e tratamento do problema de sequenciamento

	<i>Com setup</i>	
<i>Sem setup</i>	<i>Independente da sequência</i>	<i>Dependente da sequência</i>
<i>Operação unitária</i>	Objetivo: atendimento de datas	Objetivo: atendimento de datas + $\Phi_{\text{utilização}}$
<i>Operação em lote</i>	Objetivo: atendimento de datas + $\Phi_{\text{ocupação}}$	Objetivo: atendimento de datas + $\Phi_{\text{ocupação}} + \Phi_{\text{utilização}}$

O caso de operações unitárias e sem tempo de preparação (*setup*) relevante ou independente da sequência pode ser considerado como aquele em que a folga dinâmica é aplicada de maneira satisfatória. Não existem restrições técnicas que distorçam o objetivo proposto por tal regra.

Para a situação em que ainda não há tempo de preparação dependente da sequência, porém há formação de lotes, é necessário ter em conta, junto ao atendimento de datas, a **ocupação** das unidades operativas. Assim, conduz-se o problema ao campo de estudo de corte e empacotamento combinado com o sequenciamento de padrões. Ocupação aqui será considerada como o indicador que mostra a proporção entre o espaço utilizado em uma unidade de operação e o espaço disponível nela.

Por outro lado, há os centros de trabalho cuja operação é unitária, porém os tempos de preparação das ordens são impactos pela ordem em que são feitas. Neste contexto, a **utilização** do equipamento é um indicador de desempenho a ser melhorado, conduzindo a problemática ao campo da otimização do atendimento de datas em ambientes de *setup* dependente de sequência. Aqui, utilização é definida de forma particular como a proporção entre horas efetiva de fabricação (em operação) e horas disponíveis para trabalho.

Existem, por fim, os casos em que ambas as realidades se encontram, sendo necessário dar atenção simultaneamente à ocupação das unidades de operação e à utilização das máquinas, não desconsiderando, por definição, os prazos.

Para esclarecimento da terminologia, convém ressaltar que segundo a ISO 9000:2000, tanto utilização quanto ocupação acima definidas para este estudo são considerados indicadores de utilização em sentido amplo (quociente entre uma variável de entrada real e outra de entrada planejada). Em termos da classificação utilizada na tabela 1.1, utilização foi tomada em um sentido estrito descrito acima como um indicador de horas de produção efetiva em relação a horas disponíveis.

1.3 ESCOPO

Para efeito de delimitação de escopo, o presente trabalho propõe-se a estudar o caso em que existe operação em lote, sem a presença de *setup* dependente de sequência. Em outras palavras, trata-se de abordar a problemática do *trade-off* entre **ocupação** e **atendimento de**

datas em centros de trabalho com presença de formação de lotes. A escolha do subproblema de deve a uma série de motivos, dentre os quais:

- Trata-se do quadrante com **maior número de centro de trabalho** (25, comparado com 6 no quadrante com setup dependente de sequência e 1 na intersecção entre ambos);
- Neste conjunto, estão incluídos os centro de trabalho no **início do processo** (corte da matéria-prima), momento em que há interesse de corrigir distorções na emissão de ordens. Ordens emitidas com muito atraso precisam ser priorizadas e, em oposição, as muito adiantadas precisam ser retidas para diminuir o estoque em processo;
- Os casos de **setup dependente de sequência são pouco críticos** do ponto de vista do sequenciamento: ou apresentam baixo volume de ordens diariamente ou requerem troca pouco constante de ferramental;
- Há **escassa referência na literatura** a respeito do atendimento de datas relacionado com formação de lotes: trata-se de um par de temas bastante clássicos na literatura, porém ainda pouco explorados conjuntamente. Desta maneira, foi considerado interessante estudá-lo em uma aplicação prática.

Foram excluídos os centros de trabalho em que não havia restrição de agrupamento nos lotes, pois neste caso o problema recai no caso de sequenciamento apenas por atendimento de datas. Um exemplo é o processo de rebarbação, em que qualquer peça pode entrar na máquina com qualquer outra.

Foram identificadas seis classes de problemas, de acordo com a dimensionalidade e a restrição de corte, dos quais quatro delas estão presentes na planta em questão. A tabela 1.2 mostra essas classes de problemas e o número de casos identificados em cada um destes.

Novamente, haverá uma restrição do escopo do problema, desconsiderando os casos tridimensionais. Tal recorte se justifica:

- pela ausência de quaisquer fontes de dados sobre as dimensões tridimensionais dos objetos, inviabilizando um estudo de empacotamento interessante do ponto de vista teórico;
- pela irrelevância prática, dado que são centros de trabalho em que a restrição imposta para a formação de lotes é baixa, isto é, o tamanho relativo das peças

em comparação com os tanques e fornos é muito pequeno. Assim, fica facilitada a formação de lotes, mesmo tendo em vista a questão das datas.

Tabela 1.2 – Classes de problemas e quantificação

<i>Dimensionalidade</i>	<i>Restrição de corte</i>	<i>Número de casos</i>
Unidimensional	-	25
Bidimensional	Guilhotinado	4
Bidimensional	Não-guilhotinado	5
Tridimensional	Guilhotinado em uma dimensão	0
Tridimensional	Guilhotinado em duas dimensões	0
Tridimensional	Não-guilhotinado	15

A estrutura deste trabalho apresenta mais cinco tópicos textuais além desta introdução. Na seção 2 será apresentada uma revisão da literatura tanto para o sequenciamento, quanto para formação de lotes (corte e empacotamento), bem como a intersecção entre ambos. Na seção 3 serão estabelecidas as premissas para a solução do caso apresentado, bem como o desenvolvido dos métodos heurísticos escolhidos como abordagens. A seção 4 abarca os resultados experimentais, decorrente da análise comparativa entre os modelos propostos para definição do formato da aplicação. Já na seção 5 são mostrados os resultados práticos da aplicação, suportados pelos resultados da última seção. A discussão dos resultados, tanto experimentais como práticos, são feitos nas próprias seções de resultados (4 e 5). Finalmente, as principais conclusões deste trabalho ficam sintetizadas na seção 6.

2 REVISÃO BIBLIOGRÁFICA

2.1 SCHEDULING

Wight (1984 apud Herrmann, 2006, p. 1) define *scheduling* como o estabelecimento do *timing* para desempenho de uma tarefa, ou seja, o que deve ser feito primeiro e com que recursos. Nesse sentido, o sequenciamento de ordens é o processo de decisão associado à definição de prioridades, bem como a alocação de capacidade. Ovacik e Uzsoy (1997) enxergam o mesmo problema sob a lógica de alocar máquinas/recursos a ordens a produzir, de forma a otimizar ou, ao menos, melhorar o desempenho.

Pacheco e Santoro (1999) pontuam como principais objetivos do sequenciamento o atendimento de prazos (também chamados de datas de entrega), minimização do tempo de fluxo ou estoques intermediários e maximização da utilização da capacidade disponível.

Henry Gantt foi pioneiro na área do sequenciamento, no início do século XX, embora as primeiras publicações sobre o assunto tenham ocorrido somente no início da década de 50 (PINEDO, 1995). Ovacik e Uzsoy (1997) atribuem a crescente importância do sequenciamento como um subproduto do foco na satisfação do cliente trazida por metodologias como o TQM (*Total Quality Management*). Este novo ambiente de produção forçou a necessidade de reduzir *lead times* e melhorar o desempenho de entregas, mesmo que internas. Pinedo (1995) ressalta, ainda, a importância de um bom sequenciamento em empresas com datas compromissadas, pois seu não-atendimento resulta em perda de imagem.

Mesmo assim, Herrmann (2006) aponta que poucas empresas fabris tiram vantagens dos avanços na área de sequenciamento, já que este potencializou-se recentemente com o avanço da capacidade de processamento informático. Isso pode ser atribuído à complexidade em geral associada a tal tipo de problemas.

Santoro e Pacheco (1999) hierarquizam as tratativas ao problema de sequenciamento em ambiente de *jobshop* (definido como aquele em que há diferentes fluxos das ordens entre as máquinas e diferentes números de operações por ordem). Como se trata de um caso mais geral, esta classificação será considerada neste trabalho para caracterizar as diferentes abordagens na literatura. Entretanto, por efeito de simplicidade e análise comparativa, só serão comparadas com as classes de modelos mais gerais, a saber: de solução ótima e e

heurísticos. Este último subdividido em heurísticas de busca e heurísticas construtivas (ou de passo único). A figura 2.1 mostra esta divisão.

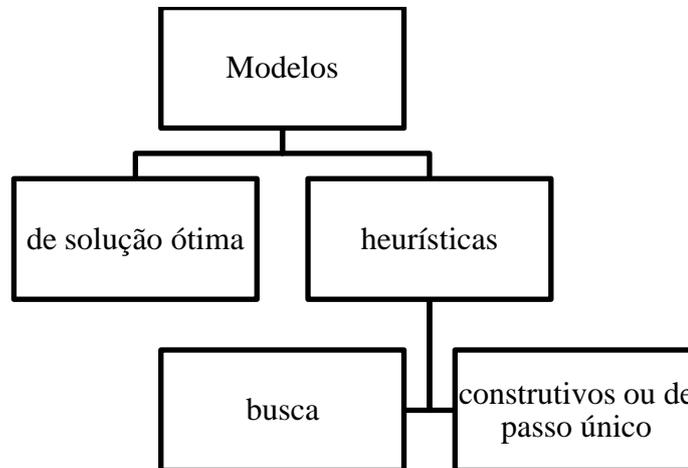


Figura 2.1 – Primeiras ramificações da hierarquia de modelos de scheduling para jobshop

Fonte: SANTORO e PACHECO (1999, p. 10)

Modelos de solução ótima, como o próprio nome diz, são aqueles que geram soluções ótimas, através da definição de uma função objetivo, restrições e parâmetros. Aqui estão incluídos os procedimentos de busca explícita (enumeração), algoritmos eficientes ou busca implícita. Embora garantam a solução ótima, são, em geral, computacionalmente muito pesados para problemas de porte real.

Heurísticas de busca são procedimentos que analisam um subconjunto das alternativas do conjunto de soluções viáveis, fornecendo a melhor como solução. Embora não garantam otimalidade, de acordo com suas regras de busca, intensificação e diversificação podem gerar boas soluções.

Heurísticas construtivas ou de passo único são aqueles que propõem uma sequência e o programa da solução de uma vez só, através de regras que se acredita que conduzam a uma solução de bom desempenho. Embora não garantam o ótimo e não explorem o espaço de soluções possíveis, costumam ser computacionalmente muito leves e guardam relação próxima com características técnicas do problema, sendo mais facilmente assimiláveis e parametrizáveis mediante mudanças no sistema.

2.2 FORMAÇÃO DE LOTES: CORTE E EMPACOTAMENTO

Situações de formação de lotes recaem em geral em um ramo da pesquisa operacional que têm ganhado grande importância nas últimas décadas, os chamados problemas de **corte e empacotamento** (C&E).

A estrutura básica de problemas de corte e empacotamento, segundo Wäscher, Haußner e Schumann (2007), compreende um conjunto de pequenos itens demandados (*outputs*) que devem ser agrupados em subgrupos (padrões de corte) a serem alocados em um conjunto de grandes objetos (*inputs*), obedecendo as restrições geométricas de não sobreposição e posicionamento completo nestes últimos, de forma a otimizar uma função-objetivo.

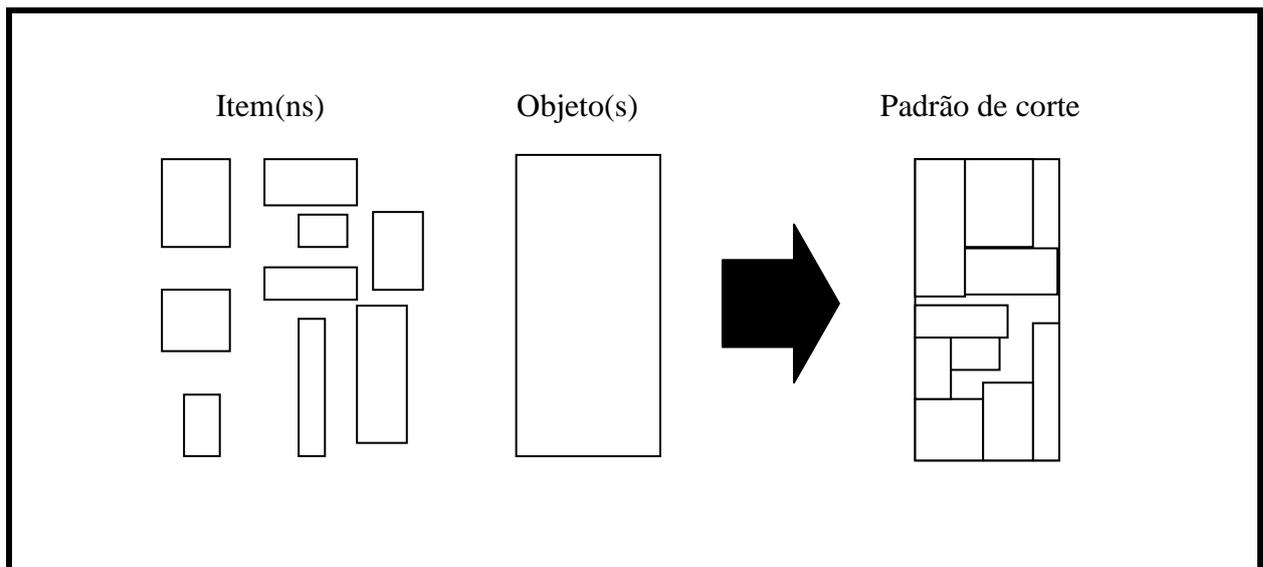


Figura 2.2 – Ilustração das entidades de um problema de corte e empacotamento

Segundo Bichoff e Wäscher (1995), o crescente interesse em pesquisas desta área pode ser explicado pela aplicabilidade do assunto; diversidade de problemas reais; e complexidade dos problemas.

A vastidão de usos é, de fato, facilmente verificável, tanto em aplicações imediatas (indústria de aço, vidro, papel, alumínio etc.), como também em situações semelhantes que recaem na mesma formulação teórica. Brown (1971 apud Bischoff; Wäscher, 1995, p. 1) e Dyckhoff (1990 apud Bischoff; Wäscher, 1995, p. 1) apontam exemplos disso, como

balanceamento de linhas, formulação de orçamentos e *layout* de unidades de produção ou mesmo de publicações de imprensa, esta última ressaltada por Downsland e Downsland, (1995).

A ramificação da questão de formação de lotes em subclasses também fica em clara evidência ao analisar a literatura: problemas de diferentes dimensionalidades, homogeneidade de itens, tipos de unidades de corte e empacotamento, restrições de corte entre outras peculiaridades combinadas fazem com que o campo de origem a inúmeras abordagens distintas. Artigos que propõe tipologias de problemas de corte e empacotamento, como Wäscher, Haußner e Schumann (2007) e Dyckhoff (1990 apud Wäscher, Haußner e Schumann, 2007, 1109) mostram esta diversidade, conforme análise detalhada em item posterior.

Em que se pese a enunciação bastante simples do problema, sua complexidade é, com efeito, alta suficiente para classificá-lo como NP-difícil em termos de processamento, mesmo para os casos simplificados com unidimensionalidade, conforme Dyckhoff (1990 apud Poldi e Arenales, 2006, p. 474), ou ainda para o caso de itens retangulares, conforme Downsland e Downsland (1992).

2.2.1 TIPOLOGIA

Uma primeira tipologia para este tipo de problema é a proposta por Dyckhoff (1990), que apresenta quatro critérios para definir um problema de corte e empacotamento:

- Dimensionalidade: 1 dimensão (1); 2 dimensões (2); 3 dimensões (3); ou n dimensões (N);
- Tipo de atribuição: todos os grandes objetos e uma seleção de pequenos itens (B); ou todos os pequenos itens e uma seleção de pequenos objetos (V);
- Conjunto de grandes objetos: um objeto (O); vários objetos idênticos (I); vários objetos distintos (D);
- Conjunto de pequenos itens: poucos itens de diferentes formatos (F); muitos itens de muitos formatos (M); muitos itens de relativamente poucos formatos (R); ou itens iguais/congruentes (C).

Embora reconheçam a importância desta tipologia para a literatura de corte e empacotamento, Wäscher, Haubner e Schumann (2007) apontam falhas na proposta: (i) restam lacunas de classificação de problemas clássicos (como o carregamento de veículos, *Vehicle Loading Problem*); (ii) há inconsistências entre a proposta e classificações usuais na literatura (como o caso bidimensional em que uma das dimensões do grande objeto é infinita, ou problema de corte de bobinas); (iii) as classes encontradas não são necessariamente homogêneas (especialmente no que diz respeito ao terceiro critério); (iv) além de uma dificuldade de entendimento linguístico decorrente de abreviações não provenientes do inglês.

Wäscher, Haubner e Schumann (2007) apresentam, portanto, uma tipologia melhorada para este tipo de problemas, inserindo novos critérios que abarcassem as novas abordagens surgidas nos últimos anos. Os novos critérios são:

- Dimensionalidade: 1 dimensão (1); 2 dimensões (2); 3 dimensões (3). Problemas n-dimensionais são vistos como variantes;
- Tipo de atribuição: maximização de outputs (valor dos itens alocados); ou minimização de inputs (valor dos objetos utilizados);
- Conjunto de pequenos itens: idênticos; fracamente heterogêneos; e fortemente heterogêneos;
- Conjunto de grandes objetos: um objeto, podendo ter dimensões fixas ou uma ou mais dimensões variáveis; e vários objetos, podendo ser idênticos, fracamente heterogêneos ou fortemente heterogêneos;
- Formato dos pequenos itens: regulares (podendo ser retângulos, círculos, caixas, cilindros, bolas, etc.); ou irregulares (não-regulares).

Partindo desta classificação, são delineados grupos básicos de problemas para cada dimensão, determinados pelo tipo de atribuição e conjunto de pequenos itens. A figura 2.3 mostra essa hierarquização.

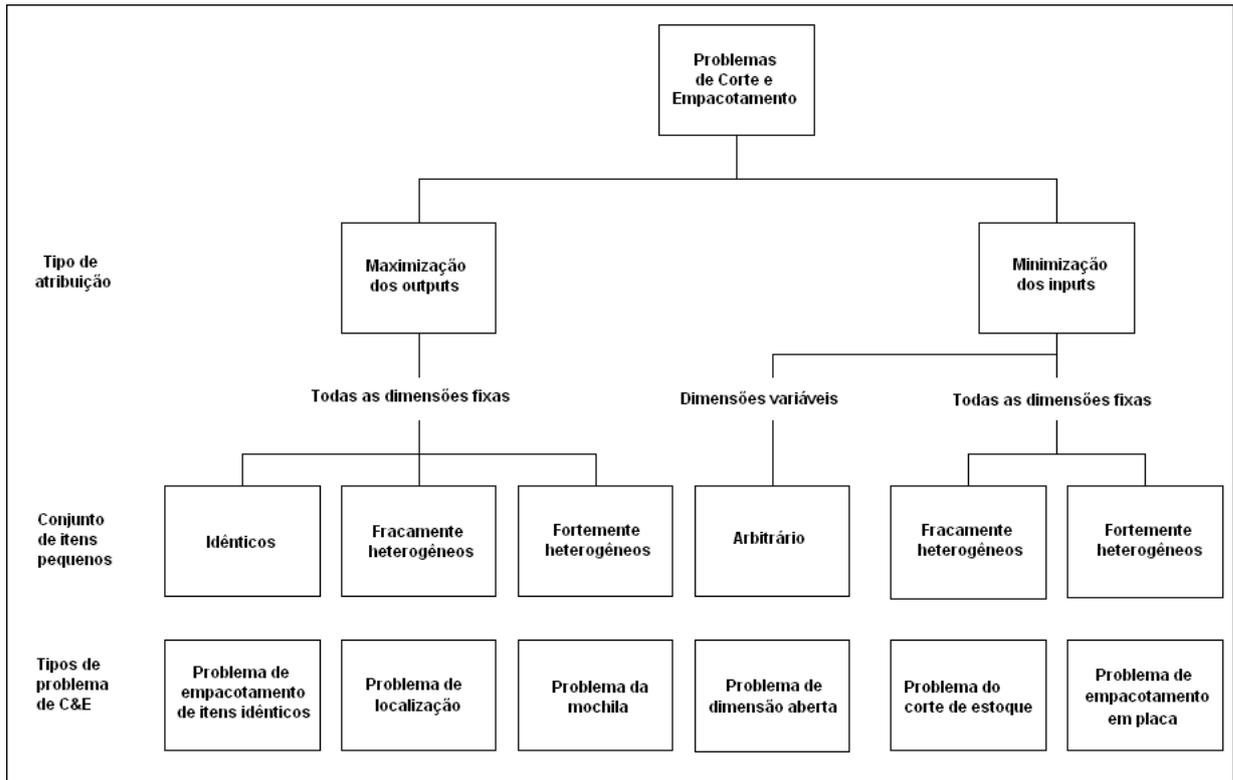


Figura 2.3 – Hierarquização dos problemas de corte e empacotamento

Fonte: WÄSCHER, HAUßNER e SCHUMANN (2007, p. 1117)

Destes grandes grupos de problemas básicos, deriva-se uma subdivisão de variantes de acordo com outras características da tipologia. Os quadros 2.1 e 2.2 mostram esta ramificação, respectivamente para o caso maximização dos *outputs* e minimização dos *inputs*. Em cada célula correspondente foi acrescentado o número de publicações científicas encontradas na revisão de Wäscher, Haubner e Schumann (2007). Foram filtrados *papers* publicados entre 1995 e 2004, em publicações ou conferências internacionais.

Grandes objetos		Pequenos itens	Idênticos	Fracamente heterogêneos	Fortemente heterogêneos
Todas as dimensões fixas	Um objeto		Problema de Empacotamento de Itens Idênticos 23	Problema de alocação em um único objeto 56	Problema da mochila única 86
	Objetos idênticos	X		Problema de alocação em múltiplos objetos idênticos 1	Problema das mochilas múltiplas idênticas 6
	Objetos heterogêneos			Problema de alocação em múltiplos objetos heterogêneos 4	Problema das mochilas múltiplas heterogêneas 6

Quadro 2.1 – Variantes dos problemas de C&E de maximização dos outputs

Fonte: WÄSCHER, HAUßNER e SCHUMANN (2007, p. 1126)

Características dos grandes objetos		Conjunto de pequenos itens	Fracamente heterogêneos	Fortemente heterogêneos
Todas as dimensões fixas	Idênticos		Problema de corte de estoque de tamanho único 38	Problema de empacotamento em objeto finito único 89
	Fracamente heterogêneos		Problema de corte de estoque de tamanhos múltiplos 18	Problema de empacotamento em múltiplos objetos finitos 4
	Fortemente heterogêneos		Problema de corte de estoque residual 10	Problema de empacotamento residual em objeto finito 2
Um grande objeto de dimensão(ões) variáveis			Problema de dimensão aberta 102	

Quadro 2.2 – Variantes dos problemas de C&E de minimização dos inputs

Fonte: WÄSCHER, HAUßNER e SCHUMANN (2007, p. 1126)

Uma sexta característica não abordada em Wäscher, Haubner e Schumann (2007), porém muito encontrada em problemas desta natureza é a restrição de corte, ou seja, se existe trata-se de um corte guilhotinado ou não.

Arenales e Morábito (1995) definem um corte guilhotinado como aquele que produz dois novos retângulos e um padrão de corte guilhotinado como aquele que obtido pela sucessão de cortes guilhotinados. A figura 2.4 mostra um corte guilhotinado e um padrão de corte guilhotinado.



Figura 2.4 – Exemplos de padrões de corte com restrição de guilhotinagem

Fonte: ARENALES e MORÁBITO (1995, p. 600)

Também é definido um corte não-guilhotinado de primeira ordem como aquele que produz cinco novos retângulos de forma que não formem um padrão de corte guilhotinado. A figura 2.5 mostra exemplos de padrões de corte não-guilhotinados de primeira ordem.

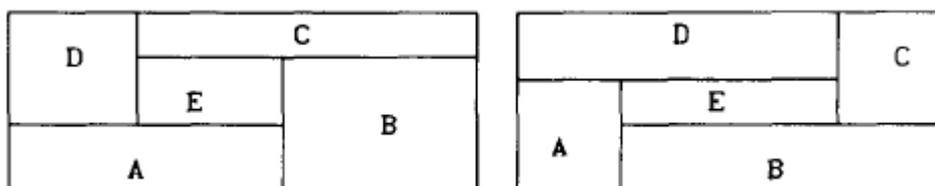


Figura 2.5 – Exemplos de padrões de corte não-guilhotinados de primeira ordem

Fonte: ARENALES e MORÁBITO (1995, p. 601)

Importante ressaltar, embora seja trivial, que quando não há restrição de guilhotinagem, pode-se formar padrões das duas natureza. Em oposição, quando existe restrição de guilhotinagem, apenas pode-se formar padrões guilhotinados. Encontra-se na literatura outras variantes de restrições, como, por exemplo, o número de cortes guilhotinados permitidos na formação de um padrão (chamados de estágios).

Outra pequena variação comum nos problemas de corte e empacotamento é o comumente chamado de problema restrito (*constrained*). Neste tipo de problema não se admite produção de itens acima das quantidades demandadas. São comumente considerados quando o foco da pesquisa é a escolha de padrões de corte pré-definidos ou gerados, pois neste caso pode-se produzir mais itens de um determinado tipo do que o necessário.

2.2.2 ABORDAGENS PARA O PROBLEMA DE OCUPAÇÃO

Como visto anteriormente, um problema de corte e empacotamento é definido como a minimização da ocupação de grandes unidades, obedecendo uma demanda por pequenos itens contidos nestes. Uma formulação clássica, bastante genérica e simplificada para o caso de grandes objetos com dimensões fixas pode ser encontrado em (1)-(3), adaptado de Lodi, Martello e Monaci (2002).

$$\min \sum_{j=1}^M x_j \quad (2.1)$$

Sujeito a

$$\sum_{j=1}^M a_{ij} x_j = d_i \quad \forall i, 1 \leq i \leq n \quad (2.2)$$

$$x_j \in \mathbb{Z} \quad \forall j, 1 \leq j \leq M \quad (2.3)$$

Neste modelo, i é o índice correspondente aos itens demandados por uma quantidade d_i . Já j é o índice correspondente aos padrões de corte utilizados para obter tais itens, sendo x_j a variável inteira que traduz o número de vezes que o padrão foi utilizado, enquanto a matriz a_{ij} retrata os padrões de corte considerados. Cada elemento da matriz diz o número de vezes que um padrão i aparece no padrão j . Em problemas de dimensões variáveis, trata-se de um formulação semelhante, porém minimiza-se o tamanho da dimensão livre e a restrição principal é que a somatória das dimensões na dimensão fixa devem obedecer o máximo do grande objeto em questão.

Embora assemelhe-se a um problema simples, Arcaro (1988) ressalta a complexidade decorrente do problema de geração de padrões a serem considerados. Tal combinatório é pesado suficiente para inviabilizar a otimização na maioria das aplicações práticas do problema, que em geral têm grande porte.

Um dos primeiros artigos sobre o tema foi o do economista russo Kantorovich (1960 apud Haessler e Sweeney, 1991, p. 141), que introduziu a primeira formulação matemática do problema, produzida em 1939. Para que o problema pudesse abranger tamanhos mais significativos demandados pela indústria, foram introduzidos por Gilmore e Gomory (1961, 1963, 1965 apud Morábito, 1994, p. 62). Por não se tratar de um problema comum onde se tem todas as variáveis disponíveis (os padrões de corte possíveis chegam facilmente à casa de milhões), Gilmore e Gomory (1961 apud Morábito, 1994, p. 60) sugere a resolução através da relaxação da programação linear do problema original gerando um subconjunto dos padrões de corte factíveis a cada iteração. Primeiro, uma base inicial factível é obtida (em geral em padrões de um único tipo de item) e depois, a cada iteração do método simplex, um novo padrão de corte é gerado pela solução de um problema da mochila que escolhe os itens através dos valores duais do problema (*shadow costs*). Também Gilmore e Gomory definiram a condição básica para que a relaxação PL do problema original seja aceita como aproximação da PLI original, que é uma demanda suficientemente grande, muito maior do que a razão das dimensões da placa original pelas dimensões da placa final. Gilmore e Gomory (1963, 1965 apud Morábito, 1994, p. 63) ainda apresentaram várias adaptações das formulações originais para as características industriais recorrentes.

Desde então, o problema de corte e empacotamento têm sido estudado em suas diferentes variações vistas no item 2.2.1, através de métodos otimizante ou heurísticas. Serão apresentadas aqui abordagens mais recentes para diferentes variações do problema, apenas desconsiderando o aproveitamento de material (ocupação).

Wang (1983 apud Oliveira e Ferreira, 1990, p. 257-258) propõe um algoritmo para o problema de corte bidimensional que avalia subpadrões de corte através do desperdício interno (ou seja, partes não ocupadas no menor retângulo que inscreve as peças posicionadas). Padrões cujo desperdício interno ultrapassam um dado valor β de aproveitamento são considerados na solução incumbente.

Oliveira e Ferreira (1990) melhoram o algoritmo de Wang (1983 apud Oliveira e Ferreira, 1990, p. 257-258) através de um procedimento de *branch-and-bound* que avalia o

desperdício total potencial que um subpadrão gerará no padrão final, rejeitando aqueles em que esse desperdício ultrapassam um aproveitamento de β .

Haessler e Sweeney (1991) discutem as abordagens até então dos problemas de corte, ressaltando modelos de programação linear, procedimentos heurísticos sequenciais e procedimentos híbridos. Enquanto o primeiro é um método de otimização, o segundo conduz à factibilização rápida do problema, sendo a qualidade dependente do procedimento escolhido. O terceiro, finalmente, utiliza soluções obtidas através de heurísticas como a mencionada para delimitar limitantes inferiores para a otimização, tornando-a mais eficiente.

Morabito e Arenales (1992) introduzem a ideia de utilizar grafos e-ou para problemas de corte bidimensionais, em uma analogia a ambientes de inteligência artificial. Trata-se de um procedimento enumerativo em árvore, cuja estratégia de busca é uma combinação entre duas das mais clássicas (*depth-first* e *hill-climbing*). O nó inicial representa o grande objeto em questão, enquanto os nós finais representam os itens a serem obtidos. Em cada nó, os ramos gerados representam subconjuntos de cortes possíveis na placa intermediária em questão, gerando placas intermediárias de uma mesma ou excludentes (de um corte diferente). Os resultados computacionais se mostraram mais satisfatórios dos que os obtidos até então na literatura.

Dowland e Dowland (1992) fazem uma revisão de abordagens para problemas de empacotamento em duas ou três dimensões, com foco em algoritmos exatos e heurísticas que atendam a problemas de tamanho real. Concluem que abordagens práticas tendem a ser heurísticas pela complexidade NP-completa do problema.

Hadjiconstantinou e Christofides (1995) propõem um procedimento exato de busca em árvore para o problema de empacotamento bidimensional, sob a forma de um problema da mochila (são atribuídos valores aos itens). O limitante superior na busca é obtido através da relaxação lagrangeana da formulação do problema correspondente sem restrição de guilhotinagem. A solução ótima é obtida de forma eficiente computacionalmente, porém apenas para problemas de porte abaixo de 15 itens, ultrapassando o limite estipulado de 800 segundos em alguns casos. Para problemas práticos de maior porte, portanto, trata-se de uma abordagem inviável.

Arenales e Morábito (1995) propõem uma abordagem em grafo E-OU para o problema de corte retangular ortogonal não-guilhotinado em duas dimensões, como uma extensão de trabalhos anteriores com a restrição de guilhotinagem em Morábito et al. (1992

apud Arenales e Morábito, 1995, p. 600) e Morabito e Arenales (1992, 1994 apud Arenales e Morábito, 1995, p. 600). Considera-se o problema como um grafo, no qual um arco representa a decisão de cortar um retângulo e os nós são as possibilidades de padrão de corte. Reduz-se o espaço de soluções a cortes de primeira ordem, ou seja, combinações de cortes guilhotinados e não-guilhotinados simples. Através de um procedimento de *branch-and-bound*, com limitantes superiores e inferiores simples, explorando padrões possíveis através das estratégias de *backtracking* (explorar um ramo completamente) e *hillclimbing* (explorar a próxima ramificação do grafo e escolher o nó com “maior potencial”) combinadas, foi possível obter soluções ótimas para uma gama de problemas propostos na literatura, porém não ultrapassando 50 peças por padrão de corte.

Já Christofides e Hadjiconstantinou (1995) consideram o mesmo problema de corte retangular ortogonal, porém com a restrição de corte guilhotinado. Propõem um algoritmo exato que se baseia em busca em árvore (*tree search*). O número de ramos analisados é limitado por uma relaxação de uma formulação do problema através de programação dinâmica. O procedimento é capaz de encontrar soluções ótimas para diferentes problemas de porte médio e se vale de um esforço computacional que varia de minutos até quase 3 horas para diferentes configurações de itens.

Uma revisão de problemas de *nesting* irregulares é feita por Dowsland e Dowsland (1995), com foco na literatura a respeito de corte tipo *nesting*, problemas de empacotamento, algoritmos de melhora de padrões de corte e detecção computacional de geometrias.

Daza, Alvarenga e Diego (1995) abordam o problema de corte bidimensional retangular guilhotinado com restrição de atendimento exato à demanda (sem itens sobrando). Um algoritmo exato é proposto, baseado em busca informada através de grafos e-ou. A metodologia é uma generalização do método de Wang (1983 apud Daza, Alvarenga e Diego, 1995, p. 642), apresentando aplicações mais amplas e tempo computacional melhorado. Morabito e Arenales (1996) abordam o mesmo problema com a mesma tratativa, porém para o caso de corte guilhotinado (em estágios).

Riehme, Scheithauer e Terno (1996) propõe uma solução otimizante para o problema de duas dimensões com a restrição de guilhotinagem em dois estágios, considerando a peculiaridade de uma demanda com ordens muito variáveis. É feita uma comparação com o modelo clássico de geração de colunas de Gilmore e Gomory (1953 apud Riehme, Scheithauer e Terno, 1996, p. 544). A relaxação do problema inicial é feita de uma forma diferente, considerando o arredondamento para baixo da solução inteira como os padrões de

corte inicial. Os itens não atendidos são os inputs para o problema residual que gera novos padrões (colunas).

Hifi (1997a) desenvolve um algoritmo híbrido para o problema bidimensional guilhotinado que se baseia em busca em árvore, gerando um limitante inferior inicial a partir de uma série de problemas da mochila unidimensionais por programação dinâmica; e limitantes inferiores melhorados a cada novo estágio. A heurística de busca em árvores é uma mistura de duas estratégias: *depth-first search* (explora um ramo em profundidade) e *hill-climbing* (explora as ramificações de um nó). Os resultados são eficientes para casos ponderados ou não, mas ainda assim se limitam a cerca de 40 itens.

Já em Hifi (1997b) o caso bidimensional guilhotinado é explorado com a restrição de não exceder a quantidade demandada de itens. É proposta uma adaptação do algoritmo otimizador de Viswanathan e Bagchi (1993 apud Hifi, 1997b, p. 727) de modo a melhorar a eficiência computacional. A alteração apresenta o mesmo princípio do mencionado anteriormente em Hifi (1997a), melhorando limitantes inferiores e superiores através de problemas da mochila unidimensionais. Hifi (2001) revisita o mesmo procedimento ainda para o caso de restrição de corte em dois estágios, explorando a programação dinâmica para a busca em árvore.

Martello e Vigo (1998) abordam o problema do empacotamento bidimensionais em unidades finitas (*bin packing*) através de abordagens otimizantes. O conteúdo centra-se no desenvolvimento de um limitante inferior mais apertado, de forma a melhorar o desempenho de procedimentos do tipo *branch-and-bound*, aumentando o porte de problemas práticos abarcados, atingindo aqueles com pouco mais de 100 itens.

Em Lodi, Martello e Vigo (1999a) o problema de empacotamento bidimensional em unidade finita com orientação de objetos é visto sob a ótica de procedimentos heurísticos para inicialização de otimização por *branch-and-bound*. São comparados os construtivos *Finite-First-Fit* (FFF) e *Finite-Best-Fit* (FBF) com um de busca (*tabu search*).

Lodi, Martello e Vigo (1999b) fazem um comparativo interessante entre heurísticas determinísticas para diferentes problemas de corte em placas distintas (sem considerar uma das dimensões infinita), variando os parâmetros de obrigatoriedade ou não de orientação da peça e restrição de corte guilhotinado ou livre.

Uma revisão de abordagens com respeito ao problema de empacotamento bidimensional em unidade finita (*bin packing*) é feito em Lodi, Martello e Vigo (2002) e Lodi,

Martello e Monaci (2002). São propostas adaptações de métodos heurísticos construtivos conhecidos para o problema de uma dimensão variável (*strip*) para o caso de dimensões fixas (*bin*), tanto no caso guilhotinado como no não-guilhotinado. Uma discussão grande também é feita acerca de limitantes inferiores e superiores, procurando os mais precisos possíveis para melhorar o desempenho de procedimentos exatos do tipo *branch-and-bound*. Finalmente, uma comparação com o desempenho de metaheurísticas como o tabu search é realizada.

Wu et al. (2002) desenvolve uma heurística construtiva determinística denominada *Less Flexible First* (“menos flexível primeiro”, em uma tradução livre). A proposta é um desenvolvimento de um princípio cognitivo de origem na Antiguidade, que considera os cantos como sendo elementos “menos flexíveis” em termos de alocação de peças, já que enrijecem duas arestas no posicionamento. São exploradas todas as possibilidades de alocação nos cantos, em diferentes posições. Os resultados computacionais mostram que se trata de uma heurística com bom desempenho para problemas com grandes objetos e pequenos itens, com esforço computacional muito reduzido pela natureza determinística.

Lodi e Monaci (2003) estudam o problema da mochila em dois estágios, que pode ser considerado uma generalização do problema de corte guilhotinado em dois estágios. Uma nova formulação de programação linear inteira é proposta e comparada (principalmente em termos de limitantes) com a abordagem clássica de geração de colunas de Gilmore e Gomory (1961 apud Lodi e Monaci, 2003, p. 277). Lodi, Martello e Vigo (2004) fazem um trabalho semelhante, porém sem a limitação de um número de estágios pré-definidos, recaindo no problema de corte guilhotinado.

Poldi e Arenales (2006) estudam heurísticas para o problema unidimensional. Todas são avaliadas sob a ótica de utilizar um padrão construído pela heurística até antes de exceder a demanda por algum dos itens nela contidos. Tais heurísticas são o *First-Fit-Decreasing* (FFD), uma heurística “gulosa” que maximiza a ocupação a unidade de produção e uma heurística dita residual. Esta última consiste da aproximação inteira da relaxação linear com geração de colunas de Gilmore e Gomory (1961 apud Poldi e Arenales, 2006, p. 474) e posterior atualização da demanda para nova geração de padrões de corte. Uma comparação dos procedimentos mostra superioridade das heurísticas residuais, porém é feita uma ressalva que para pequenas demandas costumam ser mais usuais. São propostos procedimentos residuais modificados para abarcar essa última situação, utilizando a geração de colunas.

Binkley e Hagiwara (2007) utilizam heurísticas de busca (algoritmo genético e *simulated annealing*) associadas a uma heurística construtiva: quatro cantos (*four corners*).

Considera como “código genético” os conjuntos de itens que serão posicionados em cada um dos quatro cantos da placa e realiza buscas a partir de soluções iniciais assim definidas. A maior contribuição do trabalho é não utilizar tais heurísticas de maneira parametrizada, mas sim de forma auto-adaptativa. Afirma-se que trata-se dos melhores resultados publicados na literatura até então.

Costa, Gomes e Oliveira (2009) abordam o caso particular de empacotamento de peça irregulares idênticas em uma chapa (*Periodic Packing of Irregular Shapes*). São propostas três heurísticas que são combinações entre construtivas seguidas de busca local.

2.2.3 ABORDAGENS ENVOLVENDO SEQUENCIAMENTO

A literatura a respeito de corte e empacotamento é bastante extensa e inúmeros métodos foram desenvolvidos para diferentes tipo de problemas, como visto até agora. Embora os modelos satisfaçam as restrições de demanda e possibilitem a minimização do desperdício de material para diferentes situações, os padrões de corte gerados interagem em outros níveis de decisão da indústria, como é o caso do sequenciamento.

A grande maioria das abordagens encontradas na literatura se preocupa com o problema de estoque gerado e seu armazenamento, em classes de problemas denominadas por Yanasse (1997) de MOSP (*minimization of open stack problem*) e MORP (*minimization of order spread problem*). Este último é demonstrado ser NP-difícil por Fink e Voß (1999), ainda que Foerster e Wäscher (1998) o classifique como NP-completo por ser uma generalização do problema do caixeiro viajante. O primeiro também é demonstrado ser NP-difícil por Linhares e Yanasse (2002).

Segundo Pileggi, Morabito e Arenales (2007), uma pilha é chamada de “aberta” se a manufatura da ordem em questão foi começada, porém não finalizada – o que só ocorre quando após o processamento do último item desta. Uma vez que os lotes de peças podem estar segmentados em inúmeros padrões de corte, as dificuldades de espaço e fluxo no processo são trazidas à tona. Yanasse (1997) aponta como inconvenientes de ordens inacabadas o alto custo de estoque, manipulação de materiais, potenciais atrasos em entregas (e, possivelmente, de faturamento) ou custos de transporte pela dispersão dos lotes.

Pileggi, Morabito e Arenales (2005) pontua que tais problemas podem ser vistos de maneira sequencial ou integrada. Na primeira, a geração precede a etapa de sequenciamento, ao contrário da outra em que a geração já considera aspecto da ordem de execução. Ainda é apontada a possibilidade de duas formas de abordagem otimizante integrada: uma função-objetivo que combine tanto a ocupação quanto o sequenciamento; ou ainda escolher um dos objetivos e considerar o outro através da imposição de uma restrição de mínimo (procedimento este escolhido pelo autor).

Yuen (1991) apresenta duas heurísticas construtivas para contemplar o problema do número de “pilhas abertas” (ordens cujo lote ainda não foi inteiramente cortado). Dados padrões de corte gerados e um padrão inicialmente escolhido arbitrariamente a primeira heurística escolhe o próximo padrão em dois passos: primeiro seleciona-se as peças com pilhas abertas e calcula-se o número de padrões não sequenciados que as contém como critério de seleção e a quantidade a ser produzida como desempate; e, posteriormente, escolhida a peça a ser cortada, seleciona-se entre os padrões que a contém o que apresenta menos quantidade de repetição como critério de seleção e com menor número de peças diferentes como critério de desempate. A segunda heurística apresenta-se também em dois passos, sendo o primeiro igual e o segundo utilizando como critério seletor o menor número de pilhas novas abertas e como desempate o padrão que tiver o maior número de pilhas em comum com as abertas. Os resultados mostram uma fila média de 6 pilhas para um tamanho de itens de aproximadamente $\frac{1}{4}$ da placa e 13 pilhas quando a proporção é ao redor de $\frac{1}{25}$.

Há continuidade do estudo em Yuen (1995), no qual 4 novas heurísticas, também construtivas são propostas. A ideia básica destas é a escolha do próximo padrão a ser cortado com base em um índice de compatibilidade, seja ele baseado no saldo entre pilhas encerradas e abertas, no menor número de pilhas abertas ou variantes que enxergam os próximos dois padrões. As heurísticas são comparadas em termo de fila média e tempo computacional despendido.

Yanasse (1997) propõe um algoritmo com base no procedimento de *branch-and-bound* para o problema de minimizar o máximo número de pilhas abertas, além de uma proposta de formulação do problema. A estratégia de exploração é por profundidade e a escolha do nó a ser explorado é feita por um procedimento de busca guloso. Não são reportados resultados numéricos, nem o desempenho computacional do algoritmo.

Faggioli e Bentivoglio (1998) também apresentam uma abordagem otimizante, gerando uma enumeração implícita a partir de um limitante inferior gerado por uma heurística de

busca do tipo *tabu-search*. A solução semente para a busca é gerada por uma heurística construtiva gulosa hierarquizando critérios de não abrir novas pilhas, fechar pilhas e completar pilhas abertas nesta ordem. Os resultados mostram superioridade sobre as heurísticas de Yuen (1995), encontrando a solução ótima em tempo computacional razoável para problemas até 40 padrões de corte. Para problemas maiores, sugere-se que somente o procedimento de busca tabu supera as heurísticas propostas até então apresentadas na literatura.

Um procedimento de busca por *simulated annealing* é feito por Foerster e Wäscher (1998), considerando o MORP. Estabelece-se um *benchmarking* com o procedimento de Madsen (1988 apud Foerster e Wäscher, 1998, p. 277) e o 3-opt-procedure, que consiste na geração de uma seqüência, divisão em três sub-segmentos e posterior otimização de cada um destes. Os resultados mostram superioridade na qualidade da solução em relação ao primeiro; e em esforço computacional (para um mesmo patamar de qualidade gerado) em relação ao segundo.

Em Fink e Voß (1999) ambos problemas (MOSP e MORP) são abordados através de heurísticas de busca. Soluções iniciais são dadas por uma heurística construtiva que se baseia no menor custo de inserção de um padrão de corte, definido através de uma penalidade ponderada das unidades de tempo despendidas. Foram implementadas tanto heurísticas do tipo *tabu search* como *simulated annealing*. A primeira mostrou superior para o MORP e o segundo para o MOSP. Os resultados do MORP foram comparados com Foerster e Wäscher (1998), mostrando resultados superiores.

Armbruster (2002) traz uma solução construtiva aplicada ao problema unidimensional de uma indústria de aço. O trabalho inova ao tratar da geração de padrões de corte já visando o segundo objetivo de sequenciamento, além da tradicional minimização do desperdício de matéria-prima. O processo é composto de três partes: geração de padrões através do algoritmo de Gilmore e Gomory (1961 apud Armbruster, 2002, p. 332); geração de uma solução viável inteira com os padrões; e verificação da viabilidade técnica da solução criada. O algoritmo apresenta soluções de qualidade em tempo computacional razoável, sem *benchmarking* com outros trabalhos.

Becceneri, Yanasse e Soma (2004) exploram um procedimento de *branch-and-bound* com uma heurística de escolha do ramo a ser explorado que usa um conceito de “custo do nó”. Tal medida é relacionada com o número de arcos restantes até encerrar sua exploração. Os resultados utilizam Yuen (1995) como *benchmarking* e soluções de melhor qualidade são

obtidas a custo de um desempenho computacional significativamente pior, embora classificado como “não-proibitivo” pelo autor, mesmo para problemas grandes (150 itens).

Pileggi, Morabito e Arenales (2005) propõem algoritmos para o caso unidimensional do MOSP, abordado de maneira integrada (geração e sequenciamento), impondo restrições mínimas de ocupação e máximo de pilhas abertas. Três abordagens são comparadas: na primeira resolve-se o problema de maneira tradicional (sequenciamento após a geração), iterando a geração enquanto não há solução viável; uma segunda otimiza o processo de geração impondo um número máximo de pilhas abertas ao sequenciar, iterando os padrões de maneira a atender a demanda; e, por fim, a última utiliza o método simplex para gerar colunas de forma a atender a uma restrição de número máximo de pilhas abertas. Os resultados mostraram superioridade da primeira abordagem em termos de qualidade das primeiras colocações e global, porém a terceira abordagem – em termo de esforço computacional total – apresentou-se melhor.

Em Pileggi, Morabito e Arenales (2007) as mesmas três abordagens são revisitadas para o mesmo problema, porém para o caso bidimensional guilhotinado. O resultado em termos de qualidade das soluções mantém a mesma ordem do caso unidimensional, porém em termos de esforço computacional aqui a abordagem 2 é superior. A curva de *trade-off* entre máximo de pilhas abertas e percentual de ocupação mostra que não existe abordagem dominante.

Yanasse e Lamosa (2007) utilizam relaxação Lagrangeana da formulação linear do MOSP unidimensional, para sua solução exata. Embora o ótimo seja atingido, o número de itens é muito restrito para um tempo computacional razoável: para um máximo imposto de duas pilhas abertas o algoritmo suporta apenas dois tipos de itens.

Belov e Scheithauer (2007) vai da consideração do número de pilhas abertas e também considera o número de diferentes padrões, por considerá-los relacionado ao tempo de *setup* de fabricação. É realizada uma abordagem em três passos: uma heurística para minimizar o *input* de material; uma imposição de máximo de pilhas abertas a esse modelo; e, posteriormente, uma simplificação para abarcar os tempos de *setup* de troca de padrões de corte. Os resultados mostraram bastante práticos, no sentido em que impõe uma restrição de 3 pilhas abertas, sacrificando apenas 1% de perda de material extra para um *mix* de 150 tipos de produtos.

3 MATERIAIS E MÉTODOS

Esta seção tem como objetivo definir os métodos de soluções a serem comparados nos experimentos e posteriormente aplicados no caso real. Para isso, estrutura-se alinhando os termos técnicos envolvidos na pesquisa na subseção 3.1; uma determinação do tipo de abordagem para o problema é feito na subseção 3.2; os métodos em si são descritos na subseção 3.3; e, finalmente, um resumo é feito na subseção 3.4.

3.1 DEFINIÇÕES

Com o intuito de evitar ambiguidades com relação à linguagem, são apresentadas aqui as principais definições dos termos específicos utilizados ao longo do trabalho, em consonância com a revisão bibliográfica realizada.

- **Objeto:** trata-se da unidade maior a ser cortada ou servir de limite para empacotamento de itens. Determina as restrições dimensionais e geométricas de acordo com seu formato. Aqui, distinguem-se na fábrica em questão dois tipos de objetos: os objetos-padrão, unidades estocadas da maneira como foram compradas; e os retalhos, unidades que surgem do subproduto de atividades de corte anteriores que não preenchem completamente o objeto por falta de itens;
- **Item:** é a unidade menor a ser posicionada nos objetos, correspondendo a uma peça a ser produzida. Como aqui lida-se com um caso real, estes estão agrupados em ordens de produção, que apresentam itens iguais em uma dada quantidade;
- **Padrão de corte:** é um arranjo de itens em um determinado objeto, ou seja, quais, quantos e em que posição estão encaixados;
- **Agrupamento técnico:** é a característica técnica (como material, espessura, liga etc.) de objetos e itens que restringem sua combinação entre si, de maneira biunívoca, na formação de um padrão de corte. Em outras palavras, itens de um determinado agrupamento técnico só podem ser posicionados em objetos com esta mesma característica.

3.2 DETERMINAÇÃO DO TIPO DE ABORDAGEM

Para efeito de foco da análise e de construção de uma abordagem para o problema em questão, é preciso especificar parâmetros de tratamento. Aqui serão determinados dois deles: o ambiente de produção (*shop environment*), segundo a classificação de Graham et al. (1979 apud Allahverdi et al., 2008, p. 988); e o tipo de modelo de solução, segundo a hierarquização de Santoro e Pacheco (1999).

Em termos de ambiente de produção, será considerado um seqüenciamento em ambiente de 1 máquinas ($\alpha = 1$). Esta escolha se justifica, sobretudo, pelo escopo do projeto, que não se propõe a interferir no método de seqüenciamento dos demais centros de trabalho. Assim, uma consideração mais geral como o ambiente de *jobshop* geraria uma interferência fora da esfera de implementação do projeto.

Já em relação ao modelo de solução, com base na análise da revisão bibliográfica, pôde-se comparar as três alternativas, o que foi mostrado na tabela 3.1.

Partindo de tais informações e dos objetivos do projeto em questão, considera-se que o uso de **heurísticas construtivas** é o mais adequado para tratar o problema, devido a alguns pontos:

- O porte dos problemas deve ser necessariamente grande, pois os centros de trabalho têm listas de ordens que chegam à casa das centenas;
- O esforço computacional pequeno é uma característica muito desejável. Herrmann (2006) caracteriza um bom sistema de sequenciamento ao definir como requisito não só uma seqüência de alta qualidade, como também a propriedade de reagir de forma ágil a eventos, de modo a possibilitar a revisão a um custo razoável, tanto computacional como financeiro. Tais eventos podem ser desde uma falha nos recursos (humanos ou máquinas) até a simples chegada de uma ordem prioritária na fila em questão. Como se trata de um ambiente de produção em que a chegada de ordens é uma realidade constante, a reformulação rápida do sistema de seqüenciamento é uma característica altamente desejável;

Tabela 3.1 – Comparação entre os modelos de solução segundo critérios

	<i>De solução ótima</i>	<i>Heurística de busca</i>	<i>Heurística construtiva</i>
<i>Qualidade da solução</i>	Garante otimalidade	Depende das regras de diversificação e intensificação	Depende da regra de formação de sequência
<i>Porte dos problemas</i>	Pequeno porte	Médio porte, pois a qualidade da solução depende do tempo de processamento	Grande porte
<i>Exploração do domínio de soluções viáveis</i>	Explora explicita ou implicitamente todas as soluções	Depende da parametrização e do esforço computacional	Explora pouco o espaço de soluções viáveis.
<i>Esforço computacional</i>	Grande, sendo proibitivo para problemas de porte real	Depende da parametrização, mas em geral é grande para obter solução de qualidade	Pequeno
<i>Entendimento da modelagem pelos usuários</i>	Baixa	Baixa (parametrização mais próxima à teoria)	Alta, pois parametrização e funcionamento são mais próximos a características do processo

- A acessibilidade do conhecimento envolvido também se torna decisiva no contexto do projeto, uma vez que um de seus objetivos é promover o conhecimento e a melhora dos processos no longo prazo. Neste sentido, um modelo mais próximo a características do processo faz com que possíveis reações a mudanças nas peculiaridades do processo (*mix* de produtos, cadência, novas tecnologias etc.) possam ser re-parametrizadas pela equipe, bem como a gestão participativa dos resultados pelo próprio pessoal da produção;
- A baixa exploração do domínio das alternativas viáveis pode ser mitigada através de múltiplas rodadas em geração de diversas sequências probabilisticamente.

3.3 MODELO LÓGICO ATUAL E PROPOSTO

O sequenciamento dos centros de trabalho (CTs) da empresa apresenta como *input* inicial uma lista de ordens a serem executadas, ordenadas de maneira decrescente de folga dinâmica. No caso específico dos CTs com formação de lotes, além desta informação, também existe a entrada do agrupamento técnico da ordem.

O sequenciamento é feito, atualmente, através de um procedimento intuitivo que pode ser descrito pelos seguintes passos:

1. Toma-se a ordem prioritária (menor folga dinâmica);
2. Toma-se um objeto arbitrário (padrão ou retalho);
3. Combina-se ordens do mesmo agrupamento técnico para formação do lote a ser executado.

A figura 3.1 ilustra tal procedimento, onde os círculos são ordens de produção seqüenciadas por folga dinâmica e sua cor representa o agrupamento técnico a que pertence. É importante ressaltar que não se trata de um procedimento formal, mas sim uma retratação da lógica utilizada pelos operadores para a determinação dos lotes. Desta flexibilidade surgem pequenos ajustes que fogem à regra, como, por exemplo, a não execução de ordens em que não há suficientes ordens semelhantes (mesmo agrupamento técnico) para formar um lote com boa ocupação.

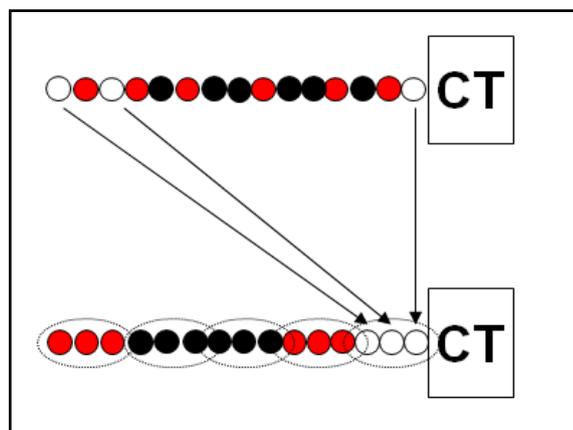


Figura 3.1 – Ilustração do modelo lógico atual de formação dos lotes

Embora seja uma possibilidade de sequenciamento efetiva, a própria figura já traz em si um questionamento ao modelo lógico adotado: eventualmente podem ocorrer situações em que ordens são executadas em grande adiantamento devido à existência de uma única ordem mais atrasada de mesmo agrupamento técnico.

A proposta do presente estudo é incorporar uma nova lógica ao processo de determinação dos lotes e seu posterior sequenciamento. Tal processo de escolha e sequenciamento pode ser descrito pelos seguintes passos lógicos:

1. Ordenação inicial da **fila de itens** por agrupamento técnico e um determinado critério de ordenação inicial, excluindo itens inativos (folga dinâmica maior que um parâmetro inicial máximo estabelecido);
2. Ordenação inicial dos **objetos em estoque** por agrupamento técnico e prioridade de uso;
3. Toma-se o conjunto de itens ativos do primeiro **agrupamento técnico**:
 - a. Para cada objeto deste agrupamento é formado um padrão de corte de acordo com a heurística determinada (Saída: um padrão de corte para cada objeto prioritário);
 - b. Os padrões de corte formados para cada objeto são comparados entre si e o melhor é escolhido por critérios de atendimento de datas e ocupação, explicados adiante (Saída: melhor padrão de corte do agrupamento técnico);
 - c. Os itens do padrão de corte escolhido são inativados para as próximas iterações e o objeto tem uma unidade debitada de seu estoque inicial;
 - d. Retorna ao item a enquanto houver itens ativos na fila.
4. Passa ao próximo agrupamento técnico e retorna a 3 enquanto houver algum com itens ainda ativos.
5. Todos os padrões gerados nos diferentes agrupamentos técnicos são sequenciados através de uma comparação entre si, levando em conta atendimento de datas e ocupação (Saída: todos os padrões sucessivamente gerados em 3b, sequenciados pelo critério estabelecido).

A figura 3.2 retrata o modelo lógico proposta a ser estudado e desenvolvido.

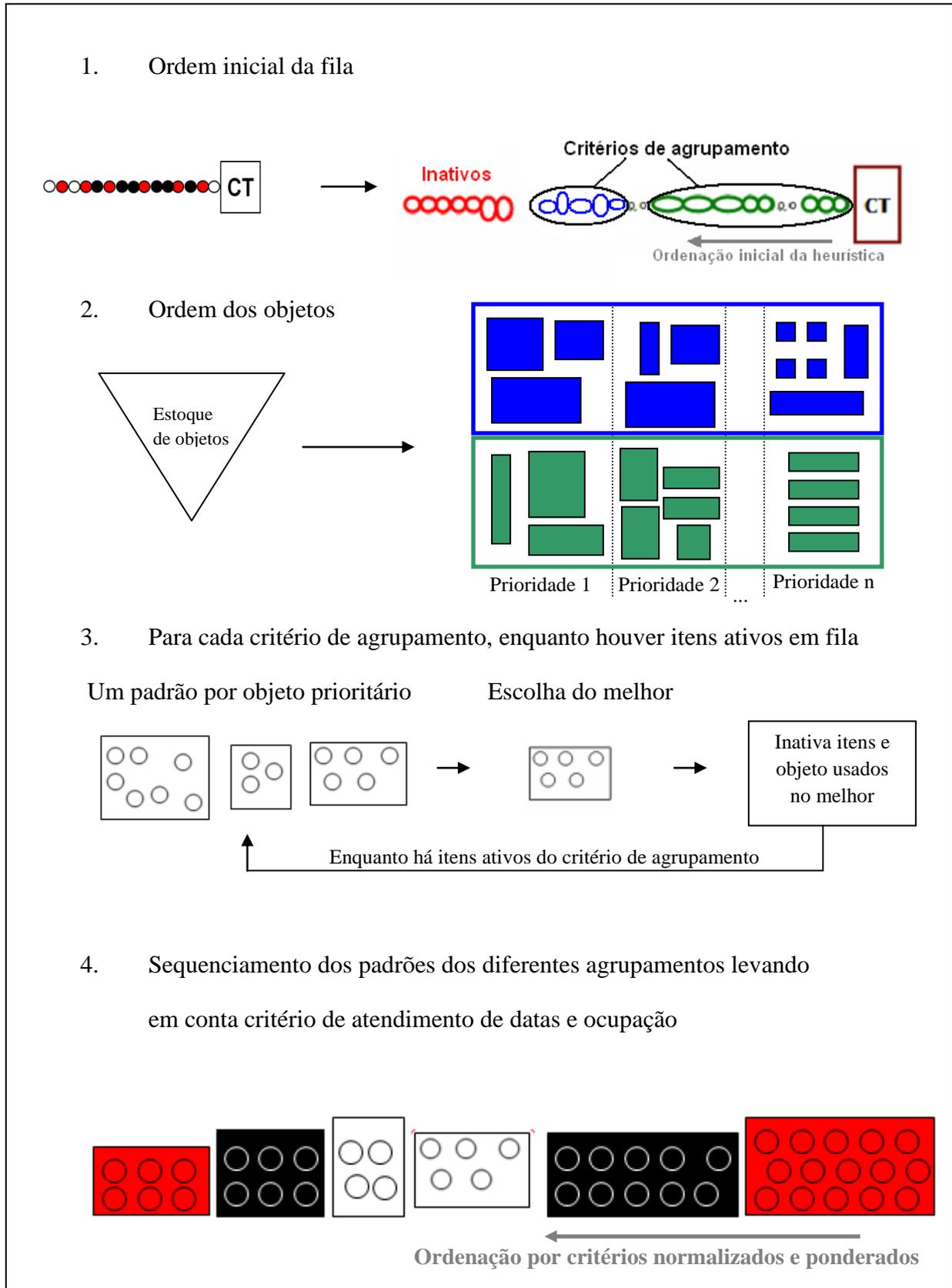


Figura 3.2 – Ilustração do modelo lógico proposto

Posto o modelo genérico de sequenciamento proposto, nas próximas subseções são detalhados os seguintes aspectos:

- a. Entidades envolvidas e seus atributos;
- b. Critérios de ordenação inicial (passo 1 descrito anteriormente);
- c. Heurísticas de formação de padrões de corte (passo 3a descrito anteriormente);
- d. Critérios de avaliação entre padrões (passo 3b e 4);

3.3.1 CARACTERIZAÇÃO DAS ENTIDADES ENVOLVIDAS

Como visto na literatura, os problemas de corte e empacotamento se caracterizam pela interação entre dois tipos de entidades – itens e objetos –, cuja combinação determina uma terceira – os padrões de corte.

Os objetos são caracterizados pelos seguintes atributos:

- **Código do objeto:** chave primária de identificação da entidade, dando unicidade quando da atribuição de um padrão de corte;
- **Critério de agrupamento:** característica técnica a que atende para a formação de lotes;
- **Dimensão 1:** uma das dimensões física do objeto. Aqui convencionou-se por uma questão de simplificação, porém sem perda de generalidade, que é a dimensão da “direção de grão”. Tal conceito advém da engenharia de materiais, e é relevante dado que há determinações do projeto do produto das peças de aeronaves que impõe um alinhamento em determinada direção do material;
- **Dimensão 2:** outra dimensão física do objeto, perpendicular à direção de grão;
- **Quantidade disponível:** limita o número de unidades que um determina objeto tem em estoque;
- **Padrão ou retalho:** atributo binário que determina se um objeto é padrão (em geral, com quantidades muito grandes em estoque, não sendo uma restrição) ou retalho (subproduto de cortes anteriores, que estão em quantidade limitada);
- **Prioridade:** pensando em termos de sistema e aplicação prática, o usuário do modelo tem o poder de decisão de fazer um recorte do conjunto de objetos a

ser considerado prioritariamente. Isso é particularmente relevante pela existência de programas como o 5S, em que periodicamente deseja-se diminuir o número de retalhos em estoque.

Os itens, por sua vez, são entidades que derivam das ordens de produção. Aqui, ao contrário do modelo lógico atualmente utilizado na empresa, não será considerada a restrição de que uma mesma ordem precisa necessariamente estar em um mesmo lote. Embora os critérios de ordenação inicial darão igual prioridade aos diferentes itens de uma mesma ordem, tendendo a estarem juntas em um padrão de corte, não se trata de uma restrição técnica ao problema. O número de itens gerados a partir de uma ordem é tal qual a quantidade de peças desta. Uma observação importante é que os itens irregulares serão considerados dimensionalmente por seu envoltório retangular. Os atributos que caracterizam um item são:

- **Código do item:** chave primária de identificação da entidade, dando unicidade quando da atribuição de um padrão de corte;
- **Critério de agrupamento:** característica técnica que determina em que objeto o item pode ser posicionado;
- **Dimensão 1:** dimensão horizontal do item. Caso exista direção obrigatória por determinação da direção de grão por projeto, a dimensão 1 armazena tal valor;
- **Dimensão 2:** dimensão vertical do item. Em caso de orientação obrigatória, é a dimensão perpendicular à direção de grão;
- **Critério de atendimento de datas:** tal atributo representa a urgência do item segundo um dado critério. Neste estudo, foi tomada a folga dinâmica como critério, já que se trata de um indicador adequado à análise em ambiente de máquina única. Além disso, é o critério utilizado pela empresa nos demais centros de trabalho;
- **Orientação:** atributo binário que determina se o item tem uma orientação obrigatória ou se a rotação é permitida.

Finalmente, a entidade padrão de corte, que é um arranjo de itens em um objeto, apresenta como atributos os seguintes:

- **Critério de agrupamento:** característica técnica tanto do objeto quanto dos itens nele posicionados;
- **Código do objeto:** código do objeto utilizado no padrão de corte;

- **Código dos itens:** atributo correspondente a um vetor de códigos de itens posicionados em tal padrão;
- **Valor de avaliação por atendimento de datas:** conforme será detalhado posteriormente, cada padrão de corte recebe um valor avaliado sob a ótica do atendimento de datas;
- **Valor de avaliação por ocupação:** analogamente ao atributo anterior, o padrão de corte também é avaliado sob o ponto de vista da ocupação do objeto utilizado;
- **Valor ponderado:** trata-se do valor a ser considerado na priorização de um padrão de corte, advindo da normalização e ponderação dos valores anteriormente mencionados.

3.3.2 CRITÉRIOS DE ORDENAÇÃO INICIAL

Uma vez que os modelos construtivos foram escolhidos para a abordagem, é pressuposto que os itens disponíveis para alocação estejam ordenados conforme um critério inicial, como indicado no passo 1 o subitem 3.3. Antes da escolha de um método a ser implantado, nesta aplicação serão testadas três ordenações iniciais:

- **Prioridade por data:** ordenação crescente por folga dinâmica, que retrata a urgência do item no centro de trabalho;
- **Prioridade por ocupação:** em consonância com modelos construtivos vistos na literatura, a prioridade por ocupação será dada por ordem decrecente da altura do item (LODI, MARTELLO e VIGO, 1999a; LODI, MARTELLO e VIGO, 1999b; LODI, MARTELLO e VIGO, 2002; LODI, MARTELLO e MONACI, 2002; WU et al, 2002; POLDI e ARENALES, 2006; BINKLEY e HAGIWARA, 2007). Aqui, a altura será considerada como sendo a dimensão paralela à direção de grão (geralmente maior nos objetos) quando há tal imposição, ou a maior dimensão quando não há tal restrição.
- **Prioridade por critério ponderado:** com a mesma ponderação para a importância entre ocupação e atendimento de datas utilizada na avaliação dos padrões, os critérios anteriores podem ser normalizados e ponderados. Tais

procedimentos serão análogos aos descritos na subseção 3.3.4, em que os critérios de avaliação também são normalizados e ponderados.

3.3.3 HEURÍSTICAS DE FORMAÇÃO DE PADRÕES DE CORTE

O passo 3a descrito na subseção 3.3 pressupõe a existência de uma heurística que recebe como entradas um conjunto de itens (inicialmente ordenados por um dos critérios visto anteriormente) e um objeto e retorna um padrão de corte. Neste estudo, considera-se que a forma de funcionamento da heurística apresenta dois componentes, conforme ilustra a figura 3.3.

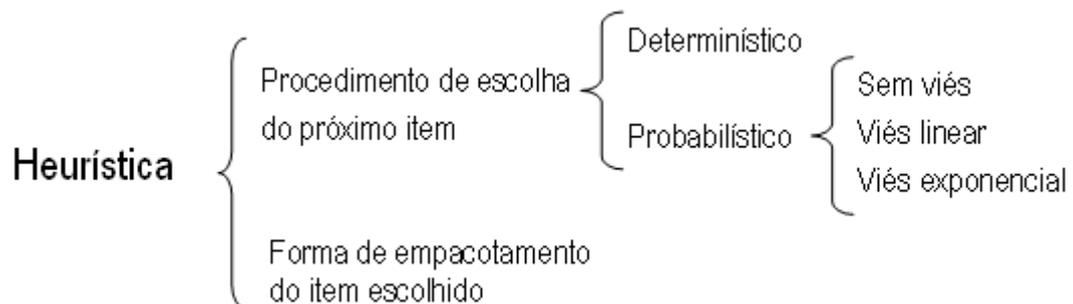


Figura 3.2 – Componentes que caracterizam as heurísticas propostas

O **procedimento heurístico determinístico** toma a ordenação inicial dos itens como a ordem exata na qual itens serão posicionados no padrão de corte. Assim, o item a ser posicionado é o primeiro item viável (que “cabe” no objeto) da fila em questão.

Já o **procedimento heurístico probabilístico**, em oposição, atribui aos itens candidatos viáveis uma determinada probabilidade, de acordo com um viés associado à sua posição na fila. O item a ser posicionado é, portanto, escolhido pela geração de um número aleatório, comparado com tal probabilidade. Como as heurísticas construtivas apresentam baixo esforço computacional, é possível gerar uma gama de soluções distintas soluções probabilisticamente e compará-las ao final, tomando ao melhor. Desta maneira, aproveita-se a ordem inicial – que retata a intuição e “bom senso” do processo –, porém explorando trocas locais que podem gerar melhores soluções, dado o caráter combinatório do problema.

Este último tipo de procedimento requer dois parâmetros: o viés da escolha do próximo item; e os critérios de parada de seu funcionamento.

3.3.3.1 VIÉS DA ESCOLHA

As heurísticas construtivas probabilísticas pressupõe como parâmetro o viés atribuído aos itens candidatos, de acordo com a ordenação inicial proposta. Serão explorados três tipos de viés:

- **Sem viés** (aleatório): todos os itens da fila apresentam igual probabilidade de serem sorteados, de forma que a escolha passa a ser ao acaso, independente da ordenação inicial. A probabilidade de escolha de um item que ocupa a i -ésima posição em uma fila de n itens segue (3.1);

$$P(i) = \frac{1}{\max_{j \leq n} j} \quad (3.1)$$

- **Viés linear (ou proporcional)**: a probabilidade de um item ser sorteado cai linearmente de acordo com sua posição da fila inicial. A probabilidade de escolha de um item que ocupa a i -ésima posição em uma fila de n itens segue (3.2), cuja dedução se encontra no apêndice A;

$$P(i) = \frac{n - (i - 1)}{(n^2 - \sum_{i=1}^{n-1} i)} \quad (3.2)$$

- **Viés exponencial (ou desproporcional)**: acentuasse-se a probabilidade dos primeiros itens serem sorteados através de uma curva de probabilidade que diminui exponencialmente. Neste estudo, atribui-se arbitrariamente uma probabilidade de 50% para o primeiro item, probabilidade esta que cai pela metade para o seguinte e assim sucessivamente. A probabilidade de escolha de um item que ocupa a i -ésima posição em uma fila de n itens segue (3.3). Importante notar que a primeira parcela refere-se à probabilidade por tal regra. A segunda parcela é uma correção para o caso em que há poucos elementos, consistindo basicamente no rateio linear da probabilidade atribuída a itens que excedem o tamanho da fila. Por se tratar inicialmente de uma regra

exponencial, é possível o sorteio de itens com infinitas posições, teoricamente. Com tal correção, ajusta-se essa probabilidade para o tamanho da fila.

$$P(i) = 0,5^{i-1} + 0,5^{n-1} \frac{0,5^{i-1}}{1 - 0,5^{n-1}} \quad (3.3)$$

O gráfico 3.1 ilustra comparativamente a probabilidade de sorteio em uma fila com 10 itens.

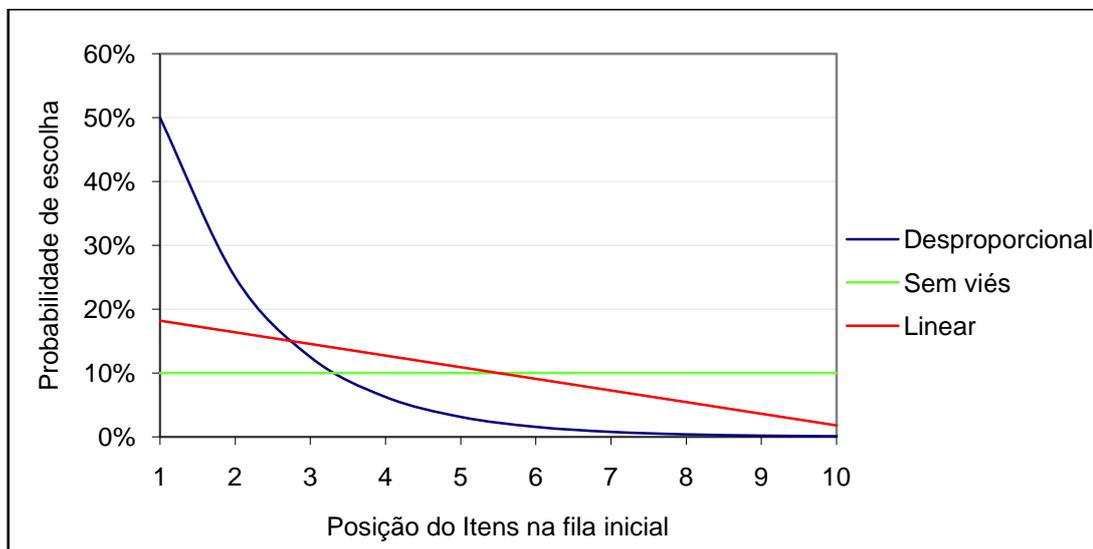


Gráfico 3.1 – Probabilidade de sorteio de uma ordem em cada tipo de viés de escolha

3.3.3.2 CRITÉRIOS DE PARADA

Outro pressuposto do procedimento utilizando heurísticas construtivas probabilísticas é a existência de critérios de parada. Para o modelo em questão, as heurísticas construtivas probabilísticas foram limitadas por três parâmetros.

- **Número máximo de iterações:** 500 iterações;
- **Número máximo de iterações sem ganho:** 100 iterações;
- **Tempo máximo de processamento:** 60 segundos.

Para determinar os valores de tais parâmetros, foi feito um estudo inicial, tomando o procedimento de formação de padrão de corte aleatoriamente (probabilístico sem viés). Foram registrados os desempenho de ocupação e de atendimento de datas do melhor padrão de corte, conforme o critério que será apresentado na subseção 3.4, utilizando respectivamente peso de 100% para ocupação e 100% para atendimento de datas em experimentos separados.

Como aqui só se almejou verificar o desempenho da heurística em função dos critérios de parada, foi considerada uma amostra de 100 itens de 1 único agrupamento técnico e 1 único objeto de dimensão 100 x 100. As dimensões dos itens foram geradas uniformemente entre 15 e 25 ($U[15,25]$) e a folgas dinâmicas uniformemente entre -15 e 15 ($U[-15,15]$). O problema bidimensional guilhotinado foi escolhido arbitrariamente para o teste. Os resultados apresentados são as médias de 5 experimentos.

O gráfico 3.2 mostra o desempenho do melhor padrão de corte gerado em função do **número máximo de iterações**. Os resultados obtidos mostram que os ganhos são muito expressivos até poucos menos de 200 iterações como limitante; há pequenos ganhos até um máximo de 400; e ganhos muito incrementais a partir deste patamar. Por isso, foi estipulado como parâmetro o máximo de 500 iterações, colocando uma margem de segurança.

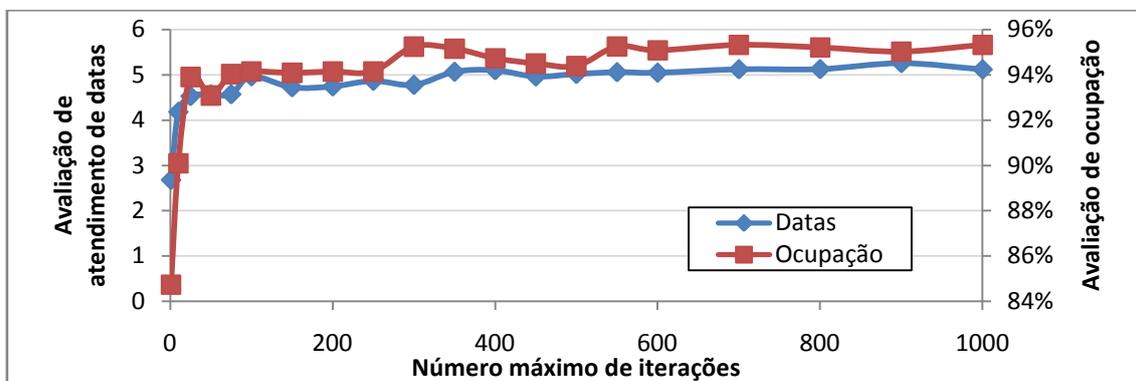


Gráfico 3.2 – Desempenho da heurística probabilística em diferentes cenários do número máximo de iterações

Analogamente, o gráfico 3.3 mostra o desempenho do melhor padrão de corte gerado em função do **número máximo de iterações sem ganho**. Os resultados obtidos mostram que os ganhos são relevantes até poucos menos de 100 iterações sem ganho de qualidade na solução; acima deste valor, os ganhos são pequenos ou menos não foram registrados. Foi tomado, portanto, o limitante de 100 iterações sem ganho como critério de parada.

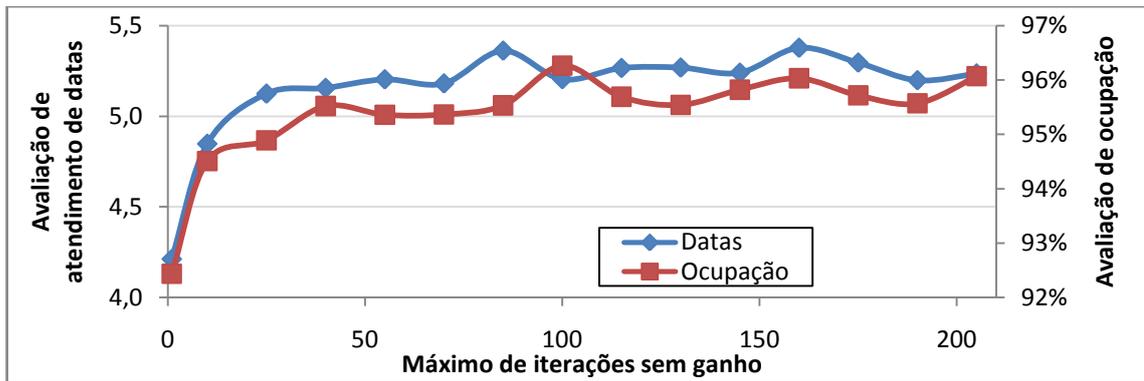


Gráfico 3.3 – Desempenho da heurística probabilística em diferentes cenários do número máximo de iterações sem ganho

Por fim, o gráfico 3.4 mostra o desempenho do melhor padrão de corte gerado em função do **tempo máximo de processamento**. Os resultados obtidos mostram que os ganhos são muito expressivos até ao redor de 50 segundos de tempo máximo; ao passo que a partir deste limitante os ganhos são incrementais. Foi tomado, portanto, o tempo de 60 segundos, acrescentando uma margem de segurança.

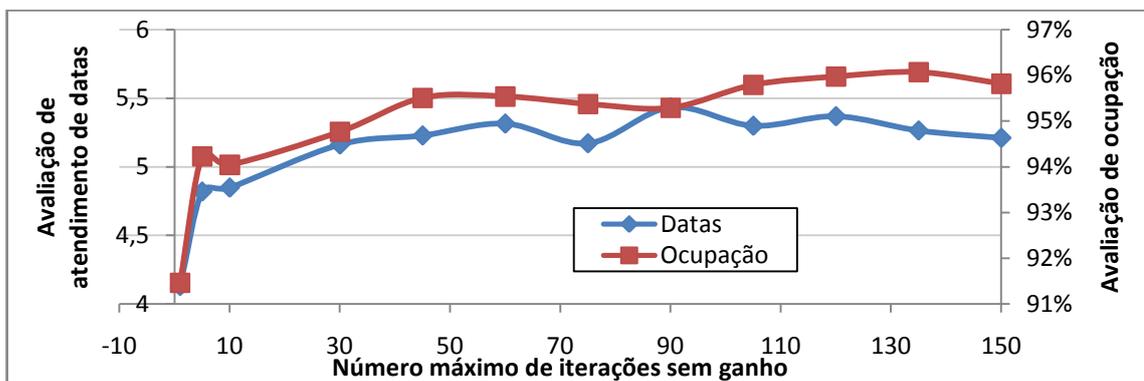


Gráfico 3.4 – Desempenho da heurística probabilística em diferentes cenários do número máximo de iterações

3.3.3.3 PROCEDIMENTO DE CORTE E EMPACOTAMENTO

Em pose do procedimento heurístico de escolha do próximo item, da ordenação inicial e dos critérios de parada, finalmente é possível discutir o algoritmo de formação dos padrões

de corte. Tais processos avaliam o encaixe dos itens e a disponibilidade restante do objeto para ocupá-lo de uma forma suficientemente boa, tendo em conta que se trata de um processo construtivo.

Importante ressaltar que ao contrário de procedimentos otimizantes ou mesmo de busca, os algoritmos abaixo descritos impõem a restrição de não retirar um item já posicionado, avaliando apenas como encaixá-lo de maneira a melhorar a disponibilidade para os itens restantes.

3.3.3.3.1 UNIDIMENSIONAL

O procedimento de corte unidimensional previsto no modelo tem como dados de entrada um conjunto de itens candidatos e um objeto, ambos pertencentes a um critério de agrupamento. Aqui, a restrição de corte não é pertinente (todo corte é guilhotinado, obrigatoriamente), assim como a questão da orientação dos itens no objeto.

O seu pseudo-código pode ser descrito pelos seguintes passos:

1. Ordenar itens segundo critério inicial escolhido (conforme o item 3.3.2);
2. Eliminar itens cujo comprimento é maior que o restante do objeto (na primeira iteração, trata-se do objeto todo);
3. Escolher o próximo item (se o procedimento geral for determinístico, é o primeiro da fila; no caso probabilístico, faz-se o sorteio e o item é escolhido conforme sua probabilidade, descrita no item 3.3.3.1);
4. Posicionar o item no objeto alinhado com o último item posicionado (se for o primeiro, com o início do objeto);
5. Atualizar o tamanho do objeto, subtraindo a dimensão do item posicionado;
6. Repetir 2 a 5 enquanto houver itens candidatos em fila.

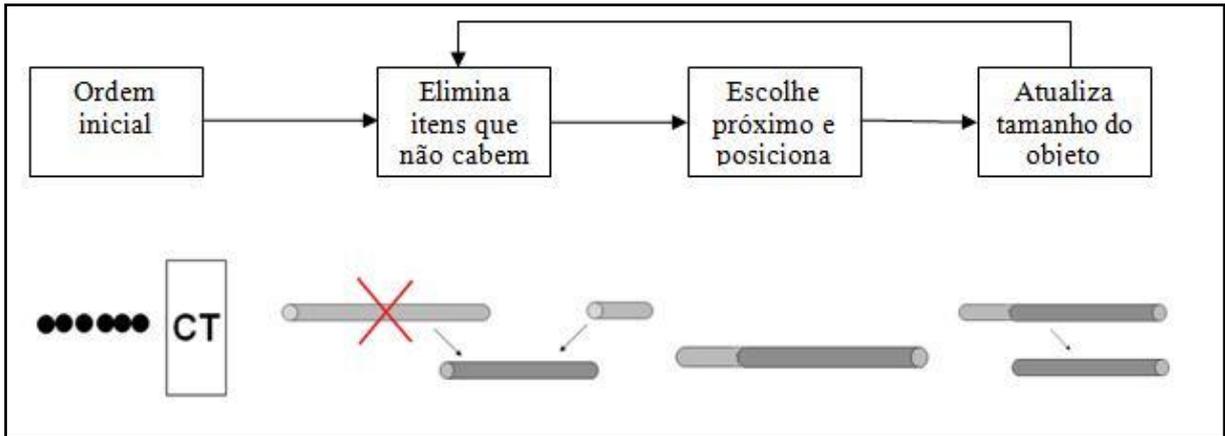


Figura 3.3 – Ilustração do procedimento unidimensional de corte

3.3.3.3.2 BIDIMENSIONAL GUILHOTINADO

No caso bidimensional guilhotinado, os dados de entrada são essencialmente os mesmos: as dimensões de um objeto e a lista de itens candidatos, ambos de um mesmo critério de agrupamento. Neste caso, ao contrário do unidimensional, o atributo de orientação dos itens apresenta relevância.

Nesta rotina, em específico, surge uma nova entidade, denominada **subobjeto**. Subobjetos são as partes restantes do corte guilhotinado depois que uma peça é posicionada. Cada corte pode gerar até 2 subobjetos se as dimensões do item posicionado são inferiores ao subobjeto inicial. No início da rotina, considera-se que há apenas 1 subobjeto, cujas dimensões coincidem com as do objeto original.

Partindo do pressuposto mencionado na seção 3.3.1.4 de que o item a ser posicionado já é definido, a heurística acrescenta duas escolhas arbitrárias:

- a. O subobjeto em que o item será posicionado: Lodi, Martello e Vigo (2002) propõem como heurística construtiva para o corte bidimensional guilhotinado a FBF (*finite best fit*), em que os itens são ordenados por critério decrescente de altura e posicionados em diferentes estágios (“prateleiras”) abertas anteriormente, de forma a minimizar a largura restante. Tal princípio foi adaptado aqui para uma situação de múltiplos estágios, escolhendo dentre a lista de subobjetos aquele em que haverá menor sobra em alguma das dimensões. Em caso de empate, dá-se a

preferência ao menor subobjeto, por considerá-lo menos flexível. Na figura 3.4 tal critério fica ilustrado;

- b. O sentido do corte decorrente de tal posicionamento: é possível gerar dois tipos de corte, conforme ilustrado na figura 3.4. É escolhido aquele que gera os dois novos subobjetos mais desiguais possíveis. Parte-se do princípio aqui que um dos subobjetos preserva maior flexibilidade em detrimento do outro, porém evitando que eventualmente ambos os gerados sejam pouco úteis.

O seu pseudo-código pode ser descrito pelos seguintes passos:

1. Ordenar itens segundo critério inicial escolhido (conforme o item 3.3.2);
2. Eliminar itens que não cabem em nenhum dos subobjetos disponíveis (na primeira iteração, trata-se do objeto todo);
3. Escolher o próximo item (se o procedimento geral for determinístico, é o primeiro da fila; no caso probabilístico, faz-se o sorteio e o item é escolhido conforme sua probabilidade, descrita no item 3.3.3.1);
4. Posicionar o item no canto inferior esquerdo do subobjeto em que houver menor diferença entre uma das dimensões do item e correspondente do subobjeto. Em caso de empate, optar pelo subobjeto de menor área;
5. Atualizar a lista de subobjetos:
 - a. Gerar novos subobjetos: das duas possibilidades de corte, escolher a que gera os subobjetos mais desiguais;
 - b. Eliminar subobjetos que não comportem nenhum dos itens restantes em fila.
6. Repetir 2 a 5 enquanto houver itens candidatos em fila.

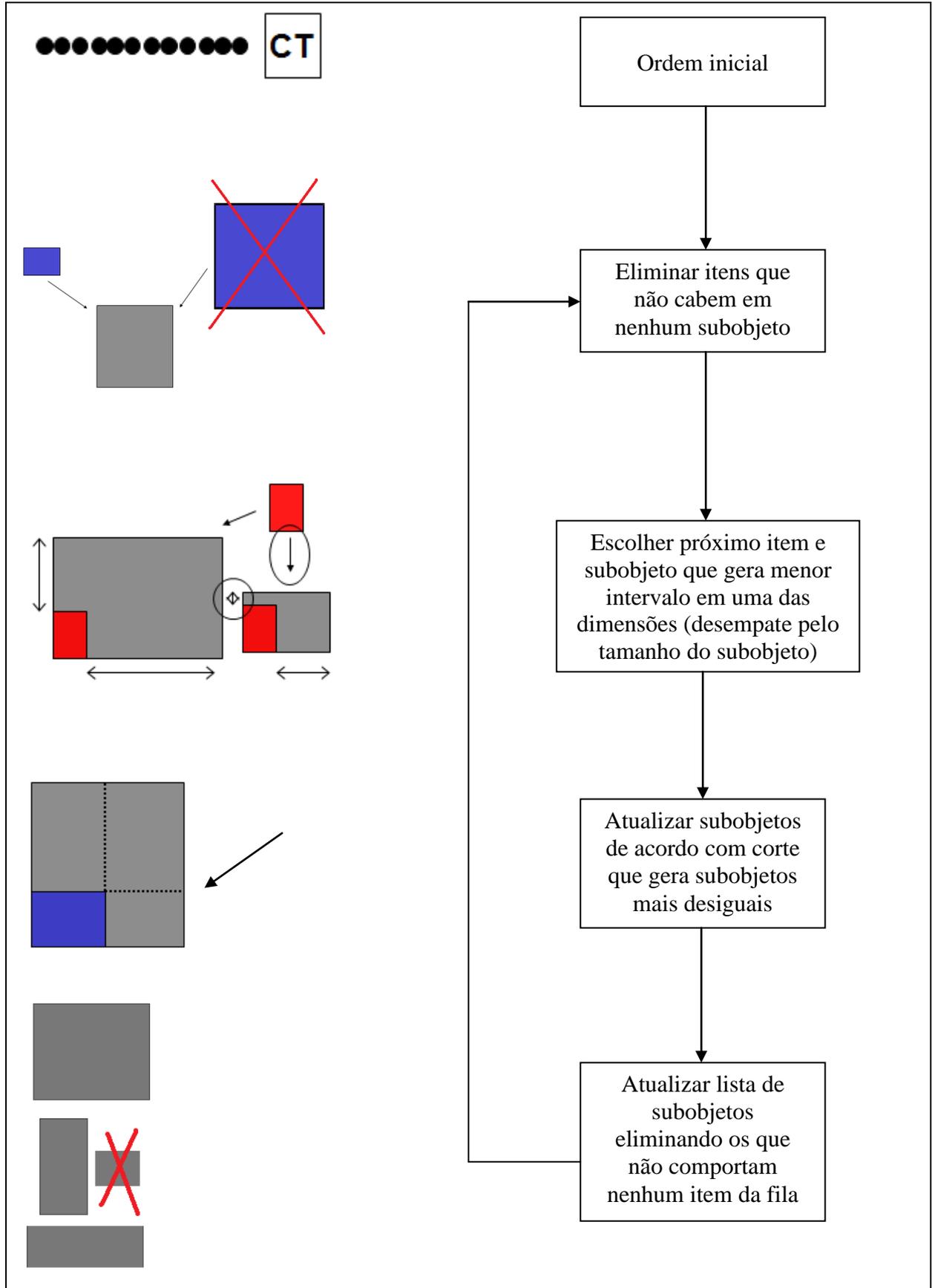


Figura 3.4 – Ilustração do procedimento bidimensional guilhotinado de corte

3.3.3.3.3 BIDIMENSIONAL NÃO-GUILHOTINADO

No caso bidimensional não-guilhotinado, novamente os dados de entrada são as dimensões de um objeto e a lista de itens candidatos, ambos de um mesmo critério de agrupamento. A orientação dos itens tem relevância assim como no caso guilhotinado.

A heurística de empacotamento proposta é uma adaptação do procedimento adotado por Wu et al. (2002). O autor propõe o princípio que chama de *Less Flexible First* (LFF), ou menos flexível primeiro. A ideia central é de que os cantos são mais restritivos que as arestas, que por sua vez são mais restritivas que os espaços vazios. Assim, os objetos seriam posicionados nos cantos potenciais, tomando-se o canto de acordo com uma função valor associada à escolha.

No caso deste estudo, a escolha do canto está associada à porção das arestas do item posicionado que é adjacente a itens previamente posicionadas. Tal critério se justifica porque em itens que são posicionamento adjacientemente a itens maiores já colocados criam cantos voltados para espaços vazios, ou seja, mais flexíveis. Em caso de empate, optou-se pelo cantos cuja distância até o canto mais próximo seja menor, procurando evitar desperdícios.

O seu pseudo-código pode ser descrito pelos seguintes passos:

1. Ordenar itens segundo critério inicial escolhido (conforme o item 3.3.2);
2. Eliminar itens que não podem ser posicionados em nenhum dos cantos disponíveis (na primeira iteração, é equivalente a não caber no objeto);
3. Escolher o próximo item (se o procedimento geral for determinístico, é o primeiro da fila; no caso probabilístico, faz-se o sorteio e o item é escolhido conforme sua probabilidade, descrita no item 3.3.3.1);
4. Verificar cantos e orientações possíveis para alocação e posicionar o item no de maior valor. Aqui, valor é definido como porcentagem das arestas posicionadas no canto que são adjacentes a alguma peça previamente posicionada (ou aresta do objeto original). Em caso de empate, é escolhido o canto com menor intervalo até o canto mais próximo com alguma coordenada em comum;
5. Atualizar a lista de cantos;
6. Repetir 2 a 5 enquanto houver itens candidatos em fila.

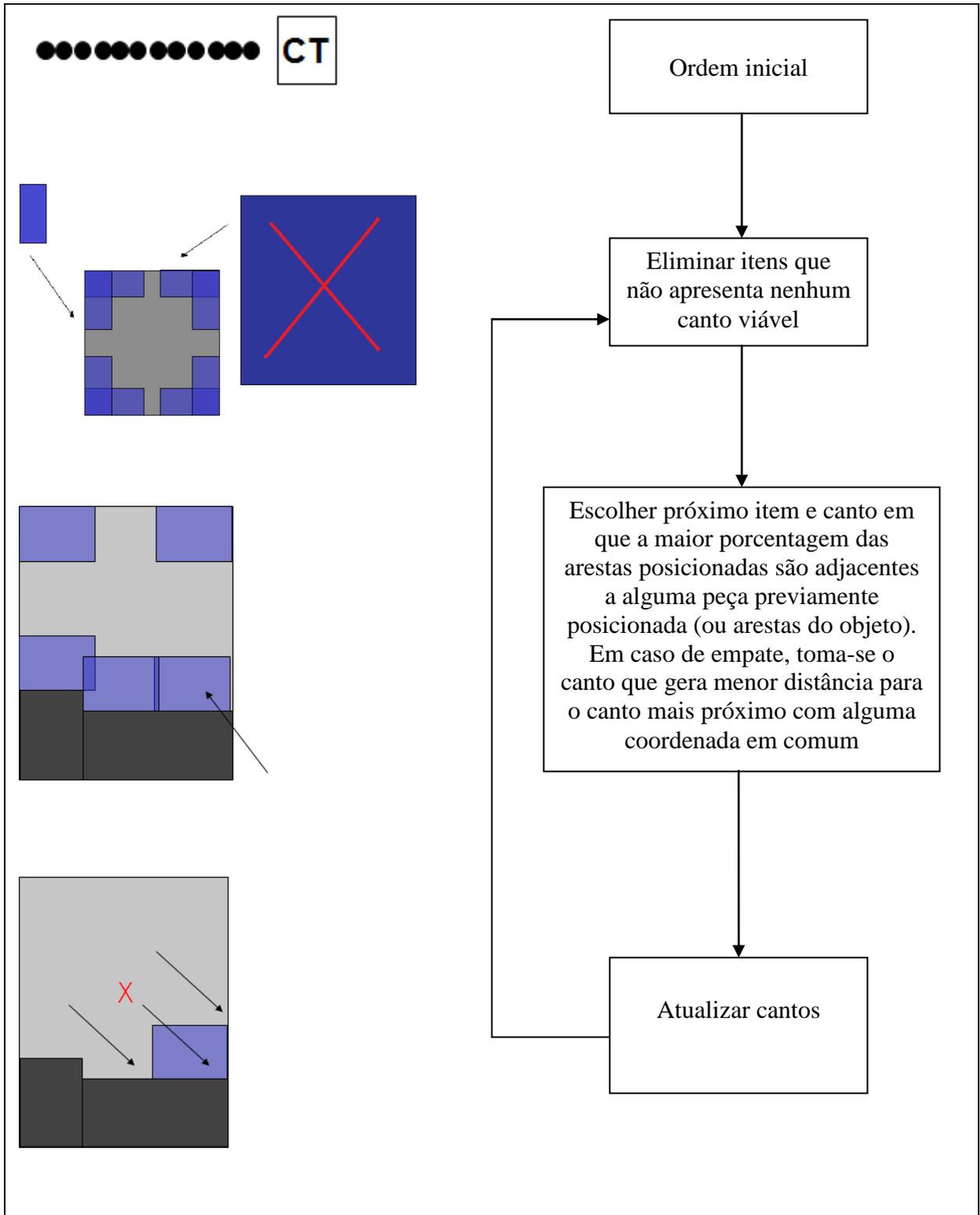


Figura 3.5 – Ilustração do procedimento bidimensional não-guilhotinado de corte

3.3.4 PROCEDIMENTO DE AVALIAÇÃO DE PADRÕES

O procedimento de avaliação dos padrões de corte, em consonância com a proposta maior deste trabalho, contempla os dois objetivos principais: ocupação e atendimento de datas.

A **avaliação da ocupação** é facilmente definida: o quociente entre a somatória das áreas ocupadas pelos itens pela área do objeto já define fisicamente a ocupação do objeto, relação esta formalizada em (3.4).

$$\text{Valor de ocupação do objeto} = \frac{\sum_{i=\text{itens posicionados}} \text{Área}_i}{\text{Área do objeto}} \quad (3.4)$$

Por outro lado, a **avaliação** da satisfação do critério de **atendimento de datas** apresenta alguma complicação adicional. O primeiro ponto é que a folga dinâmica – grandeza utilizada para medir a urgência do item pela organização – pode assumir valores positivos e negativos. Assim, a somatória das folgas não é um valor que reflete o valor de um padrão de corte. Em segundo lugar, há que se ter em mente que as folgas negativas são mais do que proporcionalmente desejáveis do que as positivas. Isto porque as primeiras representam atrasos locais que precisam ser encaminhados, enquanto estas últimas são potenciais geradoras de estoque em processo.

Partindo destas observações, formulou-se um indicador, aqui denominado de “folga dinâmica corrigida”, mostrado em (3.5). O valor de atendimento de datas de um padrão de corte é a somatória deste indicador ponderada por um peso da posição da ordem dentro do padrão, conforme (3.7). Tal peso, cujo valor segue (3.6) e está retratado no gráfico 3.5, serve para fazer com que o valor de atendimento de datas de um padrão de corte seja representado preponderantemente por seus itens mais atrasados.

$$\text{Folga dinâmica corrigida} = \begin{cases} - \text{Folga dinâmica} & , \text{ se } \text{Folga dinâmica} < 0 \\ \frac{1}{(2 + \text{Folga dinâmica})} & , \text{ se } \text{Folga dinâmica} \geq 0 \end{cases} \quad (3.5)$$

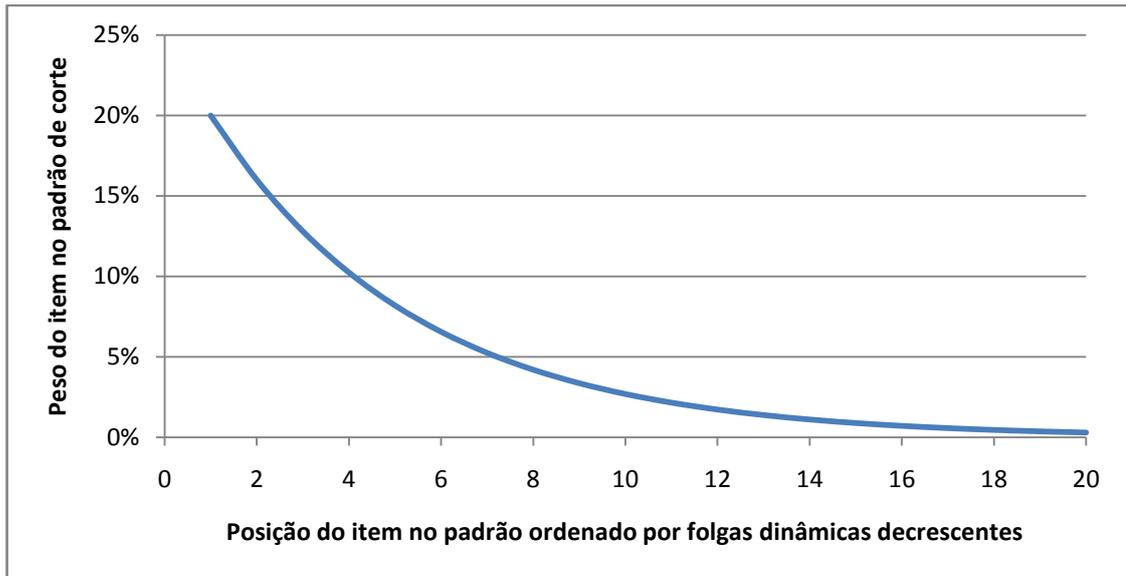


Gráfico 3.5 – Pesos pela posição do item no padrão de corte em ordem decrescente de folga dinâmica

$$Peso_{i-\text{ésimo item do padrão}} = 0,2 \times 0,8^{(i-1)} \quad (3.6)$$

$$Valor\ de\ atendimento\ de\ datas = Peso_i \times \sum_i Folga\ dinâmica\ corrigida_i \quad (3.7)$$

3.3.4.1 PONDERAÇÃO E NORMALIZAÇÃO DAS AVALIAÇÕES

Ao longo do procedimento descrito, há várias comparações entre padrões:

- Vários padrões gerados probabilisticamente para um mesmo objeto;
- Vários padrões de diferentes objetos para um mesmo conjunto de itens de um critério de agrupamento;
- Vários padrões de diferentes agrupamentos a serem sequenciados.

Fica notório em (3.4) e (3.7) que a faixa de valores abrangida nos diferentes critérios é muito distinta. Enquanto o valor de ocupação é uma faixa contínua entre 0 e 1, o valor de atendimento de datas é uma faixa contínua de 0 a $+\infty$. Assim, recorre-se a um procedimento de normalização dos valores quando de sua comparação entre padrões. A relação (3.8) mostra

o critério adotado, que é a proporção em base percentual com relação aos extremos da série comparada.

$$Valor\ normalizado_i = \frac{Valor\ calculado_i - \min_i(Valor\ calculado_i)}{\max_i(Valor\ calculado_i) - \min_i(Valor\ calculado_i)} \quad (3.8)$$

Por fim, havendo uma mesma base de pontuação, a ponderação dos valores é feita atribuindo pesos a cada um dos objetivos, conforme (3.9).

$$Valor\ do\ padrão = Peso_{ocup} \times Valor_{ocup} + Peso_{at.datas} \times Valor_{at.datas} \quad (3.9)$$

3.4 RESUMO

Nesta seção, foram detalhados os métodos propostas para cada tipo de problema. Para orientar o acompanhamento dos resultados, são mostrados a seguir, de maneira esquemática, os problemas e procedimentos de solução a serem analisados.

- i. Problemas
 - A. Unidimensional
 - B. Bidimensional guilhotinado
 - C. Bidimensional não-guilhotinado

- ii. Procedimentos
 - A. Determinístico

Heurística 1. Ordenação inicial por critério de datas (folga dinâmica)

Heurística 2. Ordenação inicial por critério de ocupação (dimensão)

Heurística 3. Ordenação inicial por critério ponderado

B. Probabilístico

Heurística 4. Sem viés (aleatório)

B1. Com viés proporcional (linear)

Heurística 5. Ordenação inicial por critério de datas (folga dinâmica)

Heurística 6. Ordenação inicial por critério de ocupação (dimensão)

Heurística 7. Ordenação inicial por critério ponderado

B3. Com viés desproporcional (exponencial)

Heurística 8. Ordenação inicial por critério de datas (folga dinâmica)

Heurística 9. Ordenação inicial por critério de ocupação (dimensão)

Heurística 10. Ordenação inicial por critério ponderado

4 RESULTADOS EXPERIMENTAIS

A seção 3.4 mostra que existem 3 diferentes problemas e 10 variantes do método heurístico proposto. Antes de proceder com as aplicações práticas, julgou-se necessário uma avaliação comparativa dos métodos tanto de sua robustez perante diferentes situações de atributos dos itens, quanto da importância de cada um dos critérios que compõem o método.

4.1 CENÁRIOS PROJETADOS

Foram identificados 5 fatores importantes no projeto de cenários para teste dos métodos heurísticos propostos, cada um com 2 a 3 níveis, além de um sexto fator que é o próprio método heurístico:

- a) **Dimensão média dos itens:** os níveis definidos quantitativamente como o centro da distribuição uniforme a ser utilizada para gerar aleatoriamente as dimensões dos itens. O nível alto corresponderá à distribuição com centro (que no caso uniforme equivale à média) de 30% da dimensão do objeto; enquanto o nível baixo será aquele com centro de 15% da dimensão do objeto;

Tabela 4.1 – Níveis do fator “Dimensão média dos itens”

<i>Dimensão média dos itens</i>		
<i>Denominação</i>	Dimensões menores (Dim Menor)	Dimensões maiores (Dim Maior)
<i>Valor quantitativo</i>	Centro da distribuição = 15% da dimensão do objeto	Centro da distribuição = 30% da dimensão do objeto

- b) **Variabilidade das dimensões dos itens:** os níveis definidos quantitativamente como a amplitude da distribuição uniforme a ser utilizadas para gerar aleatoriamente as dimensões dos itens. O nível alto corresponderá à distribuição

com amplitude de 20% da dimensão do objeto; enquanto o nível baixo será aquele com amplitude de 10% da dimensão do objeto;

Tabela 4.2 – Níveis do fator “Variabilidade das dimensões dos itens”

<i>Variabilidade das dimensões dos itens</i>		
<i>Denominação</i>	Dimensões homogêneas (Dim Ho)	Dimensões heterogêneas (Dim He)
<i>Valor quantitativo</i>	Amplitude da distribuição = 10% da dimensão do objeto	Amplitude da distribuição = 20% da dimensão do objeto

- c) **Folga dinâmica média dos itens:** os níveis definidos quantitativamente como o centro da distribuição uniforme a ser utilizadas para gerar aleatoriamente as folgas dinâmicas dos itens. O nível alto corresponderá à distribuição com centro de 0 de folga dinâmica (em média, as ordens estão “em dia”); enquanto o nível baixo será aquele com centro de -10 de folga dinâmica (em média, as ordens estão atrasadas);

Tabela 4.3 – Níveis do fator “Folga dinâmica média dos itens”

<i>Folga dinâmica média dos itens</i>		
<i>Denominação</i>	Folgas dinâmicas menores ou atrasadas (Fdin menor)	Folgas dinâmicas maiores ou “em dia” (Fdin maior)
<i>Valor quantitativo</i>	Centro da distribuição = -10	Centro da distribuição = 0

- d) **Variabilidade da folga dinâmica dos itens:** os níveis definidos quantitativamente como a amplitude da distribuição uniforme a ser utilizadas para gerar aleatoriamente as folgas dinâmicas dos itens. O nível alto corresponderá à distribuição com amplitude de 30 unidades; enquanto o nível baixo será aquele com amplitude de 10 unidades;

Tabela 4.4 – Níveis do fator “Variabilidade da folga dinâmica dos itens”

<i>Variabilidade da folga dinâmica dos itens</i>		
<i>Denominação</i>	Folgas dinâmicas homogêneas (Fdin Ho)	Folgas dinâmicas heteorgêneas (Fdin He)
<i>Valor quantitativo</i>	Amplitude da distribuição = 10	Amplitude da distribuição = 30

- e) **Peso dos critérios** (ocupação e atendimento de datas): aqui serão definidos 3 níveis, um deles com prioridade maior para atendimento de datas, que refletirá em peso de 80% para tal critério (e, conseqüentemente, 20% para ocupação); outro com prioridade maior para ocupação (80% para ocupação e 20% para atendimento de datas); e, finalmente, prioridades iguais (ou seja, 50% para cada critério).

Tabela 4.5 – Níveis do fator “Peso dos critérios”

<i>Peso dos critérios</i>			
<i>Denominação</i>	Prioridade para ocupação	Prioridades iguais	Prioridade para atendimento de datas
<i>Valor quantitativo</i>	$w_{\text{ocupação}} = 20\%$; $w_{\text{atendimento de datas}} = 80\%$	$w_{\text{ocupação}} = 50\%$; $w_{\text{atendimento de datas}} = 50\%$	$w_{\text{ocupação}} = 80\%$; $w_{\text{atendimento de datas}} = 20\%$

- f) **Métodos heurísticos**: são 10 níveis, de acordo com os métodos explicados na seção 3 e resumidos no item 3.3. São definidas pela combinação de um procedimento geral (determinístico ou probabilístico); uma ordem inicial da fila de itens (critério de ocupação, critério de atendimento de datas e critério normalizado e ponderado); e um viés na escolha do próximo item (sem viés, linear e exponencial).

Tabela 4.6 – Níveis do fator “Métodos heurísticos”

<i>Métodos heurísticos</i>			
<i>Denominação</i>	<i>Procedimento geral</i>	<i>Ordem inicial</i>	<i>Viés de escolha</i>
1	Determinístico	Critério de atendimento de datas	-
2	Determinístico	Critério de ocupação	-
3	Determinístico	Critério de ponderado	-
4	Probabilístico	Indiferente	Sem viés (aleatório)
5	Probabilístico	Critério de atendimento de datas	Linear
6	Probabilístico	Critério de ocupação	Linear
7	Probabilístico	Critério de ponderado	Linear
8	Probabilístico	Critério de atendimento de datas	Exponencial
9	Probabilístico	Critério de ocupação	Exponencial
10	Probabilístico	Critério de ponderado	Exponencial

Assim, configuram-se 480 combinações de experimentos a serem testadas, considerando as 10 variantes do procedimento heurístico proposto, sendo 48 situações para cada uma delas. A tabela 4.7 ilustra tais situações experimentais, indicando os extremos das distribuições uniformes que serão utilizadas para gerar as folgas dinâmicas dos itens, bem como suas respectivas dimensões.

4.2 RESULTADOS

Os experimentos foram rodados em paralelos nos 26 computadores do laboratório de informática do Departamento de Engenharia de Produção da Escola Politécnica. São computadores com memória RAM de 512 MB e processadores Pentium 4 de 1.6 ou 1.8 GHz (com três exceções, dois Celerons de 1.0 GHz e 1.1GHz, e um Pentium 3 de 1.0 GHz). Foram feitas 5 réplicas de cada um dos experimentos nos três tipos de problemas propostos, para cada uma das 10 heurísticas, totalizando um conjunto de 7200 experimentos.

Tabela 4.7 – Situações experimentais para cada método heurístico

Experimento	FDin Min	FDin Max	Dim Min	Dim Max	Wocupação	W _{at. data}
1	-5	5	5	25	20%	80%
2	-5	5	10	20	20%	80%
3	-5	5	25	35	20%	80%
4	-5	5	20	40	20%	80%
5	-15	15	5	25	20%	80%
6	-15	15	10	20	20%	80%
7	-15	15	25	35	20%	80%
8	-15	15	20	40	20%	80%
9	-15	-5	5	25	20%	80%
10	-15	-5	10	20	20%	80%
11	-15	-5	25	35	20%	80%
12	-15	-5	20	40	20%	80%
13	-25	5	5	25	20%	80%
14	-25	5	10	20	20%	80%
15	-25	5	25	35	20%	80%
16	-25	5	20	40	20%	80%
17	-5	5	5	25	50%	50%
18	-5	5	10	20	50%	50%
19	-5	5	25	35	50%	50%
20	-5	5	20	40	50%	50%
21	-15	15	5	25	50%	50%
23	-15	15	10	20	50%	50%
23	-15	15	25	35	50%	50%
24	-15	15	20	40	50%	50%
25	-15	-5	5	25	50%	50%
26	-15	-5	10	20	50%	50%
27	-15	-5	25	35	50%	50%
28	-15	-5	20	40	50%	50%
29	-25	5	5	25	50%	50%
30	-25	5	10	20	50%	50%
31	-25	5	25	35	50%	50%
32	-25	5	20	40	50%	50%
33	-5	5	5	25	80%	20%
34	-5	5	10	20	80%	20%
35	-5	5	25	35	80%	20%
36	-5	5	20	40	80%	20%
37	-15	15	5	25	80%	20%
38	-15	15	10	20	80%	20%
39	-15	15	25	35	80%	20%
40	-15	15	20	40	80%	20%
41	-15	-5	5	25	80%	20%
42	-15	-5	10	20	80%	20%
43	-15	-5	25	35	80%	20%
44	-15	-5	20	40	80%	20%
45	-25	5	5	25	80%	20%
46	-25	5	10	20	80%	20%
47	-25	5	25	35	80%	20%
48	-25	5	20	40	80%	20%

Em cada rodada, obteve-se como saída uma lista de padrões de corte a serem executados e sua respectiva ordem de corte, bem como o valor de ocupação ($V_{\text{ocupação}}$) e de atendimento de datas ($V_{\text{at.datas}}$) para cada um destes padrões, ambos descritos na seção 3.3.4. Para fins de avaliação do desempenho da heurística, foi considerada uma média ponderada dos 4 primeiros padrões na fila, já que para o caso prático só são relevantes os primeiros padrões (aqueles que serão feitos antes de novas alterações na fila, como por exemplo a chegada de uma nova ordem). O cálculo do valor (V) de cada critério em função dos valores dos 4 primeiros padrões de corte propostos (V_1, V_2, V_3, V_4) é dado por (4.1), em que mostra-se claramente pelos pesos que é dada prioridade aos primeiros padrões.

$$V = 60\% \times V_1 + 25\% \times V_2 + 10\% \times V_3 + 5\% \times V_4 \quad (4.1)$$

Os valores de ocupação e de atendimento de cada experimento foram considerados de acordo com a média dos valores calculados conforme (4.1) obtidos nas 5 réplicas. Um terceiro valor encontrado – utilizado com finalidade de decisão – é a normalização e ponderação de ambos, conforme os pesos do respectivo experimento. A normalização ocorre numa escala de 0 a 1 através da comparação com os resultados das demais heurísticas para as situações em que os outros cinco fatores são idênticos, conforme (4.2).

$$V_{\text{ponderado}} = w_{\text{ocupação}} \times \left(\frac{V_{\text{ocupação}} - V_{\text{ocupação}}^{\text{mínimo}}}{V_{\text{ocupação}}^{\text{máximo}} - V_{\text{ocupação}}^{\text{mínimo}}} \right) + w_{\text{at.datas}} \times \left(\frac{V_{\text{at.datas}} - V_{\text{at.datas}}^{\text{mínimo}}}{V_{\text{at.datas}}^{\text{máximo}} - V_{\text{at.datas}}^{\text{mínimo}}} \right) \quad (4.2)$$

Os resultados obtidos estão elencados em tabelas como 4.8 a 4.16 (resultados apenas do problema unidimensional, com os diferentes níveis de pesos dos critérios separados em cada tabela) mostradas a seguir. Como a amplitude de situações abordadas gerou uma massa de dados muito grande, as tabelas de resultados dos outros dois problemas encontram-se na íntegra no apêndice B.

Tabela 4.8 – Resultados do critério de ocupação para o problema unidimensional com peso de ocupação de 20% e peso de atendimento de datas de 80%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	97,62%	98,58%	95,17%	94,75%
	2	98,98%	99,73%	98,51%	99,03%
	3	97,87%	99,05%	98,34%	98,84%
	4	99,66%	99,81%	97,78%	99,60%
	5	99,78%	99,77%	98,08%	99,21%
	6	99,67%	99,87%	98,63%	99,58%
	7	99,76%	99,78%	98,15%	99,38%
	8	98,24%	99,25%	93,55%	96,96%
	9	98,27%	98,76%	98,03%	98,24%
	10	98,53%	99,37%	98,43%	98,49%
<i>Fdin menor</i>	1	98,25%	98,14%	93,83%	97,65%
	2	98,05%	99,41%	97,66%	99,37%
	3	97,91%	98,64%	97,75%	98,59%
	4	99,46%	99,75%	98,55%	98,72%
	5	99,58%	99,81%	97,19%	98,90%
	6	99,57%	99,87%	98,90%	98,90%
	7	99,37%	99,73%	97,17%	98,68%
	8	97,39%	98,66%	94,66%	94,58%
	9	98,18%	99,55%	97,97%	98,38%
	10	98,45%	98,50%	98,43%	98,74%
<i>Fdin Ho</i>	1	96,36%	97,75%	93,44%	92,72%
	2	97,51%	99,48%	98,57%	99,10%
	3	96,73%	99,22%	94,63%	97,68%
	4	99,48%	99,67%	97,06%	98,99%
	5	99,77%	99,75%	96,88%	99,04%
	6	99,67%	99,67%	98,35%	99,11%
	7	99,63%	99,83%	98,20%	99,34%
	8	96,59%	98,72%	93,08%	94,87%
	9	97,27%	99,61%	96,21%	99,02%
	10	97,37%	98,01%	96,33%	96,90%
<i>Fdin maior</i>	1	96,76%	97,20%	93,49%	96,22%
	2	98,04%	99,67%	98,62%	98,90%
	3	97,80%	98,37%	95,21%	98,23%
	4	99,69%	99,80%	96,38%	98,28%
	5	99,48%	99,62%	96,80%	97,82%
	6	99,69%	99,66%	98,02%	98,86%
	7	99,60%	99,83%	96,93%	99,15%
	8	97,10%	97,69%	92,23%	96,00%
	9	98,39%	99,24%	95,52%	99,29%
	10	98,05%	99,02%	95,47%	96,02%

Tabela 4.9 – Resultados do critério de ocupação para o problema unidimensional com peso de ocupação de 50% e peso de atendimento de datas de 50%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	99,14%	99,04%	97,33%	98,48%
	2	99,07%	99,72%	99,87%	99,68%
	3	99,39%	99,66%	99,40%	99,43%
	4	99,82%	99,91%	98,76%	99,67%
	5	99,81%	99,84%	98,82%	99,73%
	6	99,88%	99,95%	99,74%	99,73%
	7	99,93%	99,91%	99,72%	99,81%
	8	98,53%	98,92%	97,85%	98,78%
	9	99,52%	99,77%	99,78%	99,73%
	10	99,60%	99,48%	99,36%	99,34%
<i>Fdin menor</i>	1	98,07%	99,03%	97,58%	98,56%
	2	99,18%	99,75%	99,76%	99,67%
	3	99,38%	99,43%	99,48%	99,55%
	4	99,87%	99,87%	99,13%	99,45%
	5	99,85%	99,91%	98,59%	99,76%
	6	99,91%	99,94%	99,49%	99,56%
	7	99,90%	99,91%	99,52%	99,69%
	8	98,47%	98,85%	95,94%	98,65%
	9	99,41%	99,84%	99,82%	99,91%
	10	99,34%	99,64%	99,48%	99,46%
<i>Fdin Ho</i>	1	97,42%	98,88%	94,43%	97,71%
	2	99,53%	99,74%	99,46%	99,72%
	3	99,32%	99,11%	99,09%	99,53%
	4	99,78%	99,87%	98,33%	99,56%
	5	99,92%	99,94%	98,97%	99,76%
	6	99,88%	99,83%	99,26%	99,75%
	7	99,89%	99,89%	99,25%	99,54%
	8	97,77%	98,58%	97,03%	97,09%
	9	99,75%	99,80%	99,57%	99,70%
	10	99,10%	99,05%	98,04%	98,75%
<i>Fdin maior</i>	1	99,02%	98,65%	95,57%	98,08%
	2	99,52%	99,78%	99,81%	99,60%
	3	98,70%	98,75%	99,22%	99,38%
	4	99,74%	99,78%	98,92%	99,43%
	5	99,85%	99,87%	98,43%	99,58%
	6	99,87%	99,88%	99,44%	99,52%
	7	99,86%	99,88%	99,59%	99,45%
	8	97,18%	98,64%	95,55%	96,89%
	9	99,78%	99,81%	99,48%	99,85%
	10	99,15%	99,08%	98,96%	99,27%

Tabela 4.10 – Resultados do critério de ocupação para o problema unidimensional com peso de ocupação de 80% e peso de atendimento de datas de 20%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	99,31%	99,70%	98,61%	99,19%
	2	99,90%	99,90%	99,92%	99,82%
	3	99,74%	99,80%	99,77%	99,80%
	4	99,93%	99,96%	99,71%	99,91%
	5	99,96%	99,90%	99,55%	99,87%
	6	99,96%	99,96%	99,78%	99,89%
	7	99,96%	99,96%	99,84%	99,89%
	8	99,41%	99,71%	98,99%	99,58%
	9	99,89%	99,88%	99,93%	99,88%
	10	99,72%	99,81%	99,79%	99,87%
<i>Fdin menor</i>	1	99,22%	99,64%	98,68%	99,23%
	2	99,86%	99,90%	99,92%	99,91%
	3	99,75%	99,69%	99,82%	99,81%
	4	99,92%	99,96%	99,60%	99,88%
	5	99,94%	99,97%	99,59%	99,90%
	6	99,93%	99,97%	99,93%	99,88%
	7	99,94%	99,94%	99,84%	99,86%
	8	99,31%	99,54%	97,71%	99,37%
	9	99,76%	99,93%	99,90%	99,89%
	10	99,79%	99,80%	99,75%	99,79%
<i>Fdin Ho</i>	1	98,92%	99,31%	97,05%	98,58%
	2	99,86%	99,93%	99,83%	99,81%
	3	99,43%	99,71%	99,67%	99,64%
	4	99,93%	99,97%	99,65%	99,80%
	5	99,92%	99,94%	99,45%	99,83%
	6	99,92%	99,96%	99,75%	99,84%
	7	99,89%	99,96%	99,64%	99,90%
	8	99,22%	99,34%	98,34%	98,52%
	9	99,85%	99,92%	99,90%	99,84%
	10	99,56%	99,79%	99,67%	99,67%
<i>Fdin maior</i>	1	98,60%	99,29%	98,43%	98,50%
	2	99,76%	99,86%	99,86%	99,93%
	3	99,74%	99,69%	99,60%	99,78%
	4	99,93%	99,95%	99,70%	99,80%
	5	99,90%	99,96%	99,45%	99,88%
	6	99,95%	99,97%	99,69%	99,84%
	7	99,95%	99,97%	99,73%	99,84%
	8	98,96%	99,27%	98,16%	99,05%
	9	99,85%	99,87%	99,75%	99,89%
	10	99,54%	99,58%	99,74%	99,64%

Tabela 4.11 – Resultados do critério de atendimento de datas para o problema unidimensional com peso de ocupação de 20% e peso de atendimento de datas de 80%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	3,70	3,73	3,62	3,70
	2	3,69	3,70	3,57	3,65
	3	3,73	3,71	3,69	3,71
	4	3,72	3,74	3,64	3,69
	5	3,73	3,73	3,66	3,70
	6	3,71	3,72	3,66	3,67
	7	3,74	3,73	3,67	3,70
	8	3,64	3,71	3,70	3,70
	9	3,70	3,65	3,56	3,63
	10	3,73	3,73	3,70	3,71
<i>Fdin menor</i>	1	6,00	6,12	6,02	6,07
	2	6,08	6,00	5,73	5,73
	3	6,13	6,13	6,06	6,06
	4	6,10	6,11	5,86	6,04
	5	6,10	6,10	6,03	6,05
	6	6,08	6,12	5,85	5,99
	7	6,10	6,12	5,99	6,03
	8	6,01	6,14	6,05	6,03
	9	5,96	5,89	5,77	5,73
	10	6,15	6,12	6,07	6,07
<i>Fdin Ho</i>	1	1,21	1,22	1,23	1,22
	2	1,20	1,20	1,15	1,17
	3	1,21	1,22	1,22	1,22
	4	1,22	1,22	1,16	1,19
	5	1,23	1,24	1,21	1,21
	6	1,22	1,21	1,19	1,21
	7	1,24	1,23	1,18	1,20
	8	1,22	1,22	1,22	1,22
	9	1,19	1,21	1,12	1,16
	10	1,22	1,21	1,23	1,21
<i>Fdin maior</i>	1	3,62	3,65	3,57	3,64
	2	3,52	3,57	3,37	3,35
	3	3,64	3,67	3,58	3,61
	4	3,61	3,66	3,53	3,57
	5	3,63	3,61	3,48	3,49
	6	3,62	3,60	3,43	3,53
	7	3,66	3,63	3,52	3,56
	8	3,64	3,60	3,59	3,62
	9	3,40	3,49	3,27	3,43
	10	3,68	3,64	3,66	3,60

Tabela 4.12 – Resultados do critério de atendimento de datas para o problema unidimensional com peso de ocupação de 50% e peso de atendimento de datas de 50%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	3,57	3,72	3,40	3,59
	2	3,66	3,71	3,56	3,57
	3	3,70	3,67	3,68	3,68
	4	3,71	3,72	3,57	3,66
	5	3,72	3,72	3,61	3,69
	6	3,72	3,71	3,64	3,68
	7	3,72	3,72	3,66	3,68
	8	3,66	3,65	3,50	3,59
	9	3,68	3,67	3,55	3,63
	10	3,70	3,71	3,65	3,68
<i>Fdin menor</i>	1	5,83	5,96	5,37	5,83
	2	5,97	6,00	5,59	5,81
	3	5,92	6,06	5,99	5,98
	4	6,08	6,03	5,65	6,01
	5	6,09	6,17	5,85	5,94
	6	6,07	6,11	5,78	5,96
	7	6,13	6,09	5,79	6,00
	8	5,71	5,94	5,70	5,85
	9	5,85	5,95	5,59	5,86
	10	5,81	5,90	5,95	6,03
<i>Fdin Ho</i>	1	1,16	1,18	1,17	1,21
	2	1,20	1,19	1,06	1,12
	3	1,20	1,23	1,20	1,22
	4	1,20	1,23	1,15	1,19
	5	1,24	1,23	1,18	1,20
	6	1,21	1,21	1,15	1,18
	7	1,22	1,23	1,15	1,20
	8	1,19	1,22	1,09	1,14
	9	1,14	1,19	1,12	1,15
	10	1,22	1,21	1,22	1,21
<i>Fdin maior</i>	1	3,54	3,57	3,37	3,43
	2	3,47	3,42	3,19	3,38
	3	3,56	3,61	3,54	3,53
	4	3,59	3,51	3,39	3,49
	5	3,64	3,63	3,40	3,50
	6	3,56	3,56	3,36	3,46
	7	3,56	3,62	3,37	3,52
	8	3,49	3,57	3,25	3,38
	9	3,39	3,47	3,24	3,31
	10	3,58	3,55	3,57	3,58

Tabela 4.13 – Resultados do critério de atendimento de datas para o problema unidimensional com peso de ocupação de 80% e peso de atendimento de datas de 20%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	3,49	3,63	3,25	3,59
	2	3,65	3,63	3,50	3,57
	3	3,56	3,61	3,61	3,66
	4	3,72	3,71	3,58	3,65
	5	3,70	3,69	3,55	3,67
	6	3,71	3,70	3,59	3,63
	7	3,70	3,70	3,58	3,61
	8	3,47	3,59	3,36	3,50
	9	3,69	3,69	3,53	3,57
	10	3,67	3,63	3,57	3,64
<i>Fdin menor</i>	1	5,50	5,21	4,84	5,62
	2	5,83	5,96	5,38	5,73
	3	5,79	5,85	5,68	5,89
	4	6,04	6,07	5,63	5,87
	5	6,04	6,08	5,73	6,02
	6	6,02	6,01	5,74	5,84
	7	6,04	6,04	5,79	5,86
	8	5,35	5,66	4,95	5,41
	9	5,97	5,77	5,57	5,72
	10	5,86	5,81	5,81	5,83
<i>Fdin Ho</i>	1	0,91	1,15	0,88	1,05
	2	1,13	1,13	1,07	1,12
	3	1,15	1,18	1,19	1,18
	4	1,21	1,22	1,06	1,17
	5	1,22	1,20	1,10	1,19
	6	1,19	1,19	1,10	1,19
	7	1,20	1,23	1,14	1,17
	8	0,98	1,15	0,84	1,18
	9	1,16	1,18	1,10	1,12
	10	1,20	1,16	1,19	1,17
<i>Fdin maior</i>	1	3,20	3,17	2,68	3,24
	2	3,29	3,37	2,82	3,30
	3	3,30	3,18	3,43	3,39
	4	3,60	3,60	3,21	3,38
	5	3,51	3,59	3,30	3,56
	6	3,55	3,50	3,31	3,47
	7	3,59	3,54	3,33	3,38
	8	3,38	3,29	2,88	3,10
	9	3,31	3,36	3,11	3,36
	10	3,40	3,47	3,33	3,46

Tabela 4.14 – Resultados do critério ponderado para o problema unidimensional com peso de ocupação de 20% e peso de atendimento de datas de 80%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	1	52,30%	30,83%	21,09%	60,73%
	2	52,37%	56,39%	87,99%	51,79%
	3	76,36%	62,70%	60,75%	65,97%
	4	80,78%	84,22%	94,01%	90,97%
	5	86,99%	91,05%	95,69%	86,79%
	6	75,69%	90,57%	96,05%	78,80%
	7	98,04%	94,07%	96,40%	89,15%
	8	26,89%	35,12%	34,61%	65,25%
	9	58,58%	63,10%	86,57%	34,76%
	10	76,82%	70,70%	68,50%	72,16%
<i>Fdin menor</i>	1	50,83%	26,56%	48,09%	17,70%
	2	74,36%	73,51%	33,03%	82,35%
	3	48,96%	47,95%	85,86%	88,67%
	4	88,77%	94,74%	55,75%	55,86%
	5	82,85%	77,13%	65,71%	63,61%
	6	85,93%	92,03%	70,70%	90,73%
	7	86,81%	92,74%	68,58%	93,40%
	8	23,35%	29,73%	75,75%	34,29%
	9	58,06%	66,85%	22,67%	74,04%
	10	67,34%	57,13%	90,25%	79,20%
<i>Fdin Ho</i>	1	8,87%	62,41%	28,10%	11,05%
	2	90,04%	37,08%	56,94%	73,78%
	3	86,21%	85,27%	74,32%	80,09%
	4	78,86%	75,37%	78,07%	92,65%
	5	67,44%	75,96%	88,52%	90,15%
	6	83,82%	51,34%	85,51%	87,01%
	7	87,59%	74,13%	87,85%	86,40%
	8	32,34%	67,47%	40,67%	44,24%
	9	90,12%	28,65%	73,56%	77,50%
	10	85,88%	79,64%	75,77%	85,74%
<i>Fdin maior</i>	1	41,52%	44,00%	24,76%	64,62%
	2	62,25%	61,18%	75,30%	46,98%
	3	74,22%	68,31%	70,92%	78,42%
	4	71,96%	91,72%	94,03%	79,41%
	5	73,69%	89,67%	94,04%	75,70%
	6	72,20%	91,35%	93,61%	82,52%
	7	75,50%	96,65%	95,86%	83,57%
	8	37,52%	29,29%	32,22%	77,71%
	9	32,07%	58,91%	79,57%	33,99%
	10	83,87%	54,57%	81,59%	69,96%

TABELA 4.15 – Resultados do critério ponderado para o problema unidimensional com peso de ocupação de 50% e peso de atendimento de datas de 50%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	1	38,54%	27,78%	69,33%	31,24%
	2	67,94%	80,22%	25,82%	67,54%
	3	63,27%	48,03%	89,58%	90,45%
	4	77,78%	94,71%	51,59%	59,48%
	5	96,25%	97,03%	81,68%	65,75%
	6	90,58%	95,74%	50,38%	77,95%
	7	87,29%	91,94%	72,29%	77,87%
	8	28,58%	23,41%	77,19%	31,16%
	9	67,51%	83,45%	36,73%	68,42%
	10	55,90%	63,92%	92,26%	86,85%
<i>Fdin menor</i>	1	33,08%	78,25%	22,80%	11,94%
	2	89,15%	26,62%	51,66%	86,82%
	3	90,02%	81,97%	70,38%	79,05%
	4	80,27%	80,90%	71,10%	87,92%
	5	81,89%	80,74%	70,29%	93,87%
	6	95,56%	70,37%	69,09%	85,81%
	7	93,46%	76,88%	73,76%	82,85%
	8	5,77%	63,49%	26,48%	19,90%
	9	93,51%	24,62%	63,03%	82,88%
	10	88,42%	82,09%	74,31%	76,45%
<i>Fdin maior</i>	1	40,23%	30,67%	21,74%	49,94%
	2	43,11%	76,78%	83,67%	46,42%
	3	50,70%	70,01%	67,42%	62,03%
	4	68,97%	80,80%	93,83%	67,87%
	5	82,50%	98,32%	96,17%	94,36%
	6	71,58%	89,18%	94,87%	60,06%
	7	92,42%	90,95%	93,69%	84,35%
	8	47,52%	42,88%	31,78%	70,64%
	9	26,54%	64,16%	85,30%	56,92%
	10	61,81%	78,21%	68,52%	51,19%
<i>Fdin Ho</i>	1	49,76%	21,98%	80,32%	30,40%
	2	63,51%	82,55%	42,53%	56,58%
	3	68,39%	61,02%	79,08%	88,01%
	4	90,21%	96,82%	44,75%	64,91%
	5	94,94%	92,30%	74,84%	81,54%
	6	81,39%	91,59%	63,35%	72,68%
	7	91,25%	96,62%	61,45%	77,13%
	8	54,31%	43,00%	72,41%	39,71%
	9	62,74%	86,17%	20,64%	65,69%
	10	56,99%	69,54%	88,50%	83,50%

TABELA 4.16 – Resultados do critério ponderado para o problema unidimensional com peso de ocupação de 80% e peso de atendimento de datas de 20%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	1	4,77%	64,82%	59,59%	28,27%
	2	90,34%	32,81%	59,96%	82,62%
	3	91,04%	83,24%	86,67%	81,18%
	4	84,05%	53,95%	76,24%	90,60%
	5	79,71%	70,10%	86,23%	91,46%
	6	89,14%	75,92%	73,08%	91,94%
	7	87,52%	64,14%	80,80%	92,37%
	8	31,09%	67,33%	41,21%	34,49%
	9	94,59%	24,97%	62,70%	83,90%
	10	91,01%	69,47%	70,91%	80,93%
<i>Fdin menor</i>	1	65,07%	60,62%	28,86%	60,34%
	2	45,76%	65,79%	75,12%	49,02%
	3	75,18%	60,75%	75,27%	73,54%
	4	72,32%	84,49%	95,31%	81,18%
	5	80,83%	96,04%	90,57%	58,84%
	6	78,01%	84,16%	95,53%	62,63%
	7	87,86%	82,90%	96,49%	72,09%
	8	69,23%	24,37%	42,87%	47,35%
	9	15,50%	56,50%	82,87%	19,95%
	10	83,66%	73,56%	68,54%	70,21%
<i>Fdin Ho</i>	1	48,15%	27,73%	66,00%	37,73%
	2	58,01%	78,37%	47,23%	63,08%
	3	59,95%	59,89%	72,72%	88,01%
	4	69,80%	94,47%	66,76%	70,31%
	5	92,91%	95,61%	57,93%	67,58%
	6	79,31%	93,01%	53,86%	74,11%
	7	88,75%	95,28%	67,08%	76,37%
	8	45,43%	26,93%	67,24%	25,21%
	9	65,12%	81,28%	21,36%	63,92%
	10	60,75%	54,31%	86,19%	87,52%
<i>Fdin maior</i>	1	18,86%	77,64%	39,21%	12,84%
	2	82,95%	24,10%	59,71%	90,11%
	3	82,66%	83,41%	78,47%	82,42%
	4	83,48%	69,77%	73,07%	84,18%
	5	71,15%	52,81%	73,53%	93,43%
	6	85,48%	65,11%	67,59%	88,69%
	7	88,90%	74,52%	75,86%	85,74%
	8	9,03%	74,17%	18,79%	28,40%
	9	80,92%	41,77%	54,96%	88,79%
	10	90,12%	70,40%	78,73%	78,34%

4.3 ANÁLISE DOS RESULTADOS

Com o objetivo de entender o desempenho dos métodos heurísticos propostos comparativamente, sob a ótica dos critérios analisados, e ter clareza do método a ser utilizado no caso prático em questão, foram procedidas análises dos resultados com relação aos fatores apresentados.

Para tais fins, inicialmente houve a intenção de utilizar o método estatístico da análise de variância com múltiplos fatores. Entretanto, verificou-se pelo teste de Levene (para distribuições contínuas em geral) que não se verifica a condição de homocedasticidade, conforme as telas do apêndice C que retratam o *output* do *software* MINITAB 14 para os resultados obtidos no caso unidimensional.

Como fica verificado, rejeita-se a hipótese nula de que todas as variâncias são iguais. Assim, a análise das diferenças entre as heurísticas será feita através do teste não paramétrico de Kruskal-Wallis, que testa a hipótese nula de que as medianas de duas ou mais populações são iguais (MONTGOMERY e RUNGER, 2003). A única premissa do modelo é que as amostras sejam de uma distribuição contínua.

As figuras 5.1, 5.2 e 5.3 mostram os resultados obtidos também no MINITAB 14, respectivamente para os critérios de ocupação, atendimento de datas e ponderado para o problema unidimensional. Os demais problemas têm seus resultados no apêndice D.

Kruskal-Wallis Test: Ocupação versus CodHeur				
Kruskal-Wallis Test on Ocupação				
CodHeur	N	Median	Ave Rank	Z
1	240	0,9840	527,0	-15,87
2	240	0,9980	1455,1	6,00
3	240	0,9942	1000,2	-4,72
4	240	0,9975	1413,3	5,02
5	240	0,9980	1455,9	6,02
6	240	0,9981	1569,8	8,70
7	240	0,9983	1551,5	8,27
8	240	0,9836	550,8	-15,31
9	240	0,9983	1482,6	6,65
10	240	0,9941	998,7	-4,76
Overall	2400		1200,5	
H = 735,24 DF = 9 P = 0,000				
H = 735,24 DF = 9 P = 0,000 (adjusted for ties)				

Figura 5.1 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema unidimensional

Kruskal-Wallis Test: Atendimento de datas versus CodHeur				
Kruskal-Wallis Test on Atendimento de datas				
CodHeur	N	Median	Ave Rank	Z
1	240	3,599	1154,2	-1,09
2	240	3,535	1082,9	-2,77
3	240	3,647	1255,8	1,30
4	240	3,630	1244,2	1,03
5	240	3,645	1285,8	2,01
6	240	3,635	1228,1	0,65
7	240	3,645	1272,9	1,71
8	240	3,580	1135,1	-1,54
9	240	3,529	1071,4	-3,04
10	240	3,649	1274,5	1,74
Overall	2400		1200,5	
H = 30,32 DF = 9 P = 0,000				
H = 30,32 DF = 9 P = 0,000 (adjusted for ties)				

Figura 5.2 – Resultado do teste de Kruskal-Wallis para o critério de atendimento de datas do problema unidimensional

Kruskal-Wallis Test: Ponderado versus CodHeur				
Kruskal-Wallis Test on Ponderado				
CodHeur	N	Median	Ave Rank	Z
1	240	0,4105	551,9	-15,28
2	240	0,6972	1021,6	-4,21
3	240	0,8000	1291,5	2,14
4	240	0,8319	1467,8	6,30
5	240	0,8902	1608,0	9,60
6	240	0,8573	1506,1	7,20
7	240	0,8926	1666,8	10,99
8	240	0,3893	585,9	-14,48
9	240	0,6415	981,9	-5,15
10	240	0,7929	1323,5	2,90
Overall	2400		1200,5	
H = 724,62 DF = 9 P = 0,000				
H = 724,62 DF = 9 P = 0,000 (adjusted for ties)				

Figura 5.3 – Resultado do teste de Kruskal-Wallis para o critério ponderado do problema unidimensional

É possível verificar que em todos os testes o valor-p foi pequeno suficiente (todos nulos) para que a hipótese nula da igualdade das medianas seja rejeitada para níveis de significância de até 0,1%. Isso ocorreu nos três problemas para todos os 3 critérios.

Como o software disponível só possibilitou o teste para um único fator, foi testada apenas a diferença entre heurísticas para saber sobre a diferença entre elas. Os valores Z gerados na última coluna de cada tela mostram a heurística com melhor desempenho geral.

Uma vez verificada a diferença significativa entre as heurísticas em geral, para a avaliação da influência dos fatores tomou-se uma das heurísticas como *benchmarking*, analisando o desempenho das demais em relação a esta. Foi escolhida a heurística 4 (probabilística sem viés) por seu caráter de aleatoriedade na escolha dos itens a serem posicionados (embora seu procedimento geral probabilístico já lhe garanta certa eficácia melhorada).

As medidas adotadas para tal comparação foram:

- i. Variação percentual média em relação à heurística 4, ou seja, a relação (4.3);

$$\Delta Desempenho_i = \left(\sum_{j=1}^n \frac{Desempenho_{i,j} - Desempenho_{4,j}}{Desempenho_{4,j}} \right) \frac{1}{n} \quad (4.3)$$

- ii. Porcentagem de sucessos em relação à heurística 4, ou seja, em quantos dos casos analisados, a heurística obteve desempenho superior ao *benchmarking*.

Para ter uma visão completa dos experimentos em relação aos fatores estudados, foram feitos diferentes cortes dos resultados: consolidado, cenários de pesos dos critérios, cenários de distribuição das folgas dinâmicas e cenários de distribuição das dimensões dos itens. Além de tais cortes, os painéis de resultados foram subdivididos entre os 3 problemas (unidimensional, bidimensional guilhotinado e bidimensional não-guilhotinado) para melhor organização e visão de tais análises.

4.4 DISCUSSÃO DOS EXPERIMENTOS

Para as duas métricas e os quatro cortes definidos na seção anterior, foram construídos gráficos para visualizar e entender as tendências principais dos experimentos realizados. Aqui será colocado o painel de gráficos do problema unidimensional. Novamente, pelo volume grande de dados, os dois outros problemas serão colocados no apêndice E, embora serão referenciados nesta discussão.

Os gráficos 4.1 a 4.24 mostram os resultados consolidados do problema unidimensional. Os principais tópicos que emergiram da discussão serão desenvolvidos nas subseções posteriores.

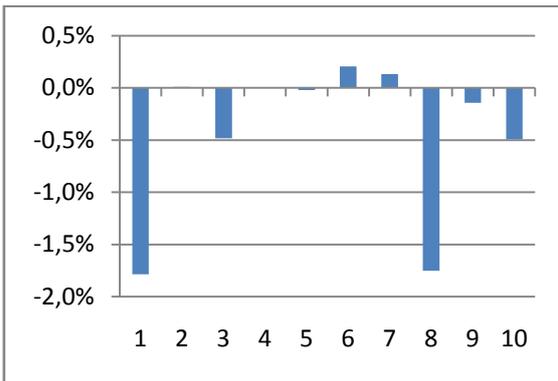


Gráfico 4.1 – Variação percentual média da ocupação em relação à heurística 4 (1D)

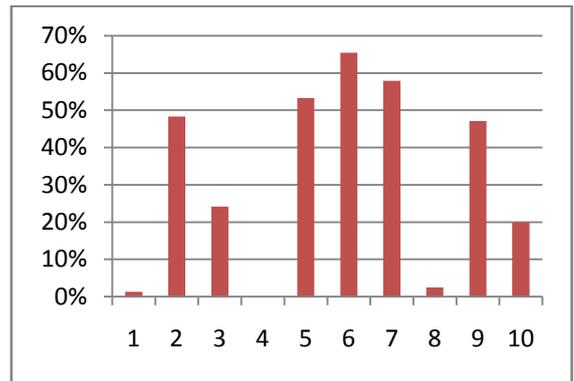


Gráfico 4.2 – Porcentagem de sucesso em ocupação em relação à heurística 4 (1D)

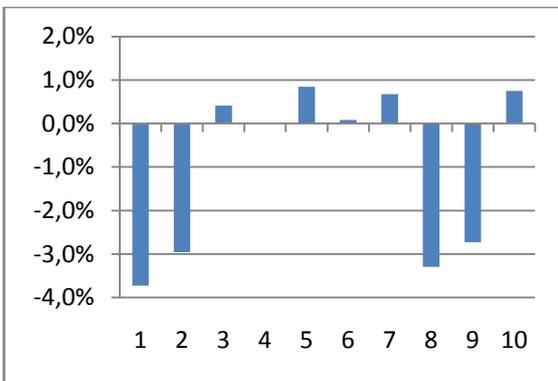


Gráfico 4.3 – Variação percentual média de atendimento de datas em relação à heurística 4 (1D)

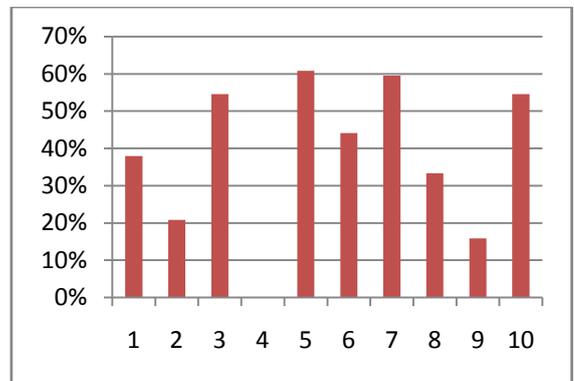


Gráfico 4.4 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 (1D)

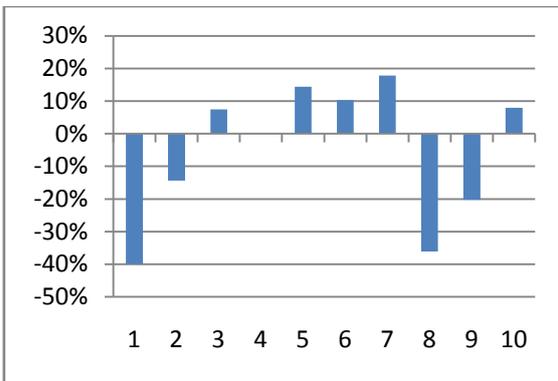


Gráfico 4.5 – Variação percentual média do critério ponderado em relação à heurística 4 (1D)

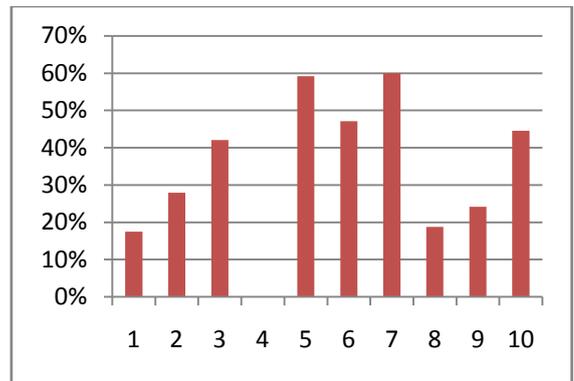


Gráfico 4.6 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 (1D)

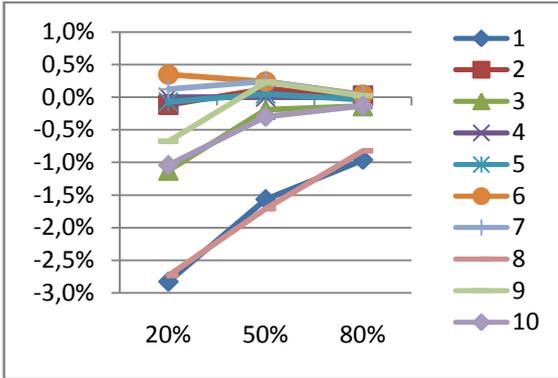


Gráfico 4.7 – Variação percentual média da ocupação em relação à heurística 4 por peso de ocupação (1D)

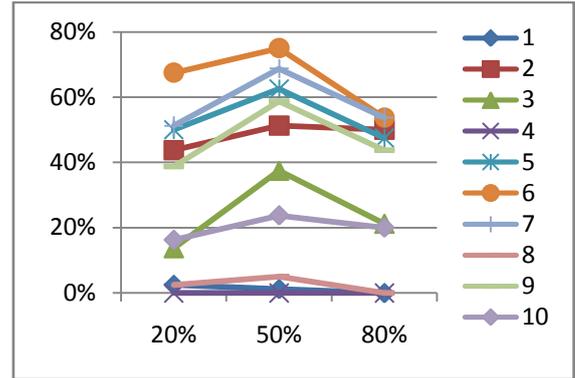


Gráfico 4.8 – Porcentagem de sucesso em ocupação em relação à heurística 4 por peso de ocupação (1D)

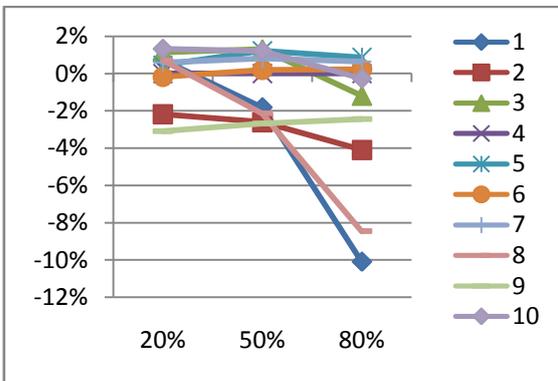


Gráfico 4.9 – Variação percentual média de atendimento de datas em relação à heurística 4 por peso de ocupação (1D)

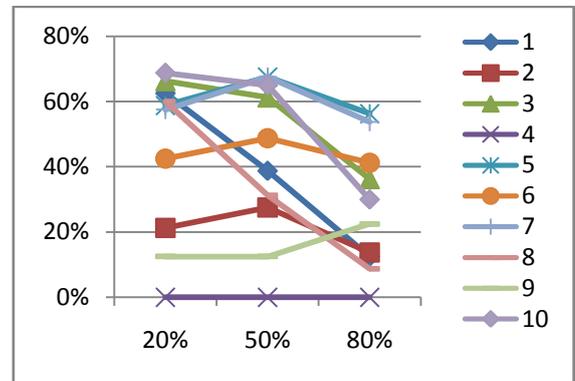


Gráfico 4.10 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 por peso de ocupação (1D)

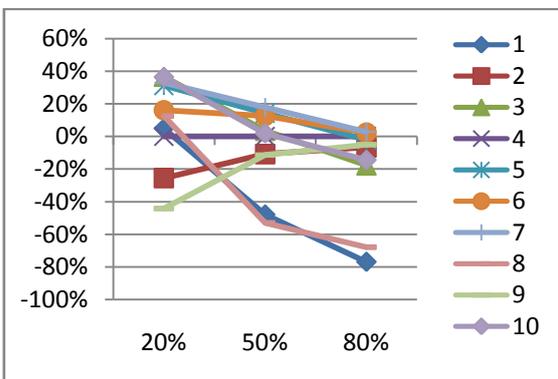


Gráfico 4.11 – Variação percentual média do critério ponderado em relação à heurística 4 por peso de ocupação (1D)

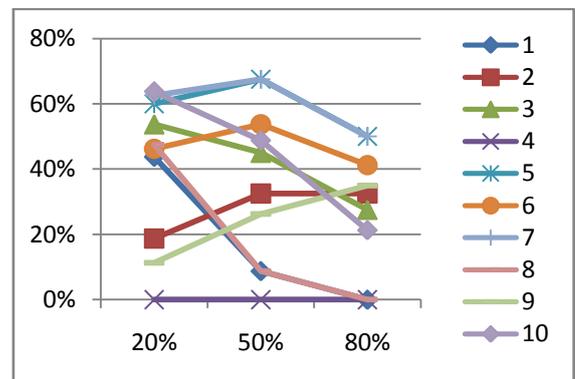


Gráfico 4.12 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 por peso de ocupação (1D)

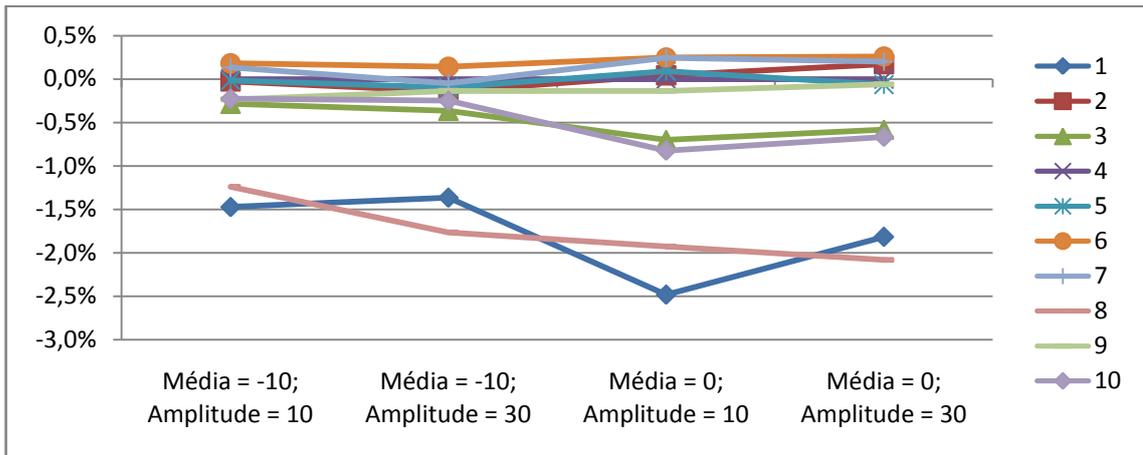


Gráfico 4.13 – Variação percentual média da ocupação em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)

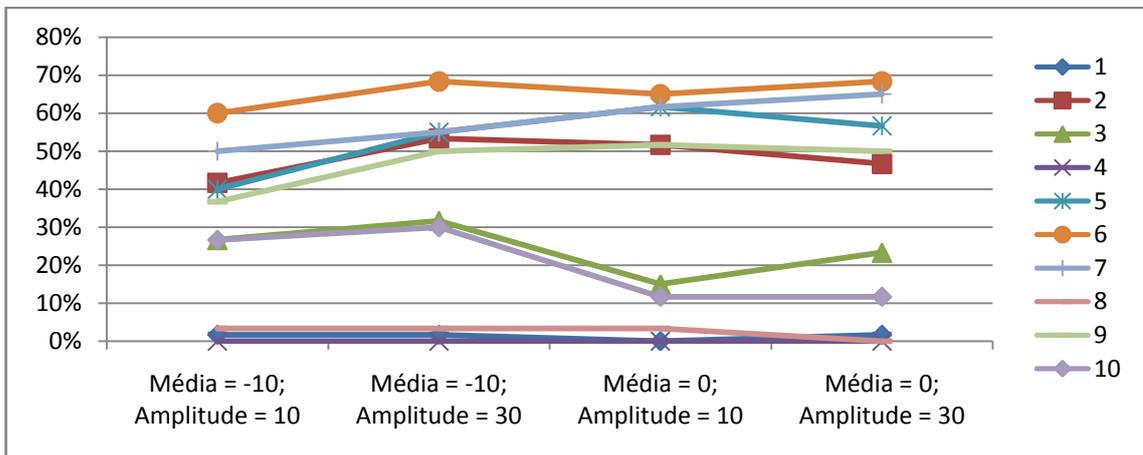


Gráfico 4.14 – Porcentagem de sucesso em ocupação em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)

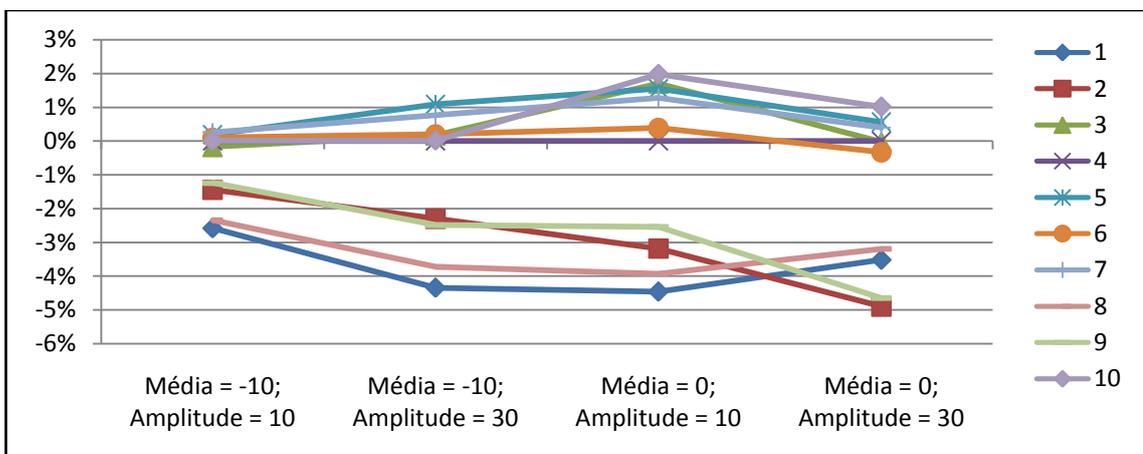


Gráfico 4.15 – Variação percentual média de atendimento de datas em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)

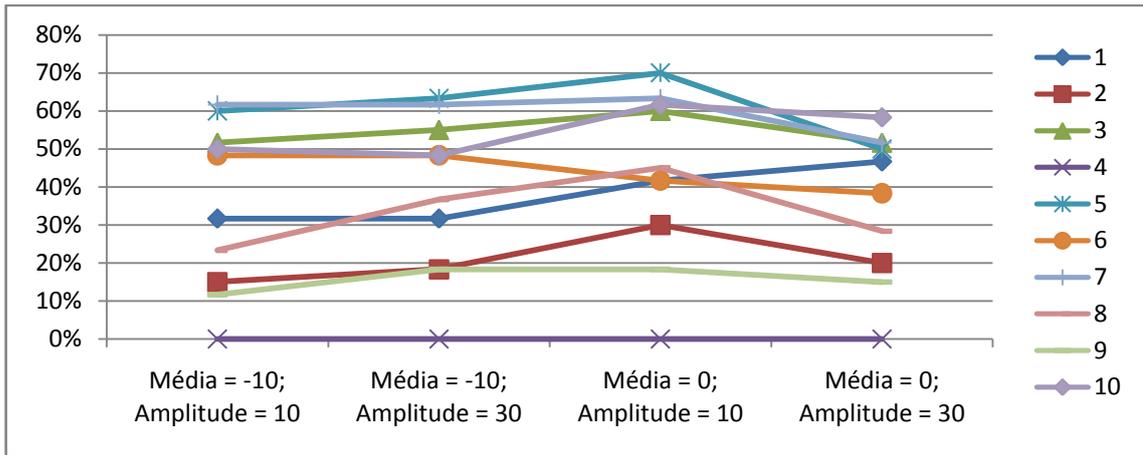


Gráfico 4.16 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)

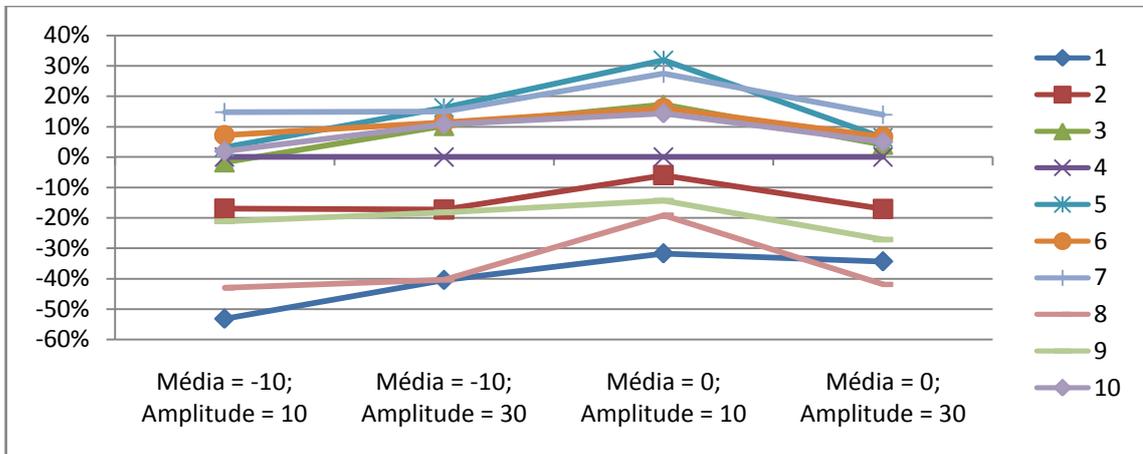


Gráfico 4.17 – Variação percentual média do critério ponderado em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)

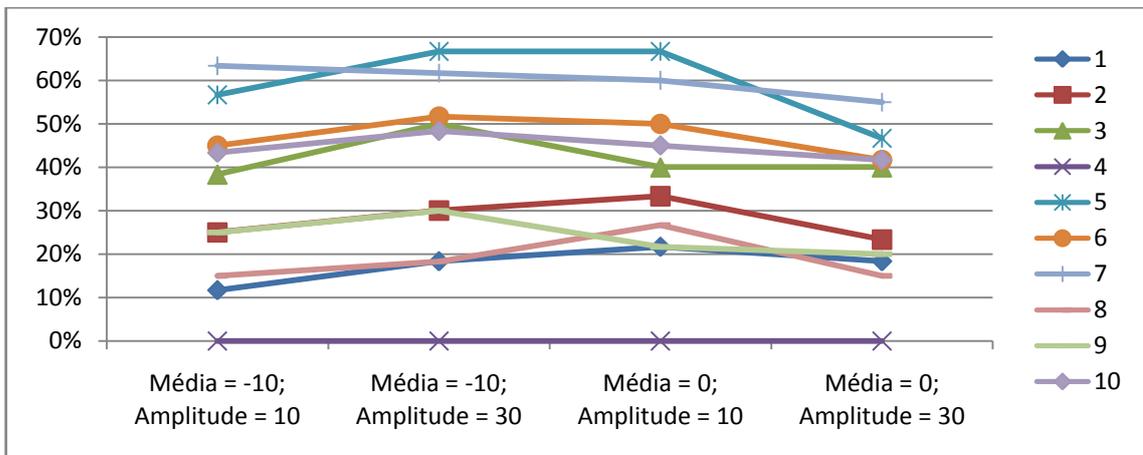


Gráfico 4.18 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (1D)

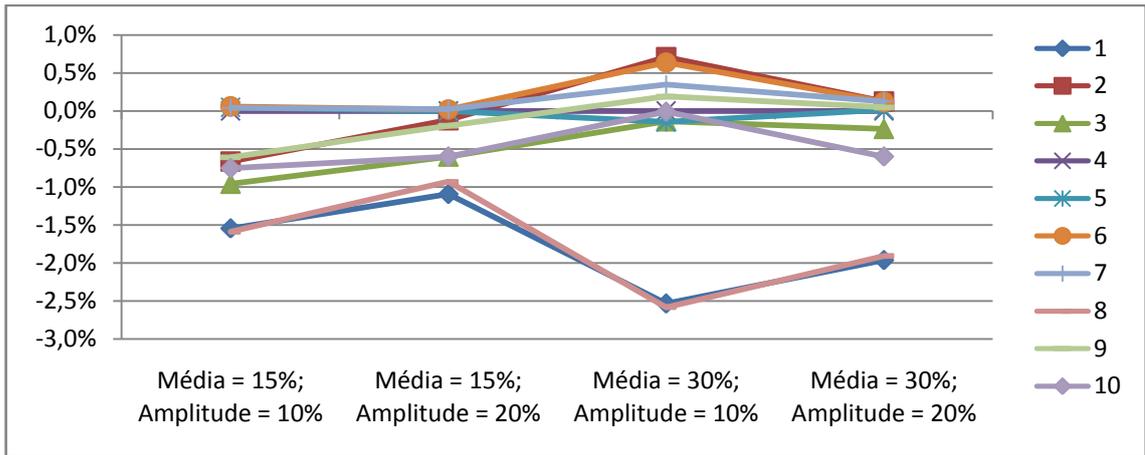


Gráfico 4.19 – Variação percentual média da ocupação em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)

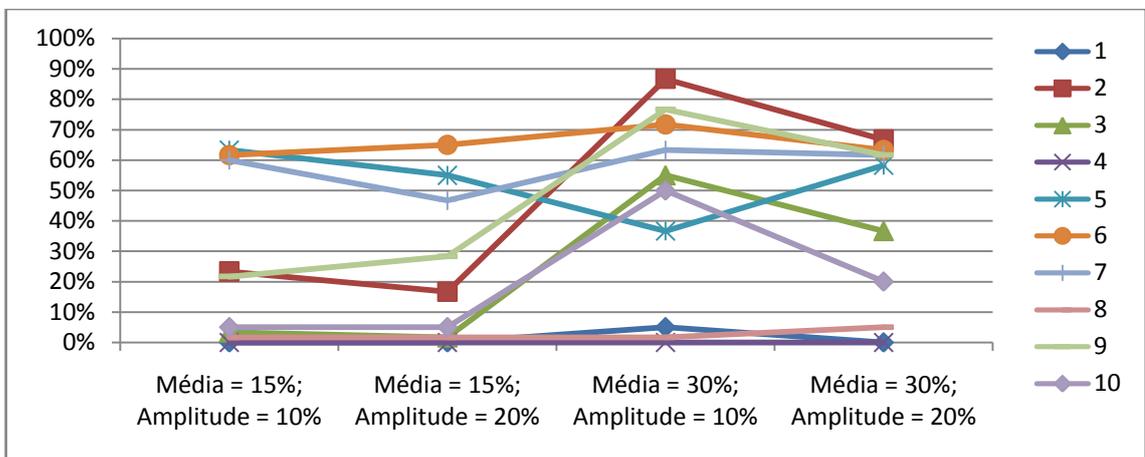


Gráfico 4.20 – Porcentagem de sucesso em ocupação em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)

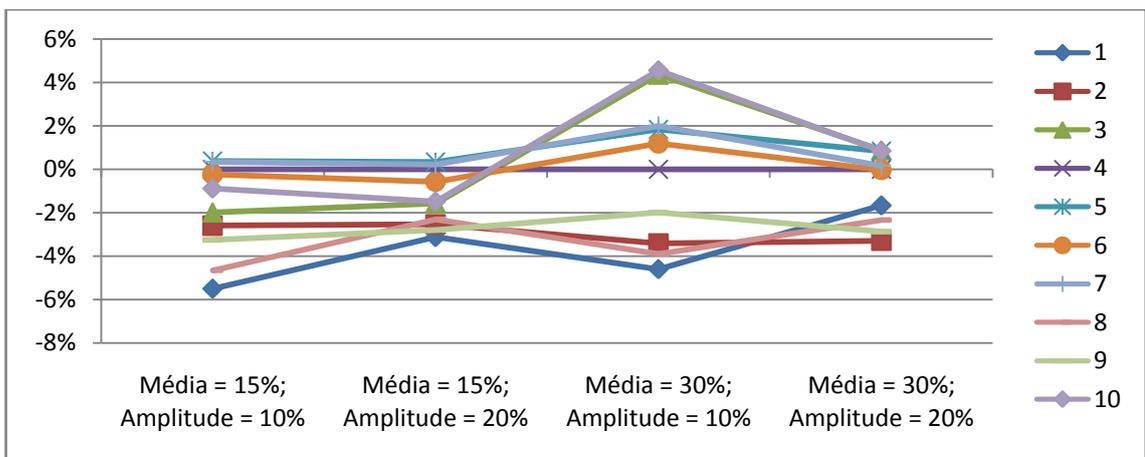


Gráfico 4.21 – Variação percentual média de atendimento de datas em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)

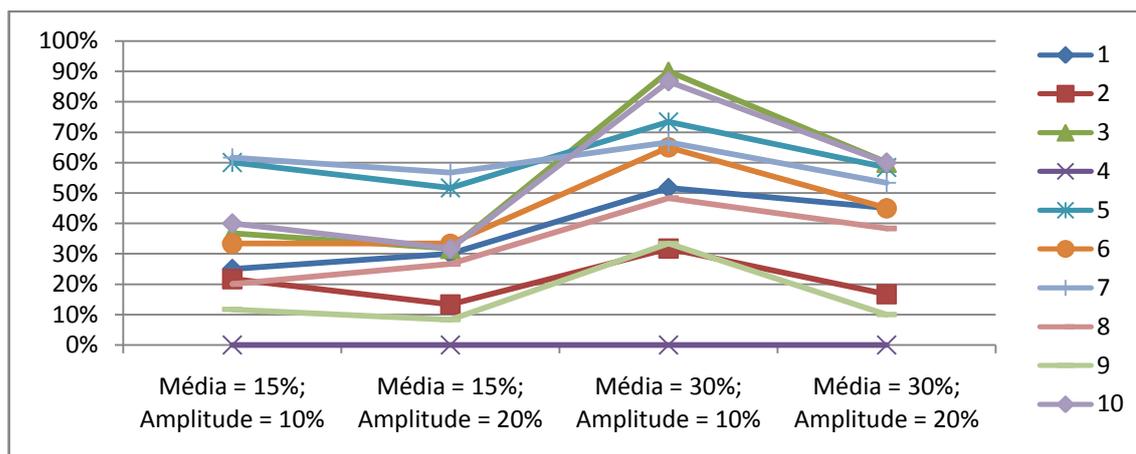


Gráfico 4.22 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)

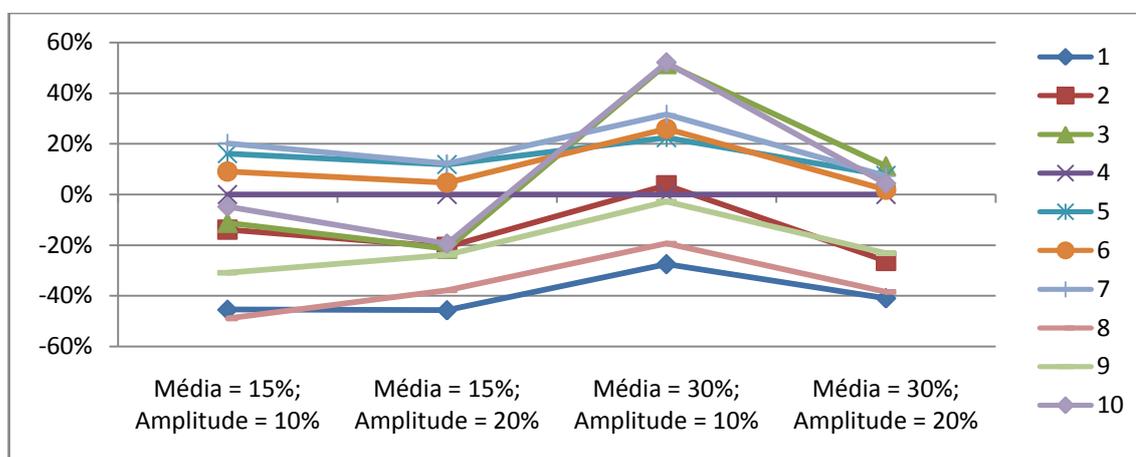


Gráfico 4.23 – Variação percentual média em critério ponderado em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)

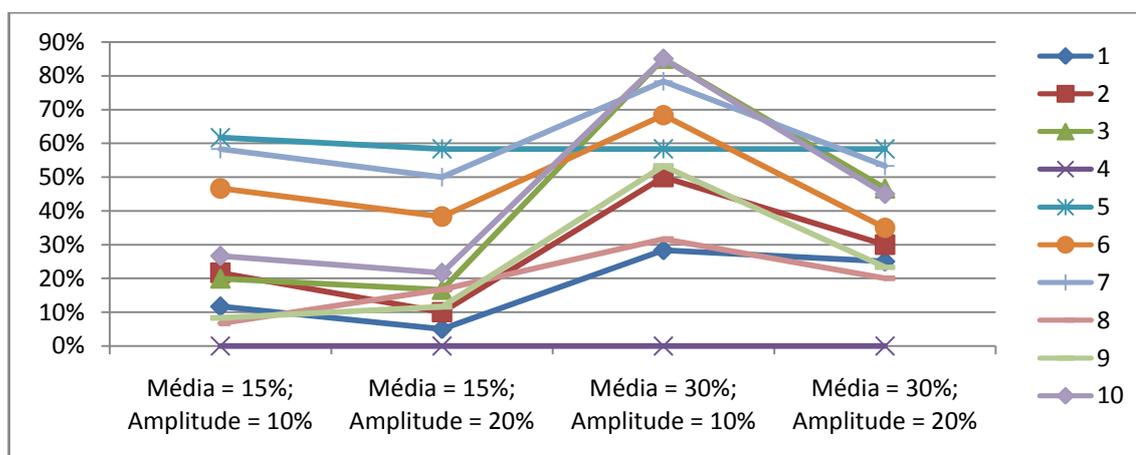


Gráfico 4.24 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (1D)

4.4.1 VIÉS DA HEURÍSTICA

A característica que mais chama atenção no conjunto de dados selecionados é a comparação entre os tipos de viés de escolha de itens no procedimento construtivo das heurísticas. Há uma superioridade nítida do conjunto que apresenta **viés linear**, que é mais tênue que os demais, com uma robustez que abarca os três tipos de problemas, os dois critérios de avaliação (bem como sua ponderação) e mantém resultados bons mesmo em diferentes ordenações iniciais.

Embora não haja grandes vantagens em média em relação à heurística 4 (probabilística sem viés), em geral duas ou as três versões deste viés conseguem manter-se com uma porcentagem de mais de 50% de sucesso comparativamente àquela.

Além dos bons resultados gerais, os corte analisados mostram que tal vantagem é suficientemente robusta em diferentes cenários: as três heurísticas apresentam-se superiores ou, ao mínimo, próximas das melhor sucedidas em cada caso.

Outro fato interessante é a qualidade do procedimento probabilístico sem viés, ainda que o seu componente de escolha na fila de ordens disponíveis seja – por definição – aleatório. A exploração de trocas pela natureza probabilística se mostra tão rica que impõe um resultado que em média supera as heurísticas mais viesadas.

Pensando nas heurísticas determinísticas como o limite superior do viés de uma heurística probabilística (em que o primeiro item da fila tem 100% de chance de ser escolhido), pode-se inferir dos resultados que a diversificação das soluções construídas é mais benéfica que a ordenação inicial imposta aos itens.

O estudo mostra não só que em média os resultados mais viesados (exponencial e determinístico) provém piores soluções, como também o quanto são “sensíveis” a mudanças do entorno, sendo sua qualidade ditada pela interação entre a ordem inicial e um conjunto de fatores “propício” ao sucesso dos primeiros itens. Isso fica claro, por exemplo, no gráfico 4.20 em que a heurística determinística ordenada por critério de ocupação se sobressai para um conjunto de itens com grandes dimensões, em que e pese o desempenho fraco para dimensões menores.

Surge o questionamento se existe uma região de viés ótimo, já que o viés tênue foi superior tanto à falta quanto ao excesso de tendências de escolha dos próximos itens.

4.4.2 ORDENAÇÃO INICIAL

Como a intuição sugere, a ordenação inicial dos itens impacta nos resultados das heurísticas, potencializando o critério que privilegia. Isso fica especialmente visível quando os indicadores são olhados isoladamente: de fato as heurísticas 2, 6, e 9 (ordenadas inicialmente por tamanho, que contempla o critério de ocupação) sobressaem-se sob a ótica exclusiva da ocupação; enquanto as heurísticas 1, 5 e 8 melhoram seu desempenho quando do crivo exclusivo do atendimento de datas, de maneira geral.

Os cortes por peso dos critérios deixam tal conclusão ainda mais evidente: gráficos como 4.7, 4.9 e 4.10 mostram claramente que as heurísticas ordenadas por folga dinâmica têm uma mudança brusca de desempenho de acordo com a força da ponderação.

Até então, os resultados são triviais e esperados. Chama a atenção, na verdade, é a discrepância de desempenho entre tais conjuntos de heurística: as heurísticas 1 e 8 – ambos ordenadas inicialmente por folga dinâmica – tiveram uma *performance* nitidamente inferior às demais. O gráfico 4.15, por exemplo, mostra que à luz de um corte por distribuição das folgas dinâmicas, tais métodos são consistentemente mais fracos mesmo no critério em que teoricamente seriam diferenciados. Tais heurísticas ordenadas por data, pelo contrário, potencializam seu desempenho quando da avaliação exclusiva por esse critério.

Outro resultado interessante diz respeito à ordenação inicial por critério ponderado. Além de não ser imediata a relação detectada de melhor desempenho das heurísticas quando nesta situação, verifica-se que estas apresentam – via de regra – valores de atendimento de datas muito competitivos, o que não ocorre com tanta facilidade em termos de ocupação. Isso fica evidente, por exemplo, no gráfico 4.4, em que as heurísticas inicialmente ordenadas por critério ponderado (3, 7 e 10) apresentam valores de atendimento de datas entre os maiores. Uma possível inferência da qualidade das soluções que se iniciam com esta ordenação é que esta impõe uma ordenação que privilegia o critério mais importante, de acordo com os pesos atribuídos; porém promove uma “diversificação dirigida”, ou seja, mescla itens com menor qualidade no critério mais pesado, tendo estes o viés de terem alta qualidade no outro critério.

4.4.3 PESO DOS CRITÉRIOS

O resultado mais aparente é bastante óbvio: o maior peso ao critério de ocupação faz com que as heurísticas ordenadas inicialmente orientadas a este quesito desempenhem melhor.

Algumas observações mais interessantes podem ser verificadas. Nos gráficos de sucesso das heurísticas para ocupação em relação ao *benchmarking*, vê-se um formato normal geral de pico em peso 50%-50%. Isso significa que a heurística sem viés tem seu valor relativo reduzido quando os pesos estão equilibrados. Tal fato pode ser encarado como resultante da falta de equilíbrio entre os critérios, fazendo com que o espaço de soluções viáveis do problema seja mais difícil de explorar.

Este corte é provavelmente o melhor indicativo das conclusões acerca da robustez associada à suavidade do viés da heurística: enquanto as determinísticas ou muito viesadas sofrem alterações muito fortes de desempenho, as de viés linear mantêm-se mais constantes nos três critérios analisados.

4.4.4 DISTRIBUIÇÕES DAS DIMENSÕES E FOLGAS DINÂMICAS

No que diz respeito às dimensões das peças, existe uma nítida mudança de comportamento da maioria das heurísticas quando o nível da dimensão média alterna de menor para maior. A avaliação de ocupação, isoladamente, apresenta uma leve tendência de alta para a maioria delas, mas uma queda brusca para as muito viesadas orientadas por folga dinâmica (1 e 8). O efeito é marcadamente este (para ambos os lados) sobretudo no nível mais homogêneo. Importante ressaltar que como se trata de uma medida relativa, eventualmente o que ocorre é uma queda de qualidade da heurística 4 (referência) neste caso.

O atendimento de datas, neste contexto do corte por dimensões, também apresentou uma peculiaridade em relação a todos os métodos: melhora com o aumento das dimensões das peças, como pode ser visto no gráfico 4.22, ainda que piore com o aumento da dispersão destas partes. Movimento idêntico é visto em 4.24 para o critério ponderado.

Por fim, a distribuição das folgas dinâmica foi o conjunto de fatores que mostrou menor influência no desempenho das heurísticas. A maioria das variações é discreta, seja qual

for o problema e o critério avaliado. Há apenas uma tendência aparente no Gráfico 4.15, em que as heurísticas ordenadas por tamanho (2, 6, 9) diminuem sua performance com o aumento das folga dinâmicas ou sua maior dispersão. O oposto não ocorre como regra nas ordenadas por folga dinâmica, que, pela lógica, se beneficiam de folgas dinâmicas mais extremas no conjunto de itens candidatos.

5 APLICAÇÃO PRÁTICA

As heurísticas propostas foram implementadas na forma de um sistema em linguagem Visual Basic, com interface em uma planilha em Microsoft Excel®. Tal sistema está em fase de implantação atualmente, tendo como fator crítico de implementação a coleta sistemática dos dados de dimensões dos itens envolvidos. Em geral, tais informações existem de forma desestruturada (campos descritivos) no ERP da empresa, de forma a dificultar seu uso. Coletas paliativas estão suprindo seu uso durante os testes e adaptações estão sendo feitas para seu uso perene.

Aqui são apresentados três pilotos feitos em centros de trabalhos importantes, cada um representando um tipo problema apresentado:

- a) O caso unidimensional foi testado no centro de trabalho de “Corte de Serra Fita”;
- b) O caso bidimensional guilhotinado foi testado no “Corte de placas”;
- c) O caso bidimensional não guilhotinado foi testado no “*Nesting*” (corte por fresamento).

A coleta de dados dos itens e objetos foi feita junto aos centros de trabalho com ajuda dos supervisores e monitores das áreas, tanto do setor de PCP quanto de fabricação.

A heurística escolhida, com base nos resultados obtidos foi a probabilística ordenada inicial por critério ponderado com viés linear (heurística 7). Os motivos para tal residem não apenas no fato de ter sido o método com maior resultado geral observando os critérios ponderados, mas também pela robustez em relação a diferentes cenários de dimensões e folgas dinâmicas das peças.

A rodada foi parametrizada com 80% de importância para atendimento de datas e 20% para ocupação, de maneira alinhada com a característica deste ambiente de produção, em que paradas de linha de montagem têm um impacto muito forte no resultado geral da fábrica. Os critérios de parada foram mantidos os mesmos dos experimentos, já que os tempos computacionais encontrados não foram considerados proibitivos, já que são muito menores do que o tempo de execução de um lote.

5.1. RESULTADOS NO CORTE DE SERRA FITA

O teste piloto do problema unidimensional foi feito no corte por serra fita, até por se tratar do único centro de trabalho com formação de lotes com esta dimensionalidade.

Ocorreu com base na fila de ordens de 21 de setembro de 2009, que totalizava **26 ordens**, distribuídas em **21 diferentes tipos de critérios de agrupamento**. Neste caso, os critérios de agrupamento são os diferentes diâmetros e materiais dos tubos e perfis a serem cortados. O grande número de critérios de agrupamento em relação ao número de ordens deixa o porte do problema bem mais simples.

Foram gerados **67 padrões de corte** para um total de **268 peças**. As avaliações de atendimento de datas e ocupação dos padrões estão retratadas no gráfico 5.1.

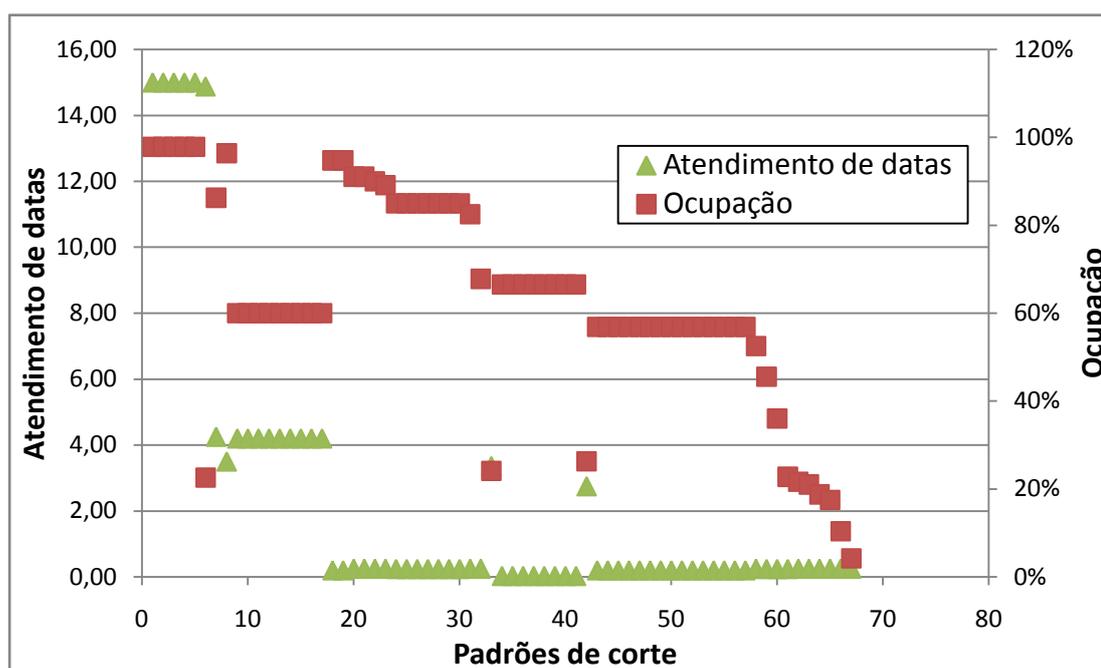


Gráfico 5.1 – Avaliações de atendimento de datas e ocupação dos padrões de corte gerados para o corte de placas (caso unidimensional)

5.2. RESULTADOS NO CORTE DE PLACAS

Para o caso bidimensional guilhotinado, o piloto tomou a máquina de corte de placas para teste. O motivo da escolha de deu, sobretudo, por se tratar do centro de trabalho – dentre os que possuem tal característica de corte – com maior fila de ordens, sendo um gargalo importante no processo de usinagem.

Igualmente ao caso anterior, tomou-se a fila do dia 21 de setembro de 2009, que apresentava um total de **923 ordens**, que consolidavam **7143 peças**. Importante mencionar que nem sempre uma peça gera um item, pois há ordens em que o desdobramento em itens ocorre em múltiplos. Neste caso, a multiplicidade ocorre na forma do recorte de uma tira maior, já que esta é mandada para usinagem em fornecedores subcontratados. As ordens estão distribuídas em **94 critérios de agrupamento**, que diferem entre si pelas espessuras e materiais das placas.

Foram gerados **189 padrões de corte** para o total de **6989 peças**. As avaliações de atendimento de datas e ocupação dos padrões estão retratadas no gráfico 5.2. Como a fila em questão é bastante grande, é apresentado um recorte dos primeiros padrões gerados no gráfico 5.3, pois são os que, de fato, serão executados com tais características.

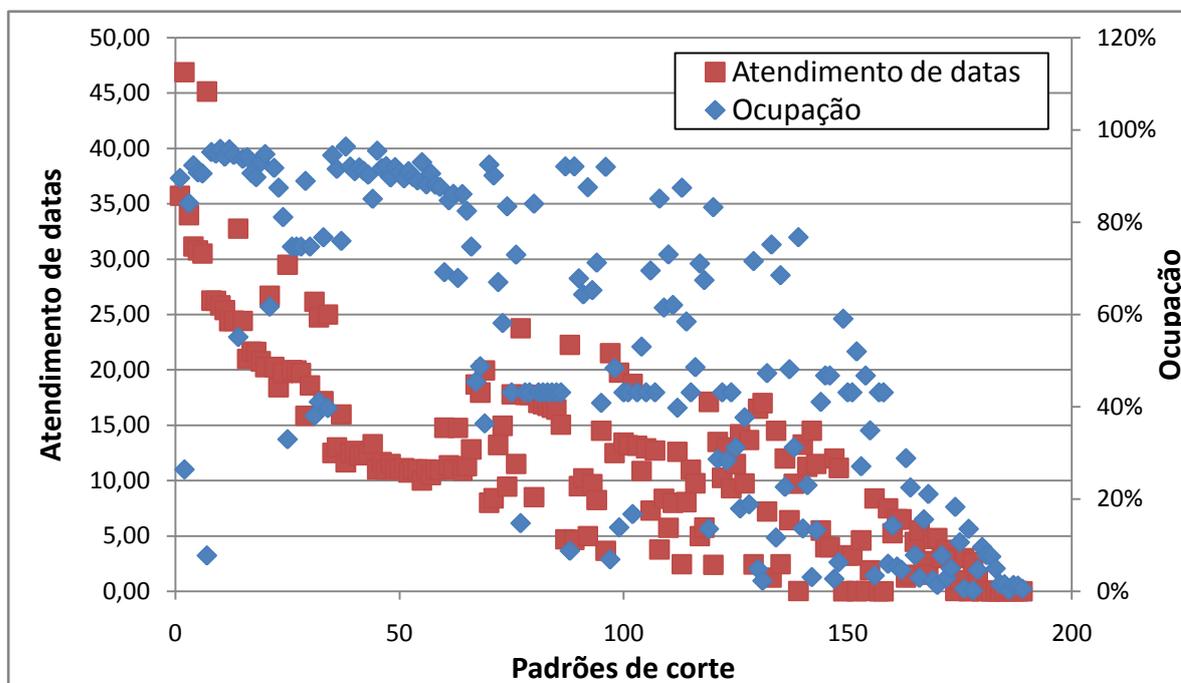


Gráfico 5.2 – Avaliações de atendimento de datas e ocupação dos padrões de corte gerados para o corte de placas (caso bidimensional guilhotinado)

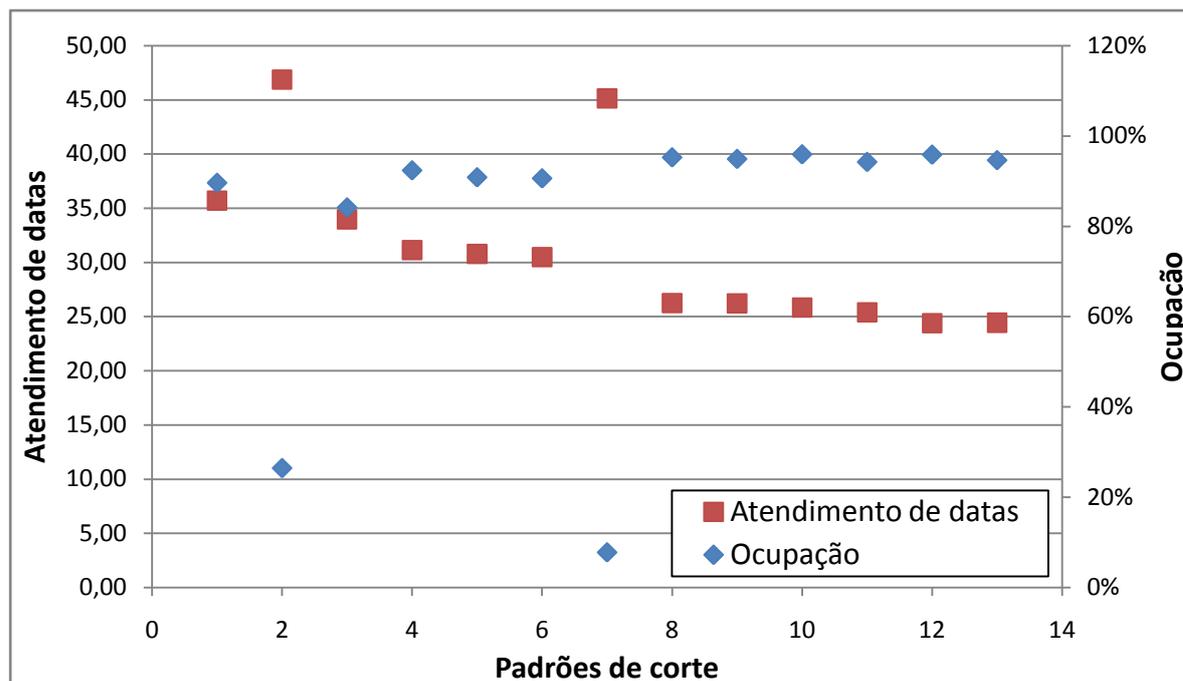


Gráfico 5.3 – Detalhe dos primeiros padrões de corte gerados para o corte de placas

5.3. RESULTADOS NO *NESTING*

Finalmente, para o caso bidimensional não-guilhotinado, o *nesting* foi escolhido para o teste-piloto. A razão para a escolha se deu por ser o centro de trabalho que centraliza o início do processo da maioria dos itens da estamparia. Embora não apresente problemas de capacidade, a questão do sequenciamento é crítica, pois o volume do fluxo diário de ordens é muito grande.

Em consonância com os casos anteriores, tomou-se a fila do dia 21 de setembro de 2009, que apresentava um total de **516 ordens**, que consolidavam **6332 peças**. Aqui vale a mesma relação de correspondência entre itens e peças através de múltiplos de produção. Neste caso, entretanto, esse se dá devido a uma necessidade técnica, pois o *nesting* é feito em pilhas de chapas. As ordens estão distribuídas em **49 critérios de agrupamento**, que diferem entre si pelas espessuras da pilhas e materiais das chapas.

Foram gerados **63 padrões de corte** para o total de **1260 peças**. As avaliações de atendimento de datas e ocupação dos padrões estão retratadas no gráfico 5.4. Como a fila em questão também é bastante grande, é apresentado um recorte dos primeiros padrões gerados no gráfico 5.5, pois são os que, de fato, serão executados com tais características.

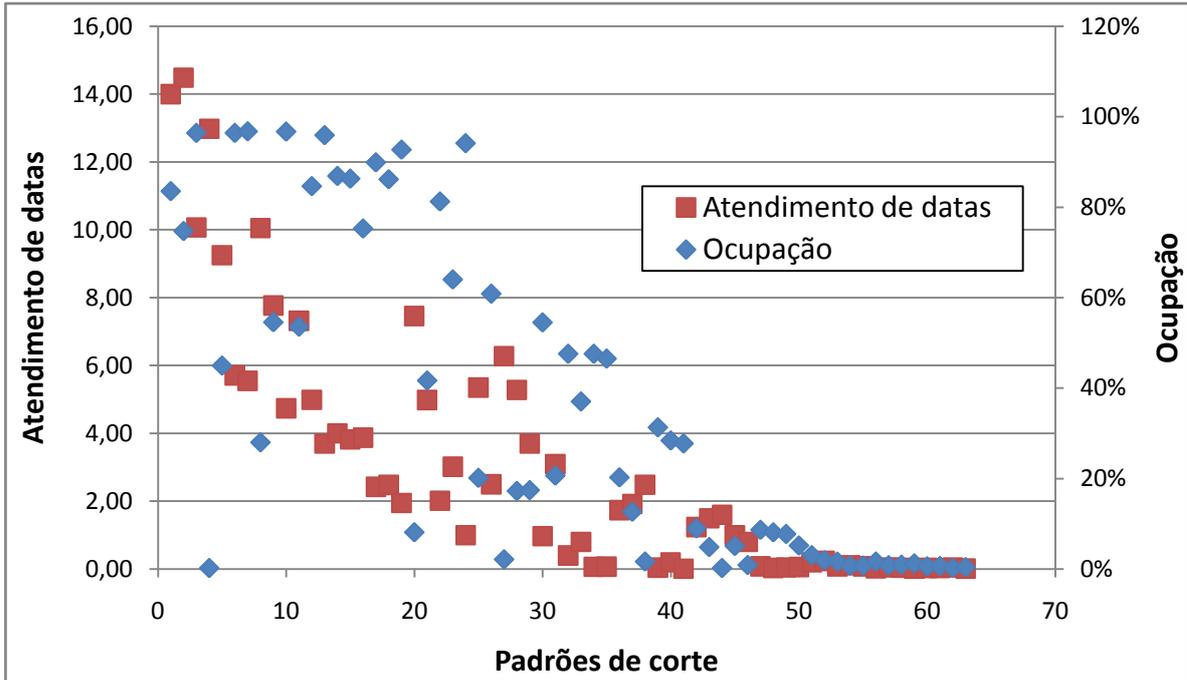


Gráfico 5.4 – Avaliações de atendimento de datas e ocupação dos padrões de corte gerados para o nesting (caso bidimensional não-guilhotinado)

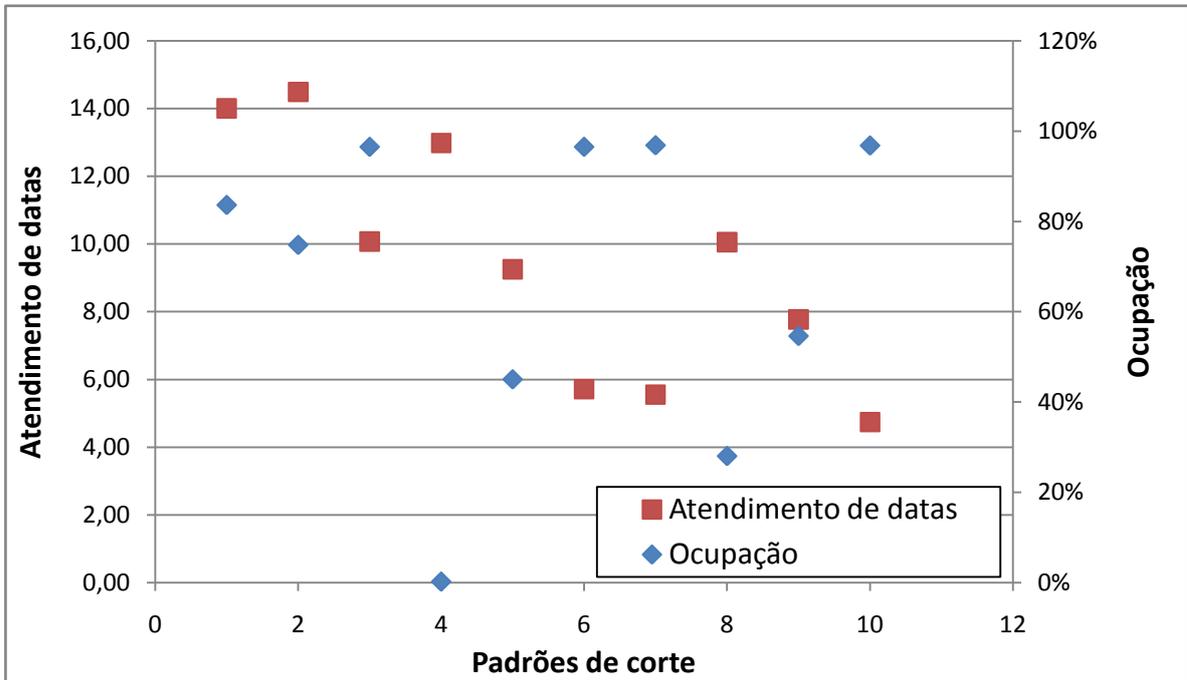


Gráfico 5.5 – Detalhe dos primeiros padrões de corte gerados para o nesting

5.4 DISCUSSÃO DOS RESULTADOS PRÁTICOS

As três aplicações, embora tenham dimensões também distintas, proveram resultados com características semelhantes.

Como o critério de atendimento de datas é preponderante pela parametrização, a ordem sequenciada dos padrões tem sua “espinha dorsal” ditada pelos valores deste critério. Isso fica especialmente claro nos casos bidimensionais e não-bidimensionais que têm maior quantidade de ordens em fila. Nestes, claramente a dispersão dos pontos que concernem a avaliação de ocupação é maior do que do outro critério, evidenciando sua menor influência na decisão final. Entretanto, a ocupação apresenta-se ativa em dois tipos de situação, via de regra:

- em casos de valores muito próximos no valor de atendimento de datas, a ocupação serve como uma espécie de critério de desempate, enriquecido pela propriedade de eventualmente compensar o que poderia se chamar de “empate técnico” (valores normalizados muito próximos);

- em casos extremos de aproveitamento da ocupação, esta pode causar um salto qualitativo na prioridade do sequenciamento de um padrão de corte. Isto quer dizer que um padrão muito pior ocupado do que os demais precisam ter um valor de atendimento de data muito forte para se equivaler. O oposto também vale: ocupações muito maiores que o normal priorizam o padrão de corte em relação aos demais, mesmo sendo mais fraca em termo de urgência. Isso é particularmente relevante para casos em que há falta de ordens de um padrão de corte.

O primeiro caso é particularmente visível no problema unidimensional, em que se nota três patamares de avaliações de atendimento de datas e o diferencial do sequenciamento está na consideração da ocupação. Já o segundo caso é mais visível nos detalhes dos problemas de maior porte (bidimensionais), em que vê-se claramente padrões muito valiosos em termos de datas perderem prioridade por terem ocupação muito fraca. Na prática, isto leva a uma situação em que padrões melhores são feitos anteriormente, possibilitando a chegada de novas ordens de tal agrupamento para melhorar a utilização da matéria-prima.

Outra característica dos resultados e do sistema como um todo vista como de grande valia é seu papel como instrumento de apoio à decisão: a empresa dispõe de *softwares* de otimização/melhoria de ocupação de objetos; entretanto não é possível fazer uma

consideração direta do atendimento de datas dos lotes, tampouco comparar diferentes critério de agrupamento em termos de priorização. Nesse sentido, o sistema provê ao seu usuário informações importantes, tais como:

- Um *lower bound* de ocupação, já orientado à melhor ocupação por atendimento de datas. Trata-se de um limitante inferior já que os sistemas de otimização de ocupação eventualmente podem superar as heurísticas construtivas aqui propostas. Entretanto, a **seleção dos itens** a serem considerados como *input* para tais sistemas agrega agilidade à sua operação;

- **Objeto** a ser escolhido: com critérios de ocupação e atendimento de datas sendo avaliados, é possível identificar o melhor momento de utilizar um retalho ou uma placa padrão. Esta decisão, anteriormente arbitrária, passa a ser suportada por uma regra sistemática;

- **Critério de agrupamento prioritário**: como há centros de trabalho com volumes grandes de ordens, como visto no caso do corte de placas e do *nesting*, a visualização do agrupamento é uma decisão de difícil incumbência apenas com o bom senso. Com uma avaliação comparativa dos melhores padrões de cada um deles, tal decisão fica agilizada para os operadores.

As máscaras do sistema em implantação na empresa estão no apêndice F, enquanto seu código-fonte em linguagem VBA encontra-se no apêndice G. O sistema pode ser manipulado na mídia magnética que acompanha este volume (Seque.xls).

6 CONCLUSÕES

O presente trabalho propôs-se a estudar o sequenciamento de ordens em centro de trabalho que apresentam a característica de operações em lotes, com a preocupação de equilibrar os objetivos de utilização da matéria-prima e atendimento de datas. A revisão da literatura mostra que se trata de uma intersecção pouco explorada de dois problemas bastante clássicos.

O estudo possibilitou a operacionalização de uma ferramenta de apoio à decisão que efetuou o equilíbrio entre estas duas perspectivas. De fato, cumpriu-se o objetivo maior de prover um sequenciamento que abarcasse ambas visões. Em essência, a problemática proposta é retrato de um dos *trade-offs* vividos em sistemas produtivos entre o PCP e a fabricação. Neste sentido, uma ferramenta flexível – por sua adaptabilidade em termos de pesos das ponderações, priorização dos objetos a serem escolhidos e uso de diferentes métodos heurísticos – dá à solução uma característica de robustez para enfrentar diferentes realidades encontradas nos centros produtivos ao longo do tempo. O sistema em questão está sujeito a mudanças constantes tanto de curto prazo (como urgências repentinas, aumento de filas ou situações de filas muito atrasadas ou adiantadas) como de longo prazo (alterações de plano de produção, surgimento de novos produtos, variações na cadência de produção).

Foram escolhidas as heurísticas construtivas, por sua agilidade de processamento, consonante com o contexto dinâmico de uma fábrica com milhares de itens e os mais diferentes roteiros de produção. A discussão do uso de heurísticas construtivas probabilísticas parece promissora no sentido de mitigar uma das deficiências iniciais de sua versão determinística: a de explorar de maneira muito pobre o espaço de soluções viáveis.

A porção experimental do trabalho emergiu questões interessantes, tanto do ponto de vista teórico, quanto prático. Para o primeiro, ficam pontos a ser melhor debatidos como a influência de parâmetros como diferentes ordenações iniciais e vieses na eficácia das heurísticas. Do ponto de vista da implementação, foram identificadas demandas da operação que impactam diretamente em decisões de produção. Mesmo em um contexto com alta formalização de procedimentos – já que existe um sequenciador único para todos os centros, a folga dinâmica – há características técnicas que geram ambiguidades, legando a decisão ao bom senso e experiências. Importante mencionar que muitas vezes tais capacidades dos operadores não só são bastante adequadas, como contribuíram para a construção da solução

neste estudo. Entretanto, a vulnerabilidade a mudanças e a inconstância dos padrões de decisão ficam em voga neste contexto.

Relevante ainda pontuar um subproduto do trabalho. Foi mencionado na seção 1 que, além do objetivo de melhora da programação e controle da produção, o projeto de consultoria que estabeleceu o vínculo desta aplicação também teve como objetivo promover o crescimento conceitual em planejamento e programação da produção na planta. Neste sentido, o envolvimento de direção, gerência, supervisão de diferentes departamentos, programadores e operadores no processo de construção e implementação da ferramenta fomentou a discussão da importância da coesão das interfaces entre áreas (em particular, PCP e fabricação), além do pensamento multiobjetivo, importante para o sucesso global da planta.

Do ponto de vista mais pragmático, as heurísticas mostraram desempenho satisfatório em termo de ocupação (acima de 80% para os primeiros padrões com um número de itens suficientes para tal), com o benefício de ter uma seleção orientada também ao atendimento de datas.

Como sugestões de estudos futuros, sugere-se:

- Exploração em maior profundidade da eficácia e eficiência de heurísticas construtivas probabilísticas, comparando com heurísticas de busca ou algoritmos otimizantes;
- Exploração de novos parâmetros na mesma lógica, sobretudo no que diz respeito ao viés de escolha e a ordenação inicial da fila;
- Desenvolvimento de procedimento análogo para o caso tridimensional e para o problema de *setup dependente de seqüência*.

REFERÊNCIAS

ALLAHVERDI, A.; NG, C. T.; CHENG, T. C. E.; KOVALYOV, M. Y. A survey of scheduling problems with setup times or costs. **European Journal of Operational Research**, Amsterdam, v. 187, p. 985-1032, 2008.

ARCARO, V. F. **Recorte de Estoque Unidimensional**. 1988. Dissertação (Mestrado) – Instituto de Ciências Matemáticas de São Carlos, Universidade de São Paulo, São Carlos, 1988.

ARENALES, M.; MORABITO, R. An AND/OR-graph approach to the solution of two-dimensional non-guillotine cutting. **European Journal of Operational Research**, Amsterdam, v. 87, p. 599-617, 1995.

ARMBRUSTER, M. A solution procedure for a pattern sequencing problem as part of a one-dimensional cutting stock problem in the steel industry. **European Journal of Operational Research**, Amsterdam, v. 141, p. 328-340, 2002.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO 9000:2000: Sistemas de gestão da qualidade. Rio de Janeiro, 2000.

BECCENERI, J. C.; YANASSE, H. H.; SOMA, N. Y. A method for solving the minimization of the maximum number of open stacks problem within a cutting process. **Computers & BOperations Research**, Oxford, v. 31, p. 2315-2332, 2004.

BELOV, G.; SCHEITHAUER, G. Setup and Open-Stacks Minimization in One-Dimensional Stock Cutting. **INFORMS Journal on Computing**, [Ottawa], v. 19, p.27-35, 2007.

BICHOFF, E. E.; WÄSCHER, G. Cutting and Packing. Editorial. **European Journal of Operational Research**, v. 84, p. 503-505, 1995.

BINKLEY, K. J.; HAGIWARA, M. Applying self-adaptive evolutionary algorithms to two-dimensional packing problems using a four corner's heuristic. **European Journal of Operational Research**, Amsterdam, v. 183, p. 1230-1248, 2007.

BROWN, A. R. Optimum Packing and Depletion: The Computer in Space – and Resource – Usage Problems. London: MacDonald & Co., 1971. 107 p.

CHRISTOFIDES, N.; HADJICONSTANTINO, E. An exact algorithm for orthogonal 2-D cutting problems using guillotine cuts. **European Journal of Operational Research**, Amsterdam, v. 83, p. 21-38, 1995.

COSTA, M. T.; GOMES, A. M.; OLIVEIRA, J. F. Heuristic approaches to large-scale periodic packing of irregular shapes on a rectangular sheet. **European Journal of Operational Research**, Amsterdam, v. 192, p. 29-40, 2009.

DAZA, V. P.; ALVARENGA, A. G.; DIEGO, J. Exact solutions for constrained two-dimensional cutting problems. **European Journal of Operational Research**, Amsterdam, v. 84, p. 633-644, 1995.

DOWNSLAND, K. A.; DOWNSLAND, W. B. Packing Problems. **European Journal of Operational Research**, Amsterdam, v. 56, p.2-14, 1992.

DOWNSLAND, K. A.; DOWNSLAND, W. B. Solution approaches to irregular nesting problems. **European Journal of Operational Research**, Amsterdam, v. 84, p.506-521, 1995.

DYCKHOFF, H. A typology of cutting and packing problems. **European Journal of Operational Research**, Amsterdam, v. 44, p. 145–159, 1990.

Empresa Brasileira de Aeronáutica – Embraer. São José dos Campos: Embraer S.A., 2009. Apresenta dados sobre a empresa e seus produtos. Disponível em: <<http://www.embraer.com.br/portugues/content/home>>. Acesso em: 30 abr. 2009.

FAGGIOLI, E.; BENTIVOGLIO, C. A. Heuristic and exact methods for the cutting sequencing problem. **European Journal of Operational Research**, Amsterdam, v. 110, p. 564-575, 1998.

FINK, A.; VOß, S. Applications of modern heuristic search methods to pattern sequencing problems. **Computers & Operations Research**, Oxford, v.26, p. 17-34, 1999.

FOERSTER, H.; WÄSCHER, G. Simulated annealing for order spread minimization in sequencing cutting patterns. **European Journal of Operational Research**, Amsterdam, v. 110, p. 272-281, 1998.

GILMORE, P. C.; GOMORY, R. E. A linear programming approach to the cutting stock problem. **Operations Research**, Linthicum, v.9, p. 849-859, 1961.

GILMORE, P. C.; GOMORY, R. E. A linear programming approach to the cutting-stock problem – Part II, **Operations Research**, Linthicum, v. 11, p. 863–888, 1963.

GRAHAM, R. L.; LAWLER, E. L.; LENSTRA, J. K.; KAN, A. H. G. Optimization and approximation in deterministic sequencing and scheduling: A survey. **Annals of Discrete Mathematics**, Amsterdam, v.5, p. 287-326, 1979.

HADJICONSTANTINO, E.; CHRISTOFIDES, N. An exact algorithm for general, orthogonal, two-dimensional knapsack problems. **European Journal of Operational Research**, Amsterdam, v. 83, p.39-56, 1995.

HAESSLER, R. W.; SWEENEY, P. E. Cutting stock problems and solution procedures. **European Journal of Operational Research**, Amsterdam, v. 54, p.141-150, 1991.

HERRMANN, J. W. **Handbook of production scheduling**. NewYork: Springer, 2006. 318 p.

HIFI, M. An improvement of Viswanathan and Bagchi's exact algorithm for constrained two-dimensional cutting stock. **Computers & Operations Research**, Oxford, v. 24, p. 727-736, 1997a.

HIFI, M. The DH/KD algorithm: a hybrid approach for unconstrained two-dimensional cutting problems. **European Journal of Operational Research**, Amsterdam, v. 97, p. 41-52, 1997b.

HIFI, M. Approximate and Exact Algorithms for Constrained (Un) Weighted Two-dimensional Two-staged Cutting Stock Problems. **Journal of Combinatorial Optimization**, New York, v. 5, p. 465-494, 2001.

JAYAMOHAN, M. S.; RAJENDRAN, C. Development and analysis of cost-based dispatching rules for job shop scheduling. **European Journal of Operational Research**, Amsterdam, v. 157, p. 307-321, 2004.

KANTOROVICH, L. V. Mathematical methods of organizing and planning production. **Management Science**, Rhode Island, v. 6, p. 366-422, 1960.

LENSTRA, J. K.; KAN, A. H. G. R. New Directions in Scheduling Theory. **Operations Research Letters**, Amsterdam, v. 2, n. 6, p. 255-259, 1984.

LINHARES, A.; YANASSE, H. H. Connections between cutting-pattern sequencing, VLSI design, and flexible machines. **Computers & Operations Research**, Oxford, v. 29, p. 1759-1772, 2002.

LODI, A.; MONACI, M. Integer linear programming models for 2-staged two-dimensional Knapsack problems. **Mathematical Programming**, Englewood Cliffs, v. 94, p. 254-278, 2003.

LODI, A.; MARTELLO, S.; VIGO, D. et al. Approximation algorithms for the oriented two-dimensional bin packing problem. **European Journal of Operational Research**, Amsterdam, v. 112, p. 379-396, 1999.

LODI, A.; MARTELLO, S.; VIGO, D. et al. Heuristic and Metaheuristic Approaches for a class of two-dimensional bin packing problems. **INFORMS Journal on Computing**, [Ottawa], v.11, p. 345-357, 1999.

LODI, A.; MARTELLO, S.; MONACI, M. Two-dimensional packing problems: A survey. **European Journal of Operational Research**, Amsterdam, v. 141, p. 241-252, 2002.

LODI, A.; MARTELLO, S.; VIGO, D. Approximation algorithms for the oriented two-dimensional bin packing problem. **European Journal of Operational Research**, Amsterdam, v. 112, p. 158-166, 1999a.

LODI, A.; MARTELLO, S.; VIGO, D. Heuristic and Metaheuristic Approaches for a Class of Two-Dimensional Bin Packing Problems. **INFORMS Journal on Computing**, [Ottawa], v. 11, p. 345-357, 1999b.

LODI, A.; MARTELLO, S.; VIGO, D. Recent advances on two-dimensional bin packing problems. **Discrete Applied Mathematics**, Amsterdam, v. 123, p. 379-396, 2002.

LODI, A.; MARTELLO, S.; MONACI, M. Two-dimensional packing problems: A survey. **European Journal of Operational Research**, Amsterdam, v. 141, p. 241-252, 2002.

LODI, A.; MARTELLO, S.; VIGO, D. Models and Bounds for Two-Dimensional Level Packing Problems. **Journal of Combinatorial Optimization**, New York, v. 8, p. 363-379, 2004.

MADSEN, O. B. G.; 1988. An application of travelling-salesman routines to solve pattern-allocation problems in the glass industry. **Journal of the Operational Research Society**, [Hampshire], v. 39, p. 249-256, 1988.

MARTELLO, S.; VIGO, D. Exact solution of the two-dimensional finite bin packing problem. **Management Science**, Oxford, v. 44, p. 388-399, 1998.

MONTGOMERY, D. C.; RUNGER, G. C. **Estatística aplicada e probabilidade para engenheiros**. 2^a ed. Rio de Janeiro: LTC, 2003. 463 p.

MORABITO, R.; ARENALES, M. N. An AND/OR-graph approach to the container packing problem. **International Transactions in Operational Research**, [New York], v. 1, p. 59-73, 1992.

MORABITO, R. Modelos de otimização para o problema de corte nas indústrias de papel e papelão e de móveis. **Gestão & Produção**, São Carlos, v.1, n.1, p. 59-76, abr. 1994.

MORABITO, R.; ARENALES, M. N. Staged and constrained two-dimensional guillotine cutting problems: An AND/OR-graph approach. **European Journal of Operational Research**, Amsterdam, v. 94, p. 548-560, 1996.

OLIVEIRA; FERREIRA. An improved version of Wang's algorithm for two-dimensional cutting problems. **European Journal of Operational Research**, Amsterdam, v. 44, p. 256-266, 1990.

OVACIK, I. M.; UZSOY, R. **Decomposition methods for complex factory scheduling problems**. Norwell: Kluwer Academic Publishers, 1997. 232 p.

PILEGGI, G. C. F.; MORABITO, R.; ARENALES, M. N. Abordagens para otimização integrada dos problemas de geração e sequenciamento de padrões de corte: caso unidimensional. **Pesquisa Operacional**, Rio de Janeiro, v. 25, p. 417-447, 2005.

PILEGGI, G. C. F.; MORABITO, R.; ARENALES, M. N. Heurísticas para os problemas de geração e sequenciamento de padrões de corte bidimensionais. **Pesquisa Operacional**, Rio de Janeiro, v. 27, p. 549-568, 2007.

PINEDO, M. **Scheduling: Theory, algorithms and systems**. New Jersey: Prentice Hall, 1995.

POLDI, K. C.; ARENALES, M. N. Heurística para o problema de corte de estoque unidimensional inteiro. **Pesquisa Operacional**, Rio de Janeiro, v.26, n.3, p. 473-492, 2006.

RIEHME, J.; SCHEITHAUER, G.; TERNO, J. The solution of two-stage guillotine cutting stock problems having extremely varying order demands. **European Journal of Operational Research**, Amsterdam, v. 91, p. 543-552, 1996.

PACHECO, R. F.; SANTORO, M. C. Proposta de Classificação Hierarquizada dos Modelos de Solução para o Problema de Job Shop Scheduling. **Gestão & Produção**, São Carlos, v. 6, n. 1, p. 1-15, 1999.

SPERANZA, M. G.; WOERLEE, A. P. A Decision Support System for operational production scheduling. **European Journal of Operational Research**, Amsterdam, v. 55, p. 329-343, 1991.

VISWANATHAN, K. V.; BAGCHI, A. Best-first search methods for constrained two-dimensional cutting stock problems. **Operations Research**, Linthicum, v. 41, n. 4, p. 768-776, 1993.

WANG, P. Y. Two algorithms for constrained two-dimensional cutting stock problems. **Operations Research**, Linthicum, v. 31, p. 573-586, 1983.

WÄSCHER, G.; HAUßNER, H.; SCHUMANN, H. An improved typology of cutting and packing problems. **European Journal of Operational Research**, Amsterdam, v. 183, p. 1109-1130, 2007.

WIGHT, O. W. **Production and Inventory Management in the Computer Age**. New York: Van Nostrand Reinhold Company, Inc., 1984.

WU, Y. L.; HUANG, W.; LAU, S. C.; WONG, C. K.; YOUNG, G. H. An effective quase-human based heuristic for solving the rectangle packing problem. **European Journal of Operational Research**, Amsterdam, v. 141, p. 341-358, 2002.

YANASSE, H. H.; LAMOSA, M. J. P. An integrated cutting stock and sequencing problem. **European Journal of Operational Research**, Amsterdam, v. 183, p. 1353-1370, 2007.

YANASSE, H. H. On a pattern sequencing problem to minimize the maximum number of open stacks. **European Journal of Operational Research**, Amsterdam, v. 100, p. 454-463, 1997.

YUEN, B. J. Heuristics for sequencing cutting patterns. **European Journal of Operational Research**, Amsterdam, v. 55, p. 183-190, 1991.

YUEN, B. J. Improved heuristics for sequencing cutting patterns. **European Journal of Operational Research**, Amsterdam, v. 87, p. 57-64, 1995.

APÊNDICES

APÊNDICE A - PROBABILIDADE LINEAR DE SORTEIO

Seja $P(i)$ a probabilidade do i -ésimo item da fila ser sorteado e α o valor pelo qual decrescente essa probabilidade linearmente. A relação (1) decorre da definição do viés linear, enquanto (2) e (3) são as condições de contorno relativas, respectivamente, à propriedade de soma unitária das probabilidades; e de que o item após o último (n -ésimo) da fila apresenta probabilidade nula de ser sorteada (já que não há item nesta posição).

$$P(i) = P(1) - (i - 1) \times \alpha \quad (1)$$

$$\sum_{i=1}^n P(i) = 1 \quad (2)$$

$$P(n + 1) = 0 \quad (3)$$

De (1) e (2), conclui-se (4).

$$\sum_{i=1}^n P(i) = \sum_{i=1}^n (P(1) - (i - 1) \times \alpha) = n \times P(1) - \alpha \sum_{i=1}^{n-1} i = 1$$

$$\therefore P(1) = \frac{1 + \alpha \sum_{i=1}^{n-1} i}{n} \quad (4)$$

De (1) e (3) decorre que:

$$P(n + 1) = P(1) - (n + 1 - 1) \times \alpha = 0$$

$$\therefore P(1) = n \times \alpha \quad (5)$$

Igualando (5) e (4):

$$P(1) = \frac{1 + \alpha \sum_{i=1}^{n-1} i}{n} = n \times \alpha \Rightarrow \alpha = \frac{1}{(n^2 - \sum_{i=1}^{n-1} i)} \Rightarrow P(1) = \frac{n}{(n^2 - \sum_{i=1}^{n-1} i)n}$$

Logo, voltando a (1), pode-se tirar a relação de probabilidade:

$$P(i) = \frac{n}{(n^2 - \sum_{i=1}^{n-1} i)n} - (i-1) \times \frac{1}{(n^2 - \sum_{i=1}^{n-1} i)} = \frac{n - (i-1)}{(n^2 - \sum_{i=1}^{n-1} i)}$$

APÊNDICE B: TABELAS COM RESULTADOS DOS EXPERIMENTOS PARA OS PROBLEMAS BIDIMENSIONAIS E NÃO-BIDIMENSIONAIS

TABELA B.1 – Resultados do critério de ocupação para o problema bidimensional guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	84,40%	87,46%	81,09%	82,75%
	2	85,39%	89,15%	86,07%	90,01%
	3	85,89%	88,37%	85,43%	86,25%
	4	86,73%	88,35%	86,12%	88,96%
	5	87,56%	89,36%	88,33%	88,88%
	6	87,15%	87,12%	90,26%	92,41%
	7	89,03%	89,13%	87,03%	91,30%
	8	83,45%	86,83%	79,53%	82,63%
	9	87,95%	89,61%	87,58%	90,26%
	10	86,06%	88,52%	84,85%	86,85%
<i>Fdin menor</i>	1	84,93%	86,44%	81,13%	81,80%
	2	84,91%	88,11%	89,17%	88,91%
	3	85,66%	87,61%	83,49%	84,87%
	4	85,33%	84,84%	85,37%	88,94%
	5	88,13%	89,12%	86,65%	90,18%
	6	87,65%	89,20%	89,07%	90,11%
	7	85,67%	88,99%	86,48%	89,72%
	8	85,38%	86,66%	80,51%	83,06%
	9	85,96%	89,02%	86,69%	91,79%
	10	85,16%	87,66%	84,87%	84,64%
<i>Fdin Ho</i>	1	84,46%	86,95%	79,45%	84,07%
	2	85,57%	88,36%	85,93%	87,14%
	3	86,27%	88,69%	83,39%	83,52%
	4	87,42%	89,00%	87,70%	89,27%
	5	88,45%	87,73%	86,18%	88,91%
	6	87,32%	90,26%	89,04%	91,06%
	7	89,19%	89,07%	89,43%	91,44%
	8	84,49%	86,68%	79,77%	82,18%
	9	84,36%	90,34%	86,30%	90,50%
	10	86,01%	88,88%	83,09%	85,13%
<i>Fdin maior</i>	1	85,73%	85,29%	82,27%	80,72%
	2	83,45%	89,51%	85,68%	90,53%
	3	84,95%	87,99%	83,70%	82,60%
	4	85,99%	87,42%	86,10%	88,45%
	5	87,36%	87,58%	85,74%	88,79%
	6	85,40%	89,28%	89,96%	91,16%
	7	90,13%	88,02%	87,44%	89,98%
	8	85,17%	87,62%	81,65%	83,50%
	9	84,56%	87,21%	87,97%	88,65%
	10	84,57%	88,07%	83,59%	81,32%

TABELA B.2 – Resultados do critério de ocupação para o problema bidimensional guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	84,67%	87,63%	82,99%	87,51%
	2	86,58%	90,14%	90,69%	91,79%
	3	86,77%	89,14%	89,54%	90,65%
	4	87,94%	88,91%	88,98%	90,07%
	5	88,04%	88,85%	87,49%	91,73%
	6	57,13%	90,96%	90,73%	93,17%
	7	88,24%	90,55%	90,34%	92,17%
	8	86,13%	87,66%	85,46%	85,46%
	9	87,34%	90,51%	90,04%	93,16%
	10	85,99%	89,97%	89,33%	90,11%
<i>Fdin menor</i>	1	84,77%	87,04%	84,27%	86,70%
	2	87,66%	90,86%	90,17%	91,32%
	3	86,28%	89,57%	88,96%	89,47%
	4	88,86%	89,12%	89,26%	91,29%
	5	89,21%	88,88%	87,59%	92,11%
	6	89,66%	90,26%	90,21%	93,20%
	7	89,04%	89,16%	90,85%	92,71%
	8	83,83%	86,71%	83,32%	86,94%
	9	88,08%	90,95%	88,90%	92,34%
	10	87,38%	90,01%	89,10%	91,34%
<i>Fdin Ho</i>	1	85,55%	86,57%	82,65%	84,87%
	2	85,13%	91,09%	89,52%	92,44%
	3	86,76%	88,81%	86,99%	89,64%
	4	86,10%	88,96%	88,21%	91,49%
	5	87,50%	87,22%	89,73%	91,47%
	6	87,98%	90,11%	90,53%	92,99%
	7	87,05%	90,09%	89,55%	93,93%
	8	85,23%	87,27%	84,52%	85,05%
	9	86,85%	90,40%	90,88%	90,70%
	10	87,36%	89,10%	87,21%	89,06%
<i>Fdin maior</i>	1	85,09%	86,60%	82,41%	83,61%
	2	86,18%	90,84%	90,34%	92,21%
	3	85,89%	89,49%	88,44%	88,86%
	4	89,60%	88,37%	88,21%	92,04%
	5	89,18%	88,58%	88,44%	92,17%
	6	88,16%	90,82%	90,44%	92,86%
	7	88,25%	89,73%	90,94%	92,57%
	8	84,82%	87,90%	81,47%	84,95%
	9	87,18%	90,41%	89,58%	92,99%
	10	86,15%	89,65%	89,03%	88,74%

TABELA B.3 – Resultados do critério de ocupação para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	86,42%	87,27%	86,33%	88,40%
	2	88,67%	91,66%	90,59%	93,29%
	3	87,22%	90,25%	91,53%	92,64%
	4	89,01%	89,55%	89,57%	91,88%
	5	87,85%	89,03%	89,46%	92,95%
	6	88,82%	91,36%	92,16%	94,24%
	7	90,77%	89,77%	92,70%	94,05%
	8	85,15%	87,16%	88,25%	88,75%
	9	87,98%	89,10%	91,27%	93,26%
	10	87,29%	89,76%	92,06%	92,80%
<i>Fdin menor</i>	1	85,44%	87,50%	86,77%	87,81%
	2	88,09%	90,39%	91,42%	92,02%
	3	86,87%	90,59%	91,66%	92,83%
	4	88,15%	89,18%	89,96%	93,06%
	5	88,93%	88,52%	90,48%	92,44%
	6	89,71%	91,95%	92,38%	93,77%
	7	89,61%	89,93%	91,89%	94,81%
	8	86,41%	87,57%	86,51%	87,69%
	9	87,76%	91,37%	91,61%	93,00%
	10	88,64%	90,02%	90,76%	92,84%
<i>Fdin Ho</i>	1	86,57%	87,15%	88,07%	87,66%
	2	88,34%	90,73%	90,77%	92,97%
	3	87,66%	90,54%	91,95%	93,16%
	4	87,76%	89,44%	88,72%	93,10%
	5	87,10%	89,73%	90,60%	93,40%
	6	89,26%	91,56%	92,14%	93,69%
	7	89,22%	90,74%	90,76%	93,92%
	8	85,07%	87,60%	85,38%	87,86%
	9	87,46%	91,28%	92,01%	93,24%
	10	88,03%	89,84%	91,14%	93,20%
<i>Fdin maior</i>	1	85,30%	87,38%	86,12%	87,59%
	2	87,74%	90,60%	92,11%	93,20%
	3	88,62%	90,14%	92,49%	91,32%
	4	88,82%	87,63%	89,37%	92,84%
	5	87,19%	89,86%	89,41%	92,33%
	6	89,18%	91,38%	92,15%	93,89%
	7	90,13%	91,32%	92,52%	93,83%
	8	86,64%	86,68%	86,79%	87,60%
	9	88,54%	91,32%	91,09%	94,41%
	10	86,86%	90,02%	91,33%	93,35%

TABELA B.4 – Resultados do critério de atendimento de datas para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	3,63	3,61	3,67	3,71
	2	3,73	3,73	3,70	3,71
	3	3,64	3,64	3,74	3,73
	4	3,74	3,73	3,73	3,73
	5	3,74	3,73	3,71	3,69
	6	3,74	3,73	3,72	3,73
	7	3,73	3,74	3,73	3,73
	8	3,62	3,62	3,69	3,69
	9	3,74	3,73	3,70	3,70
	10	3,65	3,62	3,73	3,72
<i>Fdin menor</i>	1	5,82	5,85	6,09	5,96
	2	6,14	6,14	5,99	6,03
	3	5,78	5,79	6,15	6,12
	4	6,11	6,16	6,12	6,13
	5	6,18	6,14	6,10	6,00
	6	6,12	6,11	6,05	6,11
	7	6,16	6,19	6,10	6,06
	8	5,79	5,87	6,08	6,09
	9	6,15	6,12	6,04	6,04
	10	5,83	5,88	6,13	6,14
<i>Fdin Ho</i>	1	1,12	1,12	1,21	1,21
	2	1,23	1,24	1,21	1,21
	3	1,12	1,11	1,22	1,22
	4	1,24	1,24	1,22	1,23
	5	1,23	1,23	1,24	1,23
	6	1,24	1,24	1,21	1,22
	7	1,24	1,22	1,23	1,23
	8	1,12	1,12	1,22	1,21
	9	1,22	1,23	1,21	1,20
	10	1,13	1,12	1,21	1,21
<i>Fdin maior</i>	1	3,36	3,37	3,60	3,60
	2	3,59	3,67	3,55	3,57
	3	3,33	3,37	3,63	3,55
	4	3,68	3,64	3,61	3,57
	5	3,59	3,61	3,60	3,59
	6	3,67	3,65	3,57	3,57
	7	3,68	3,63	3,59	3,60
	8	3,37	3,35	3,63	3,57
	9	3,69	3,68	3,59	3,51
	10	3,31	3,31	3,57	3,60

TABELA B.5 – Resultados do critério de atendimento de datas para o problema bidimensional guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>	1	3,59	3,61	3,58	3,51
	2	3,73	3,73	3,68	3,60
	3	3,70	3,71	3,69	3,70
	4	3,73	3,73	3,68	3,70
	5	3,71	3,73	3,70	3,62
	6	3,73	3,74	3,71	3,69
	7	3,73	3,74	3,70	3,71
	8	3,62	3,62	3,54	3,54
	9	3,73	3,73	3,65	3,62
	10	3,73	3,72	3,68	3,66
<i>Fdin menor</i>	1	5,87	5,76	5,68	5,84
	2	6,06	6,10	5,95	5,97
	3	6,07	6,06	6,02	6,05
	4	6,17	6,11	6,03	6,08
	5	6,11	6,11	6,01	6,04
	6	6,18	6,15	5,91	6,00
	7	6,08	6,16	5,97	6,01
	8	5,80	5,81	5,59	5,53
	9	6,09	6,18	5,96	5,92
	10	6,02	6,06	6,06	6,02
<i>Fdin Ho</i>	1	1,13	1,12	1,15	1,14
	2	1,23	1,24	1,15	1,11
	3	1,18	1,18	1,22	1,21
	4	1,23	1,23	1,22	1,20
	5	1,23	1,24	1,20	1,21
	6	1,23	1,23	1,18	1,22
	7	1,22	1,24	1,21	1,21
	8	1,10	1,13	1,15	1,16
	9	1,23	1,23	1,17	1,17
	10	1,17	1,17	1,23	1,13
<i>Fdin maior</i>	1	3,31	3,34	3,37	3,50
	2	3,56	3,62	3,52	3,40
	3	3,46	3,41	3,46	3,52
	4	3,64	3,61	3,50	3,57
	5	3,59	3,64	3,56	3,55
	6	3,63	3,67	3,48	3,42
	7	3,61	3,66	3,58	3,62
	8	3,32	3,30	3,31	3,36
	9	3,60	3,49	3,26	3,25
	10	3,54	3,45	3,50	3,55

TABELA B.6 – Resultados do critério de atendimento de datas para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin Ho</i>		1	3,43	3,59	2,93	3,33
		2	3,71	3,65	3,61	3,61
		3	3,74	3,69	3,65	3,61
		4	3,72	3,73	3,66	3,65
		5	3,73	3,71	3,57	3,53
		6	3,73	3,72	3,52	3,64
		7	3,73	3,73	3,57	3,67
		8	3,35	3,53	2,98	2,76
		9	3,73	3,73	3,52	3,56
		10	3,72	3,73	3,58	3,47
<i>Fdin menor</i>		1	5,12	5,71	2,78	5,03
		2	6,07	6,12	5,62	5,75
		3	6,11	6,15	5,78	5,55
		4	6,12	6,14	5,91	5,80
		5	6,10	6,09	5,63	5,84
		6	6,13	6,09	5,81	5,90
		7	6,10	6,16	5,83	5,93
		8	5,43	5,82	4,57	5,19
		9	6,10	6,06	5,52	5,77
		10	6,12	6,15	5,65	5,80
<i>Fdin Ho</i>		1	1,04	1,11	0,75	0,84
		2	1,21	1,22	1,06	1,11
		3	1,22	1,24	1,11	1,07
		4	1,21	1,24	1,17	1,17
		5	1,22	1,24	1,13	1,11
		6	1,21	1,22	1,12	1,13
		7	1,22	1,22	1,15	1,16
		8	0,90	1,12	0,75	0,99
		9	1,22	1,23	1,00	1,09
		10	1,23	1,23	1,16	1,00
<i>Fdin maior</i>		1	2,76	3,28	2,30	2,95
		2	3,57	3,53	3,12	3,32
		3	3,59	3,57	3,32	3,27
		4	3,59	3,60	3,34	3,27
		5	3,55	3,63	3,42	3,29
		6	3,65	3,65	3,27	3,29
		7	3,62	3,62	3,21	3,41
		8	2,98	3,22	2,03	2,52
		9	3,58	3,59	3,11	2,82
		10	3,60	3,59	3,38	3,17

TABELA B.7 – Resultados do critério ponderado para o problema bidimensional guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	10,67%	10,06%	24,64%	53,75%
		2	79,08%	85,94%	47,56%	65,25%
		3	20,49%	30,70%	87,68%	68,46%
		4	84,11%	79,72%	77,53%	76,22%
		5	87,91%	87,56%	63,92%	37,61%
		6	91,10%	81,56%	72,02%	94,23%
		7	88,26%	91,43%	81,85%	86,35%
		8	5,05%	13,67%	31,23%	37,43%
		9	88,54%	88,11%	60,96%	50,25%
		10	26,56%	21,93%	78,22%	64,11%
<i>Fdin</i>	<i>He</i>	1	18,67%	25,61%	46,64%	29,89%
		2	74,06%	80,39%	27,97%	61,37%
		3	16,64%	18,03%	73,82%	68,84%
		4	70,11%	75,24%	67,35%	75,85%
		5	89,80%	82,56%	62,55%	47,57%
		6	77,15%	77,92%	53,27%	70,38%
		7	76,96%	91,26%	61,30%	57,67%
		8	16,78%	28,39%	42,71%	54,39%
		9	80,42%	77,94%	38,70%	61,40%
		10	21,44%	32,55%	68,07%	73,57%
<i>Fdin</i>	<i>Ho</i>	1	8,24%	14,58%	28,64%	34,97%
		2	80,27%	86,53%	32,62%	43,48%
		3	14,35%	15,89%	40,46%	53,82%
		4	85,90%	88,52%	62,56%	88,72%
		5	84,79%	77,68%	90,25%	75,57%
		6	90,70%	91,47%	45,98%	77,71%
		7	91,78%	77,13%	84,67%	82,66%
		8	12,73%	12,04%	39,08%	33,00%
		9	69,33%	87,54%	35,13%	41,58%
		10	18,01%	19,30%	34,85%	44,17%
<i>Fdin</i>	<i>He</i>	1	24,84%	19,23%	58,25%	62,09%
		2	65,01%	89,21%	40,99%	62,60%
		3	19,22%	28,68%	71,73%	47,80%
		4	85,73%	79,71%	69,29%	60,48%
		5	72,54%	71,74%	63,37%	71,12%
		6	81,23%	83,89%	69,60%	67,85%
		7	94,06%	78,29%	57,74%	73,89%
		8	25,21%	22,03%	60,86%	52,51%
		9	86,12%	85,37%	61,89%	48,10%
		10	14,97%	15,10%	53,61%	63,26%

TABELA B.8 – Resultados do critério ponderado para o problema bidimensional guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	8,38%	6,14%	15,92%	29,59%
		2	67,99%	73,50%	76,46%	70,28%
		3	58,98%	57,31%	70,87%	74,84%
		4	79,96%	65,13%	70,45%	70,79%
		5	72,24%	64,58%	66,81%	71,84%
		6	74,79%	83,00%	89,23%	88,07%
		7	82,42%	84,38%	82,95%	82,51%
		8	27,86%	7,68%	28,47%	26,57%
		9	77,44%	81,21%	69,86%	76,06%
		10	60,98%	70,68%	71,93%	62,37%
<i>Fdin</i>	<i>He</i>	1	18,49%	8,89%	23,69%	34,94%
		2	61,62%	82,01%	81,35%	69,74%
		3	53,54%	64,90%	81,34%	59,85%
		4	85,76%	63,80%	82,40%	77,87%
		5	78,99%	61,12%	72,54%	81,72%
		6	90,60%	81,68%	74,84%	85,48%
		7	76,82%	71,57%	85,36%	78,10%
		8	7,16%	11,03%	24,10%	7,36%
		9	70,95%	90,41%	72,33%	72,15%
		10	56,12%	67,80%	85,62%	72,15%
<i>Fdin</i>	<i>Ho</i>	1	25,37%	9,07%	21,53%	36,92%
		2	59,15%	89,08%	50,01%	56,71%
		3	52,94%	49,51%	62,89%	66,87%
		4	69,53%	68,02%	66,48%	76,42%
		5	76,29%	59,81%	72,37%	79,47%
		6	79,57%	85,01%	66,78%	90,01%
		7	68,12%	83,43%	71,67%	91,29%
		8	16,47%	15,57%	33,79%	35,79%
		9	73,42%	80,11%	63,60%	62,62%
		10	56,84%	48,61%	69,60%	45,17%
<i>Fdin</i>	<i>He</i>	1	15,01%	10,70%	36,43%	36,60%
		2	55,45%	84,33%	79,62%	69,14%
		3	36,60%	49,15%	65,29%	63,47%
		4	85,29%	58,22%	70,90%	83,93%
		5	80,34%	63,70%	76,25%	81,82%
		6	73,54%	89,23%	78,27%	69,84%
		7	74,09%	77,10%	89,65%	90,28%
		8	15,41%	18,95%	21,19%	26,49%
		9	62,28%	63,55%	52,82%	54,35%
		10	49,37%	54,17%	75,73%	62,47%

TABELA B.9 – Resultados do critério ponderado para o problema bidimensional guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	25,71%	21,84%	11,30%	22,75%
		2	65,29%	84,58%	70,41%	81,05%
		3	47,04%	67,41%	79,84%	71,27%
		4	69,91%	60,86%	60,30%	65,14%
		5	57,91%	53,37%	54,71%	70,42%
		6	69,54%	86,41%	85,55%	89,62%
		7	92,15%	66,28%	90,70%	89,13%
		8	9,20%	17,19%	33,08%	14,95%
		9	61,72%	56,77%	77,35%	78,45%
		10	48,53%	64,24%	85,06%	67,23%
<i>Fdin</i>	<i>He</i>	1	10,00%	9,17%	11,44%	10,62%
		2	60,44%	66,81%	79,52%	66,02%
		3	42,70%	70,70%	84,63%	71,78%
		4	64,17%	50,73%	65,46%	77,55%
		5	72,02%	38,59%	68,96%	72,02%
		6	85,45%	84,86%	91,88%	85,49%
		7	79,12%	61,84%	86,03%	96,65%
		8	25,21%	13,97%	18,15%	14,14%
		9	54,00%	74,12%	80,45%	76,43%
		10	68,65%	62,97%	68,45%	75,33%
<i>Fdin</i>	<i>Ho</i>	1	38,78%	4,05%	29,65%	12,92%
		2	72,07%	75,54%	70,70%	77,32%
		3	64,04%	76,76%	87,96%	76,29%
		4	63,27%	57,97%	51,73%	81,41%
		5	60,84%	64,42%	74,40%	82,18%
		6	85,43%	90,19%	88,37%	86,05%
		7	82,86%	77,09%	74,05%	88,97%
		8	12,19%	12,59%	4,97%	16,61%
		9	59,90%	86,36%	83,05%	78,84%
		10	68,43%	63,72%	80,44%	73,06%
<i>Fdin</i>	<i>He</i>	1	3,19%	22,13%	14,67%	17,41%
		2	50,86%	72,85%	84,10%	77,16%
		3	65,18%	66,45%	86,37%	55,45%
		4	66,49%	39,52%	56,39%	71,48%
		5	43,68%	64,52%	56,59%	66,65%
		6	73,68%	83,95%	83,22%	82,76%
		7	88,01%	83,19%	86,17%	84,12%
		8	25,11%	9,08%	14,08%	10,56%
		9	61,44%	77,92%	72,76%	78,79%
		10	40,57%	62,89%	74,76%	75,06%

TABELA B.10 – Resultados do critério de ocupação para o problema bidimensional não- guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	85,64%	88,57%	82,67%	86,16%
		2	87,65%	91,21%	86,19%	91,65%
		3	86,01%	89,19%	85,63%	87,30%
		4	87,68%	91,35%	87,14%	92,09%
		5	89,18%	91,14%	87,39%	91,47%
		6	89,10%	92,93%	90,04%	93,35%
		7	88,49%	92,01%	87,77%	91,57%
		8	85,21%	89,28%	81,88%	84,08%
		9	86,83%	88,97%	88,27%	92,67%
		10	85,51%	90,61%	85,76%	86,90%
<i>Fdin</i>	<i>He</i>	1	86,01%	89,00%	81,73%	85,24%
		2	78,87%	83,13%	84,70%	89,73%
		3	86,18%	90,29%	83,89%	84,63%
		4	87,57%	91,15%	87,23%	91,42%
		5	89,34%	91,63%	88,24%	91,83%
		6	88,65%	92,91%	90,75%	92,71%
		7	88,30%	92,06%	86,34%	91,70%
		8	85,45%	89,02%	80,52%	84,25%
		9	80,22%	88,61%	82,50%	90,88%
		10	85,28%	89,38%	85,45%	86,12%
<i>Fdin</i>	<i>Ho</i>	1	84,59%	88,85%	82,84%	85,58%
		2	85,77%	92,06%	86,49%	91,36%
		3	85,79%	89,53%	84,29%	85,67%
		4	88,14%	91,78%	86,81%	90,83%
		5	89,98%	91,72%	87,33%	89,60%
		6	88,59%	92,85%	88,86%	92,01%
		7	89,20%	92,69%	89,27%	92,34%
		8	85,28%	88,87%	82,12%	84,35%
		9	85,61%	91,02%	85,52%	92,69%
		10	87,05%	90,11%	84,63%	85,66%
<i>Fdin</i>	<i>He</i>	1	84,62%	88,52%	82,71%	83,58%
		2	85,31%	92,96%	88,07%	90,62%
		3	85,90%	89,44%	83,49%	86,95%
		4	88,82%	91,07%	87,51%	90,96%
		5	88,52%	91,81%	88,12%	91,84%
		6	89,86%	92,22%	90,15%	92,23%
		7	88,85%	93,09%	87,79%	92,21%
		8	85,38%	89,46%	81,31%	80,91%
		9	82,07%	90,15%	88,16%	90,68%
		10	84,72%	90,44%	84,32%	88,17%

TABELA B.11 – Resultados do critério de ocupação para o problema bidimensional não-guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	86,45%	87,79%	84,90%	88,70%
		2	87,25%	92,13%	92,60%	92,92%
		3	87,93%	90,82%	90,94%	92,02%
		4	89,12%	91,49%	87,85%	92,59%
		5	88,32%	91,92%	90,17%	93,36%
		6	88,26%	93,39%	92,77%	94,17%
		7	89,55%	92,65%	91,43%	94,48%
		8	86,14%	89,32%	84,49%	89,37%
		9	87,30%	92,38%	91,20%	92,98%
		10	86,32%	90,21%	90,30%	92,32%
<i>Fdin</i>	<i>He</i>	1	85,82%	88,62%	83,95%	88,86%
		2	88,40%	91,99%	89,79%	93,67%
		3	87,15%	91,26%	88,67%	91,24%
		4	88,97%	91,87%	88,81%	93,53%
		5	89,59%	91,75%	89,06%	93,33%
		6	90,07%	93,39%	92,45%	94,35%
		7	90,04%	93,35%	90,34%	93,55%
		8	86,11%	88,82%	83,61%	87,54%
		9	87,89%	92,10%	89,39%	94,29%
		10	86,97%	91,31%	89,43%	92,42%
<i>Fdin</i>	<i>Ho</i>	1	84,88%	88,44%	83,51%	86,15%
		2	87,93%	91,95%	91,84%	93,46%
		3	86,34%	90,86%	88,51%	89,38%
		4	88,45%	91,63%	88,29%	92,77%
		5	89,03%	91,70%	90,39%	92,44%
		6	89,94%	93,47%	90,51%	94,73%
		7	90,08%	93,45%	92,75%	93,65%
		8	85,62%	88,37%	82,58%	85,62%
		9	87,35%	91,61%	89,83%	93,15%
		10	86,95%	91,29%	89,28%	89,79%
<i>Fdin</i>	<i>He</i>	1	84,83%	88,98%	84,40%	85,35%
		2	87,68%	91,57%	92,44%	93,50%
		3	87,72%	90,49%	88,86%	91,87%
		4	88,46%	91,81%	88,02%	93,20%
		5	89,36%	92,34%	87,47%	93,63%
		6	89,55%	92,50%	90,68%	94,84%
		7	90,85%	92,08%	90,94%	94,24%
		8	85,55%	88,86%	82,42%	86,76%
		9	86,85%	92,02%	90,82%	92,71%
		10	86,35%	90,75%	90,34%	91,86%

TABELA B.12 – Resultados do critério de ocupação para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	85,73%	88,80%	86,31%	91,24%
		2	88,18%	93,14%	92,23%	95,61%
		3	87,74%	92,10%	92,22%	94,67%
		4	89,23%	91,79%	90,05%	93,70%
		5	89,42%	92,11%	89,97%	93,05%
		6	89,21%	93,79%	92,67%	95,03%
		7	90,86%	93,63%	92,64%	94,94%
		8	86,58%	89,99%	86,95%	89,87%
		9	88,98%	92,88%	91,75%	94,65%
		10	88,15%	91,70%	91,78%	94,78%
<i>Fdin</i> <i>menor</i>		1	85,97%	88,74%	86,19%	88,95%
		2	89,00%	92,34%	91,75%	94,54%
		3	87,48%	92,48%	94,29%	93,64%
		4	89,68%	92,08%	89,72%	94,79%
		5	89,21%	91,77%	91,21%	93,91%
		6	88,66%	93,10%	93,89%	95,05%
		7	90,51%	93,85%	92,81%	94,98%
		8	86,18%	88,72%	86,24%	90,64%
		9	87,69%	92,74%	91,90%	95,73%
		10	87,29%	92,34%	91,85%	94,56%
<i>Fdin</i> <i>maior</i>		1	86,12%	89,41%	87,29%	90,21%
		2	88,55%	92,62%	90,90%	95,22%
		3	86,29%	92,24%	91,43%	94,72%
		4	88,05%	91,80%	90,98%	93,72%
		5	89,90%	92,13%	89,95%	93,44%
		6	89,49%	93,80%	92,88%	95,68%
		7	91,55%	93,22%	92,66%	94,87%
		8	86,20%	89,06%	84,30%	89,66%
		9	87,90%	92,62%	91,33%	93,76%
		10	87,32%	92,33%	91,52%	95,17%
<i>Fdin</i> <i>He</i>		1	86,32%	89,59%	85,17%	89,54%
		2	88,01%	93,00%	92,46%	94,83%
		3	87,73%	91,96%	92,00%	94,65%
		4	89,98%	92,40%	90,46%	93,80%
		5	89,02%	92,20%	91,48%	93,68%
		6	90,53%	93,22%	93,48%	95,06%
		7	90,80%	93,17%	92,44%	94,53%
		8	86,25%	89,10%	86,96%	90,71%
		9	88,21%	92,93%	93,08%	95,39%
		10	86,83%	92,47%	92,19%	93,95%

TABELA B.13 – Resultados do critério de atendimento de datas para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	1	3,61	3,61	3,71	3,71
	2	3,74	3,74	3,69	3,71
	3	3,63	3,64	3,72	3,70
	4	3,74	3,73	3,73	3,72
	5	3,72	3,72	3,71	3,72
	6	3,74	3,74	3,71	3,71
	7	3,73	3,73	3,73	3,72
	8	3,61	3,61	3,70	3,70
	9	3,74	3,74	3,69	3,70
	10	3,62	3,61	3,73	3,70
<i>Fdin menor</i>	1	5,82	5,83	6,11	6,07
	2	6,15	6,19	6,04	6,06
	3	5,86	5,86	6,13	6,09
	4	6,17	6,17	6,09	6,11
	5	6,14	6,12	6,15	6,12
	6	6,17	6,17	6,06	6,10
	7	6,09	6,17	6,12	6,05
	8	5,78	5,82	6,08	6,07
	9	6,15	6,18	5,99	6,05
	10	5,79	5,83	6,13	6,06
<i>Fdin Ho</i>	1	1,12	1,12	1,21	1,21
	2	1,24	1,24	1,20	1,20
	3	1,10	1,12	1,21	1,21
	4	1,24	1,24	1,22	1,23
	5	1,24	1,22	1,23	1,21
	6	1,25	1,24	1,22	1,22
	7	1,24	1,22	1,23	1,23
	8	1,11	1,10	1,21	1,20
	9	1,24	1,24	1,20	1,21
	10	1,12	1,12	1,22	1,21
<i>Fdin maior</i>	1	3,32	3,26	3,64	3,56
	2	3,67	3,61	3,54	3,55
	3	3,37	3,33	3,63	3,59
	4	3,64	3,71	3,64	3,61
	5	3,64	3,65	3,65	3,56
	6	3,65	3,65	3,59	3,63
	7	3,62	3,63	3,64	3,65
	8	3,32	3,33	3,58	3,58
	9	3,65	3,59	3,57	3,56
	10	3,35	3,36	3,60	3,63

TABELA B.14 – Resultados do critério de atendimento de datas para o problema bidimensional não-guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

	<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
		<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	1	3,60	3,62	3,56	3,63
	2	3,72	3,72	3,61	3,66
	3	3,71	3,72	3,69	3,68
	4	3,73	3,73	3,68	3,69
	5	3,72	3,72	3,68	3,69
	6	3,73	3,73	3,68	3,70
	7	3,72	3,73	3,72	3,70
	8	3,51	3,63	3,54	3,47
	9	3,74	3,71	3,62	3,67
	10	3,73	3,71	3,70	3,70
<i>Fdin menor</i>	1	5,84	5,81	5,92	5,43
	2	6,15	6,11	5,82	5,89
	3	6,10	6,10	6,03	6,00
	4	6,12	6,16	6,02	6,08
	5	6,17	6,08	5,90	5,89
	6	6,18	6,08	5,96	5,99
	7	6,20	6,16	6,02	6,07
	8	5,72	5,77	5,76	5,21
	9	6,14	6,13	5,97	5,78
	10	6,10	6,05	5,99	6,02
<i>Fdin Ho</i>	1	1,12	1,11	1,11	1,13
	2	1,24	1,23	1,19	1,18
	3	1,18	1,17	1,23	1,21
	4	1,23	1,23	1,21	1,21
	5	1,22	1,23	1,21	1,18
	6	1,24	1,23	1,16	1,21
	7	1,24	1,23	1,21	1,20
	8	1,12	1,12	1,17	1,17
	9	1,23	1,24	1,16	1,20
	10	1,19	1,18	1,22	1,19
<i>Fdin maior</i>	1	3,31	3,24	3,38	3,55
	2	3,57	3,63	3,38	3,32
	3	3,51	3,48	3,63	3,48
	4	3,64	3,63	3,48	3,53
	5	3,60	3,65	3,59	3,50
	6	3,66	3,66	3,54	3,48
	7	3,64	3,56	3,58	3,54
	8	3,24	3,18	3,48	3,36
	9	3,58	3,60	3,44	3,39
	10	3,54	3,50	3,62	3,57

TABELA B.15 – Resultados do critério de atendimento de datas para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	3,36	3,51	3,24	2,93
		2	3,74	3,71	3,63	3,53
		3	3,73	3,74	3,57	3,63
		4	3,72	3,74	3,60	3,67
		5	3,72	3,74	3,66	3,65
		6	3,73	3,73	3,60	3,64
		7	3,73	3,73	3,64	3,70
		8	3,40	3,54	2,93	3,24
		9	3,72	3,72	3,54	3,57
		10	3,73	3,74	3,62	3,60
<i>Fdin</i> <i>menor</i>		1	5,54	5,55	4,86	4,94
		2	6,07	6,06	5,85	5,71
		3	6,02	6,11	5,76	5,88
		4	6,13	6,13	5,77	5,80
		5	6,10	6,12	5,87	5,64
		6	6,12	6,12	5,75	5,88
		7	6,11	6,15	5,87	5,84
		8	5,24	5,62	5,00	3,50
		9	6,01	6,17	5,66	5,92
		10	6,08	6,07	5,72	5,62
<i>Fdin</i> <i>maior</i>		1	1,10	1,13	0,74	0,76
		2	1,22	1,21	1,08	1,10
		3	1,23	1,23	1,14	1,11
		4	1,24	1,22	1,09	1,18
		5	1,23	1,23	1,12	1,17
		6	1,24	1,23	1,10	1,10
		7	1,24	1,23	1,16	1,17
		8	1,05	1,09	0,81	1,01
		9	1,23	1,24	1,11	1,08
		10	1,23	1,23	1,13	1,15
<i>Fdin</i> <i>He</i>		1	2,93	3,20	3,12	1,62
		2	3,54	3,55	3,21	3,29
		3	3,59	3,60	3,44	3,07
		4	3,60	3,66	3,30	3,43
		5	3,56	3,58	3,53	3,36
		6	3,60	3,64	3,42	3,37
		7	3,69	3,66	3,23	3,31
		8	2,93	3,34	2,29	2,20
		9	3,56	3,55	3,15	3,23
		10	3,58	3,61	3,42	3,26

TABELA B.16 – Resultados do critério ponderado para o problema bidimensional não-guilhotinado com peso de ocupação de 20% e peso de atendimento de datas de 80%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	12,92%	15,48%	54,24%	38,44%
		2	87,35%	87,63%	35,82%	61,48%
		3	22,88%	30,06%	69,83%	35,71%
		4	89,44%	86,61%	83,03%	72,76%
		5	84,06%	80,09%	64,56%	70,46%
		6	95,09%	93,84%	64,98%	62,21%
		7	85,57%	85,02%	78,78%	75,74%
		8	10,34%	18,64%	36,28%	31,72%
		9	86,90%	83,16%	43,21%	40,93%
		10	17,62%	23,19%	84,61%	38,54%
<i>Fdin</i> <i>menor</i>		1	25,80%	20,17%	63,93%	42,62%
		2	76,10%	83,27%	45,72%	50,06%
		3	32,08%	29,11%	72,44%	49,60%
		4	89,32%	86,32%	65,25%	64,02%
		5	87,39%	78,92%	87,98%	69,45%
		6	90,46%	90,92%	63,12%	63,82%
		7	74,71%	88,56%	75,01%	53,25%
		8	19,95%	17,91%	51,97%	41,31%
		9	78,40%	88,22%	30,93%	51,31%
		10	19,24%	20,16%	72,91%	42,29%
<i>Fdin</i> <i>maior</i>		1	15,19%	10,88%	48,43%	34,38%
		2	82,33%	89,64%	32,79%	34,17%
		3	7,59%	17,11%	35,82%	30,57%
		4	87,64%	92,59%	68,93%	81,37%
		5	94,34%	82,05%	78,09%	47,31%
		6	94,16%	94,79%	76,92%	74,24%
		7	90,86%	81,03%	77,17%	91,99%
		8	10,51%	4,06%	44,92%	29,19%
		9	80,85%	83,60%	30,40%	50,49%
		10	18,54%	18,81%	51,08%	35,72%
<i>Fdin</i> <i>He</i>		1	11,46%	6,92%	67,89%	43,60%
		2	83,81%	79,90%	43,57%	52,30%
		3	25,03%	20,74%	67,61%	58,69%
		4	84,40%	90,71%	79,66%	73,34%
		5	85,17%	84,41%	84,21%	59,12%
		6	88,26%	84,27%	65,69%	78,20%
		7	81,91%	84,56%	81,61%	90,65%
		8	12,89%	20,02%	41,04%	45,34%
		9	76,16%	67,33%	51,28%	50,91%
		10	17,34%	29,69%	55,37%	69,06%

TABELA B.17 – Resultados do critério ponderado para o problema bidimensional não-guilhotinado com peso de ocupação de 50% e peso de atendimento de datas de 50%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	28,01%	4,89%	25,11%	36,66%
		2	65,41%	79,63%	69,89%	67,26%
		3	69,67%	67,48%	78,21%	60,32%
		4	81,88%	77,00%	60,74%	72,12%
		5	73,99%	76,90%	71,70%	77,84%
		6	80,31%	94,01%	85,08%	87,41%
		7	83,49%	86,27%	85,50%	87,66%
		8	15,77%	19,22%	16,28%	20,74%
		9	70,18%	78,10%	64,59%	75,29%
		10	59,46%	57,20%	76,37%	73,15%
<i>Fdin</i>	<i>He</i>	1	17,13%	10,02%	31,31%	22,24%
		2	71,57%	74,33%	54,87%	77,75%
		3	54,63%	66,77%	66,05%	65,48%
		4	71,59%	77,88%	65,78%	84,31%
		5	81,05%	69,15%	60,08%	72,92%
		6	86,51%	81,52%	78,37%	90,23%
		7	87,73%	92,95%	77,97%	84,80%
		8	8,37%	7,14%	20,56%	22,96%
		9	62,96%	75,13%	69,22%	74,60%
		10	50,74%	63,33%	64,46%	77,14%
<i>Fdin</i>	<i>Ho</i>	1	7,40%	9,81%	19,12%	29,62%
		2	73,14%	78,21%	76,92%	67,50%
		3	38,97%	47,97%	74,67%	64,62%
		4	74,43%	75,06%	68,73%	80,42%
		5	75,35%	75,61%	76,68%	58,99%
		6	88,50%	91,57%	60,78%	86,16%
		7	88,01%	92,12%	89,41%	76,75%
		8	12,07%	12,85%	32,55%	28,86%
		9	65,82%	80,61%	57,50%	77,84%
		10	46,81%	56,85%	77,33%	58,93%
<i>Fdin</i>	<i>He</i>	1	12,95%	15,12%	24,95%	43,29%
		2	61,10%	74,74%	56,47%	55,98%
		3	54,88%	53,24%	77,54%	67,70%
		4	75,18%	76,49%	52,82%	73,57%
		5	75,13%	86,28%	61,32%	78,28%
		6	84,68%	81,67%	73,43%	81,17%
		7	90,77%	75,18%	79,20%	81,52%
		8	12,09%	13,07%	23,35%	33,15%
		9	56,48%	76,47%	64,96%	63,60%
		10	48,26%	55,61%	80,82%	73,52%

TABELA B.18 – Resultados do critério ponderado para o problema bidimensional não-guilhotinado com peso de ocupação de 80% e peso de atendimento de datas de 20%

		<i>Heurística</i>	<i>Dim Menor</i>		<i>Dim Maior</i>	
			<i>Dim Ho</i>	<i>Dim He</i>	<i>Dim Ho</i>	<i>Dim He</i>
<i>Fdin</i>	<i>Ho</i>	1	15,25%	6,38%	14,61%	22,05%
		2	59,56%	85,80%	82,89%	89,98%
		3	56,13%	70,14%	80,33%	78,89%
		4	70,76%	65,58%	62,87%	70,94%
		5	74,36%	71,12%	61,21%	57,64%
		6	67,76%	97,05%	85,51%	86,68%
		7	95,32%	95,69%	87,81%	83,48%
		8	22,58%	23,51%	13,52%	10,62%
		9	69,40%	81,32%	76,50%	79,75%
		10	56,31%	65,74%	79,19%	83,54%
<i>Fdin</i>	<i>He</i>	1	18,32%	12,34%	7,14%	11,39%
		2	69,28%	73,80%	69,81%	80,14%
		3	45,37%	78,74%	92,30%	70,85%
		4	82,51%	72,18%	50,68%	83,14%
		5	72,66%	68,52%	63,96%	72,65%
		6	68,94%	86,50%	86,33%	87,04%
		7	91,35%	98,24%	79,79%	85,62%
		8	15,54%	11,19%	9,16%	19,14%
		9	51,43%	83,76%	69,91%	95,13%
		10	44,64%	76,07%	71,50%	78,96%
<i>Fdin</i>	<i>Ho</i>	1	17,98%	12,81%	29,20%	18,70%
		2	58,45%	75,72%	68,45%	87,32%
		3	32,52%	72,64%	73,63%	83,30%
		4	54,65%	64,58%	68,09%	73,49%
		5	74,17%	70,91%	61,86%	68,95%
		6	66,54%	97,37%	83,26%	91,78%
		7	95,80%	88,66%	84,27%	85,78%
		8	15,60%	5,22%	5,51%	20,67%
		9	53,43%	79,55%	73,10%	68,50%
		10	43,32%	73,12%	72,74%	87,89%
<i>Fdin</i>	<i>He</i>	1	12,16%	9,47%	11,64%	2,79%
		2	48,23%	80,29%	81,81%	83,48%
		3	47,60%	60,45%	74,96%	80,15%
		4	76,93%	73,38%	60,94%	71,93%
		5	65,52%	66,11%	74,54%	70,59%
		6	84,61%	83,90%	92,65%	90,70%
		7	91,01%	86,29%	81,49%	83,69%
		8	11,85%	6,33%	15,94%	20,24%
		9	51,75%	78,93%	86,18%	91,80%
		10	31,60%	73,71%	78,80%	74,55%

APÊNDICE C – RESULTADOS DOS TESTES DE LEVENE E BARTLETT PARA HOMOCEDESTICIDADE

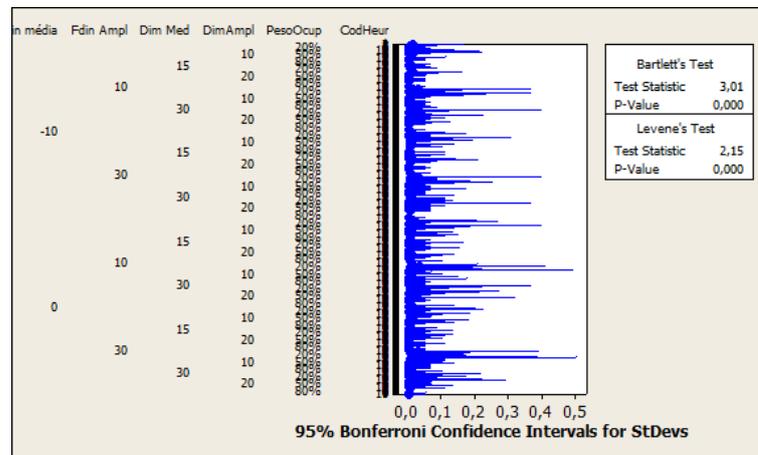


Figura C.1 – Teste de homocedasticidade para critério de ocupação

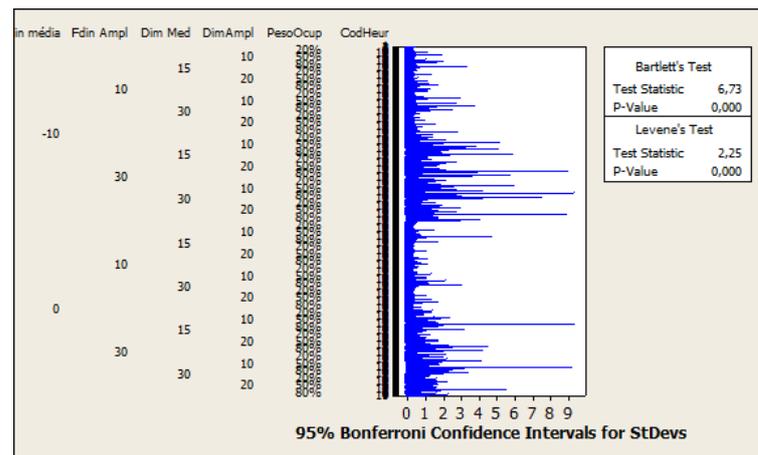


Figura C.2 – Teste de homocedasticidade para critério de atendimento de datas

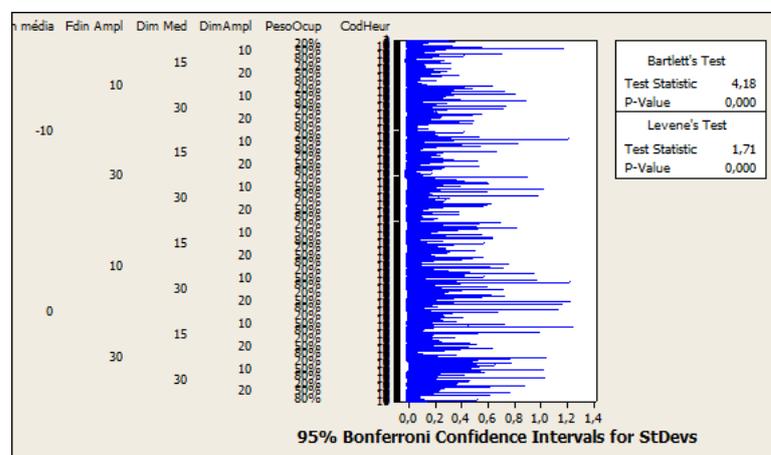


Figura C.3 – Teste de homocedasticidade para critério ponderado

APÊNDICE D – RESULTADOS DOS TESTES DE KRUSKAL-WALLIS

Kruskal-Wallis Test: Ocupação versus CodHeur				
Kruskal-Wallis Test on Ocupação				
CodHeur	N	Median	Ave Rank	Z
1	240	0,8578	527,0	-15,87
2	240	0,8979	1411,2	4,96
3	240	0,8847	1127,8	-1,71
4	240	0,8881	1242,9	1,00
5	240	0,8913	1304,7	2,46
6	240	0,9082	1654,0	10,69
7	240	0,9050	1592,2	9,23
8	240	0,8599	537,5	-15,62
9	240	0,9003	1462,1	6,16
10	240	0,8855	1145,6	-1,29
Overall	2400		1200,5	
H = 692,73 DF = 9 P = 0,000				
H = 692,73 DF = 9 P = 0,000 (adjusted for ties)				

Figura D.1 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema bidimensional guilhotinado

Kruskal-Wallis Test: Atendimento de datas versus CodHeur				
Kruskal-Wallis Test on Atendimento de datas				
CodHeur	N	Median	Ave Rank	Z
1	240	3,501	978,8	-5,23
2	240	3,645	1231,5	0,73
3	240	3,622	1192,3	-0,19
4	240	3,671	1313,2	2,66
5	240	3,650	1275,2	1,76
6	240	3,680	1296,2	2,26
7	240	3,686	1319,2	2,80
8	240	3,411	978,8	-5,22
9	240	3,660	1233,4	0,77
10	240	3,629	1186,4	-0,33
Overall	2400		1200,5	
H = 71,05 DF = 9 P = 0,000				
H = 71,05 DF = 9 P = 0,000 (adjusted for ties)				

Figura D.2 – Resultado do teste de Kruskal-Wallis para o critério de atendimento de datas do problema bidimensional guilhotinado

Kruskal-Wallis Test: Ponderado versus CodHeur				
Kruskal-Wallis Test on Ponderado				
CodHeur	N	Median	Ave Rank	Z
1	240	0,1808	385,9	-19,20
2	240	0,7372	1373,5	4,08
3	240	0,6191	1102,0	-2,32
4	240	0,7513	1415,8	5,07
5	240	0,7363	1385,6	4,36
6	240	0,8504	1731,7	12,52
7	240	0,8607	1743,1	12,79
8	240	0,1695	383,1	-19,26
9	240	0,7407	1396,5	4,62
10	240	0,6265	1087,7	-2,66
Overall	2400		1200,5	
H = 1039,33 DF = 9 P = 0,000				
H = 1039,35 DF = 9 P = 0,000 (adjusted for ties)				

Figura D.3 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema bidimensional guilhotinado

Kruskal-Wallis Test: Ocupação versus CodHeur				
Kruskal-Wallis Test on Ocupação				
CodHeur	N	Median	Ave Rank	Z
1	240	0,8663	554,2	-15,23
2	240	0,9123	1380,7	4,25
3	240	0,8952	1086,6	-2,68
4	240	0,9077	1307,7	2,53
5	240	0,9103	1374,4	4,10
6	240	0,9264	1676,5	11,22
7	240	0,9226	1619,6	9,88
8	240	0,8664	566,7	-14,94
9	240	0,9098	1337,2	3,22
10	240	0,8957	1101,3	-2,34
Overall	2400		1200,5	
H = 668,39 DF = 9 P = 0,000				
H = 668,39 DF = 9 P = 0,000 (adjusted for ties)				

Figura D.4 – Resultado do teste de Kruskal-Wallis para o critério de ocupação do problema bidimensional não-guilhotinado

Kruskal-Wallis Test: Atendimento de datas versus CodHeur				
Kruskal-Wallis Test on Atendimento de datas				
CodHeur	N	Median	Ave Rank	Z
1	240	3,457	988,7	-4,99
2	240	3,649	1230,0	0,70
3	240	3,640	1187,6	-0,30
4	240	3,685	1312,7	2,64
5	240	3,677	1285,0	1,99
6	240	3,674	1294,3	2,21
7	240	3,688	1319,4	2,80
8	240	3,437	959,2	-5,69
9	240	3,649	1237,2	0,86
10	240	3,623	1190,9	-0,23
Overall	2400		1200,5	
H = 74,07 DF = 9 P = 0,000				
H = 74,07 DF = 9 P = 0,000 (adjusted for ties)				

Figura D.5 – Resultado do teste de Kruskal-Wallis para o critério de atendimento de datas do problema bidimensional não-guilhotinado

Kruskal-Wallis Test: Ponderado versus CodHeur				
Kruskal-Wallis Test on Ponderado				
CodHeur	N	Median	Ave Rank	Z
1	240	0,1673	394,1	-19,00
2	240	0,7352	1352,2	3,58
3	240	0,6260	1047,3	-3,61
4	240	0,7617	1460,4	6,12
5	240	0,7680	1422,8	5,24
6	240	0,8757	1762,0	13,23
7	240	0,8801	1817,7	14,54
8	240	0,1597	355,1	-19,92
9	240	0,7321	1328,6	3,02
10	240	0,6131	1064,8	-3,20
Overall	2400		1200,5	
H = 1129,27 DF = 9 P = 0,000				
H = 1129,29 DF = 9 P = 0,000 (adjusted for ties)				

Figura D.6 – Resultado do teste de Kruskal-Wallis para o critério ponderado do problema bidimensional não-guilhotinado

APÊNDICE E – GRÁFICOS DE DESEMPENHO DAS HEURÍSTICAS DOS PROBLEMAS BIDIMENSIONAIS GUILHOTINADO E NÃO-GUILHOTINADO

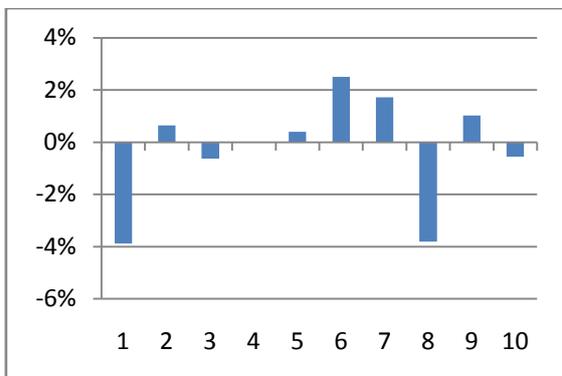


Gráfico 5.1 – Variação percentual média da ocupação em relação à heurística 4 (2DG)

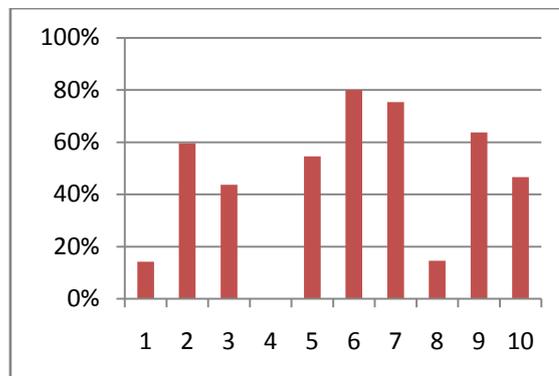


Gráfico 5.2 – Porcentagem de sucesso em ocupação em relação à heurística 4 (2DG)

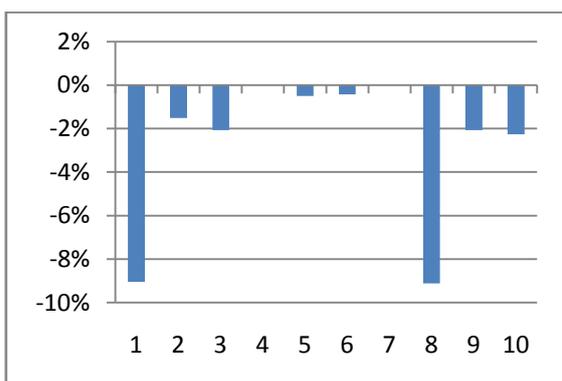


Gráfico 5.3 – Variação percentual média de atendimento de datas em relação à heurística 4 (2DG)

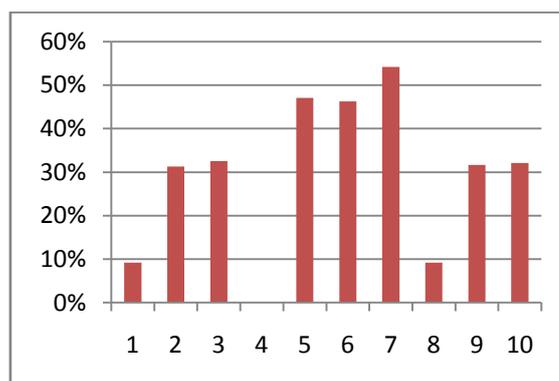


Gráfico 5.4 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 (2DG)

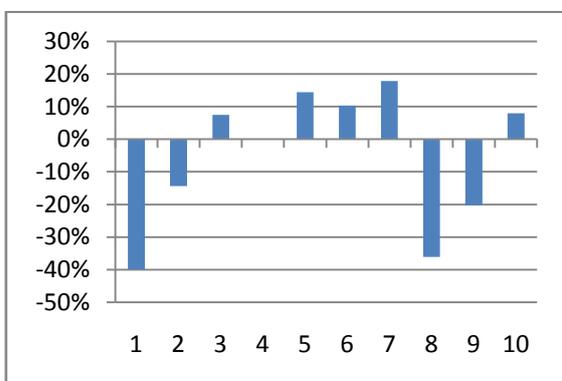


Gráfico 5.5 – Variação percentual média do critério ponderado em relação à heurística 4 (2DG)

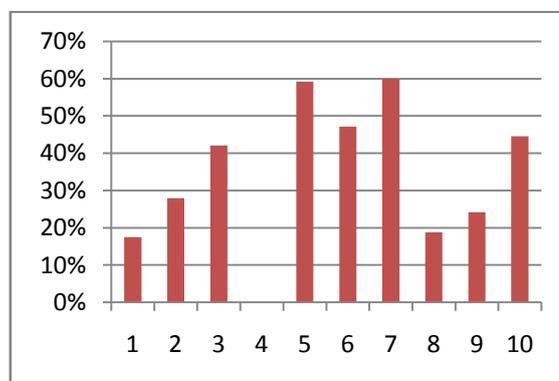


Gráfico 5.6 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 (2DG)

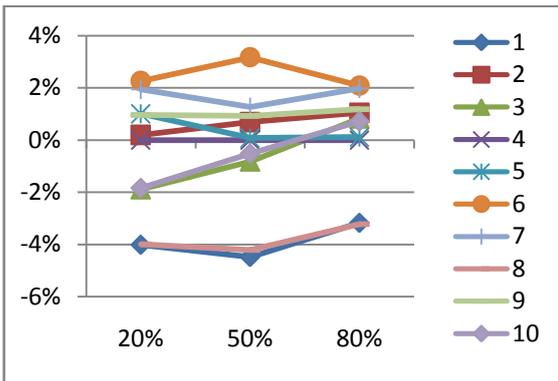


Gráfico 5.7 – Variação percentual média da ocupação em relação à heurística 4 por peso de ocupação (2DG)

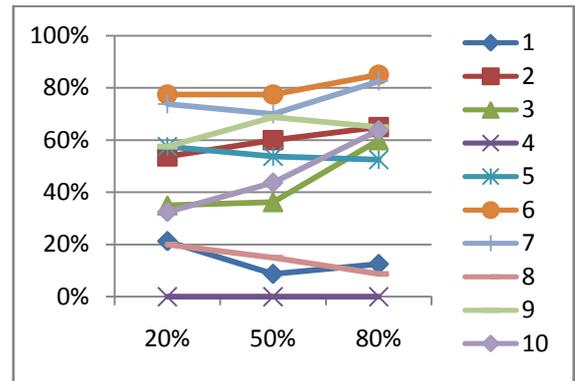


Gráfico 5.8 – Porcentagem de sucesso em ocupação em relação à heurística 4 por peso de ocupação (2DG)

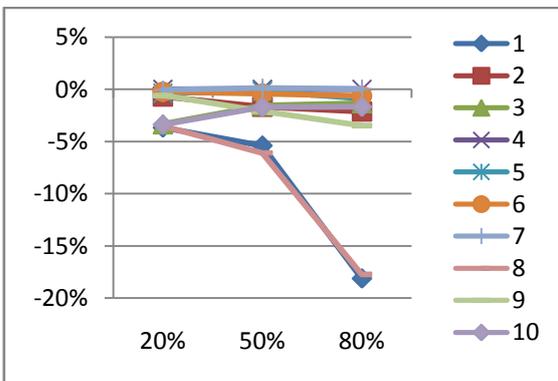


Gráfico 5.9 – Variação percentual média de atendimento de datas em relação à heurística 4 por peso de ocupação (2DG)

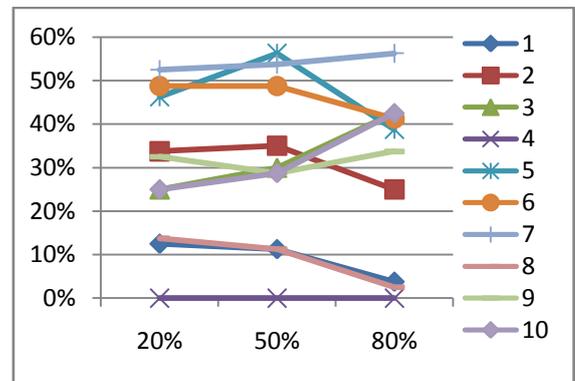


Gráfico 5.10 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 por peso de ocupação (2DG)

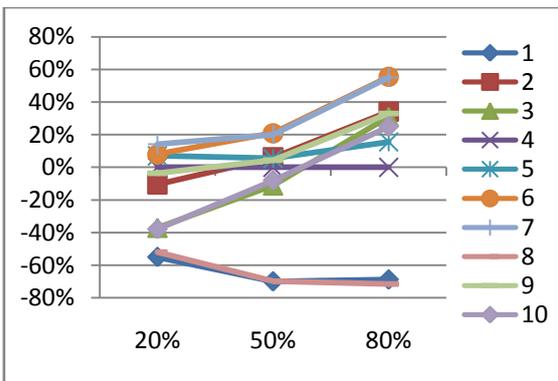


Gráfico 5.11 – Variação percentual média do critério ponderado em relação à heurística 4 por peso de ocupação (2DG)

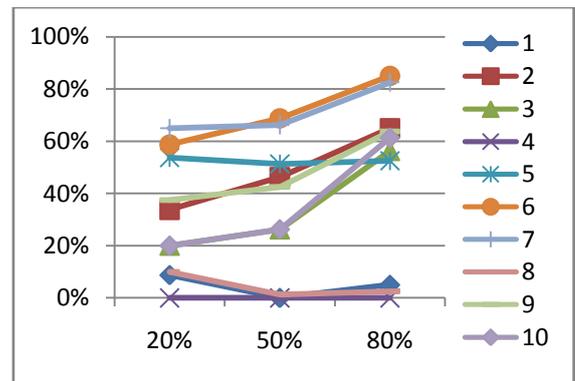


Gráfico 5.12 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 por peso de ocupação (2DG)

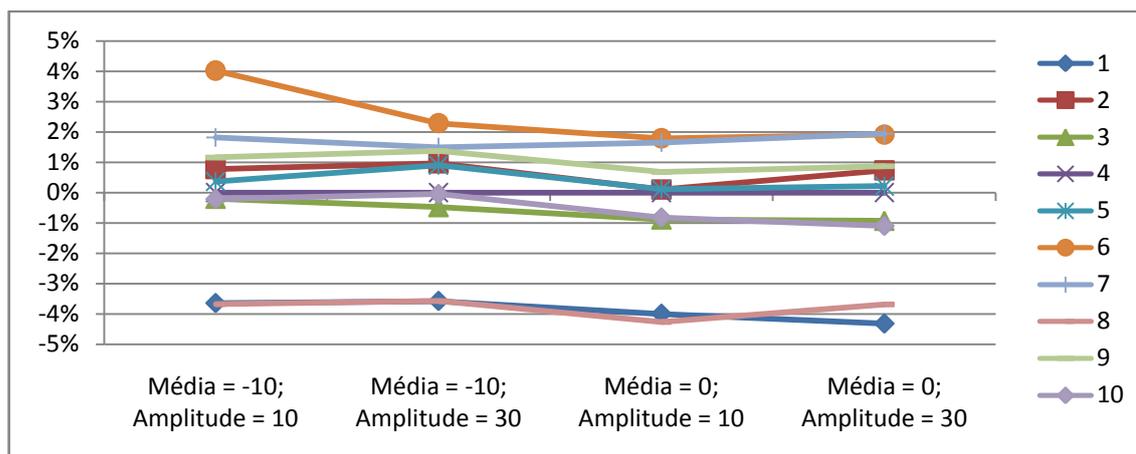


Gráfico 5.13 – Variação percentual média da ocupação em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)

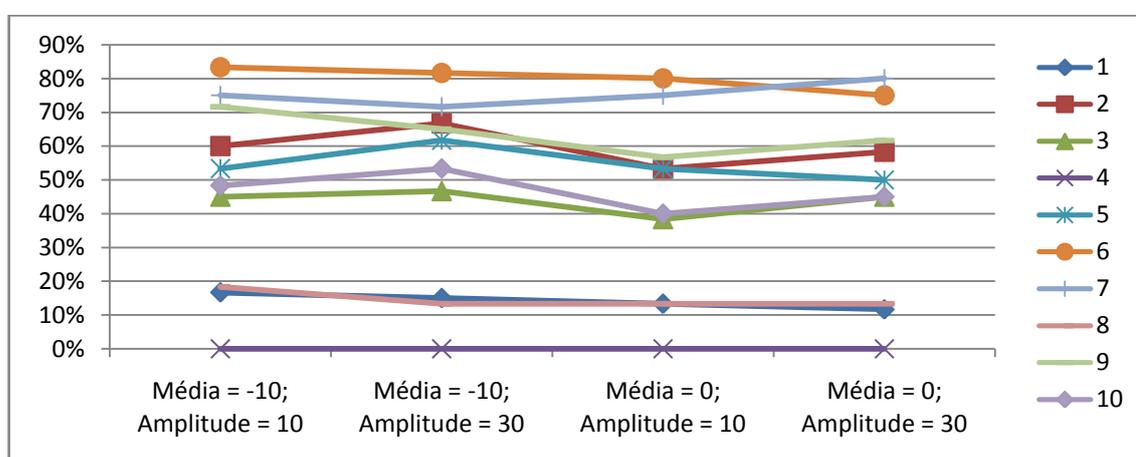


Gráfico 5.14 – Porcentagem de sucesso em ocupação em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)

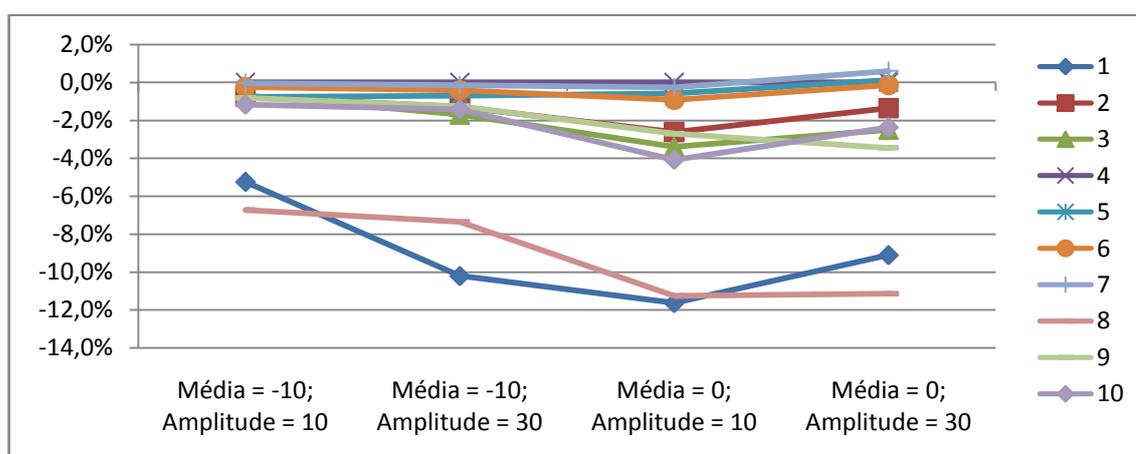


Gráfico 5.15 – Variação percentual média de atendimento de datas em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)

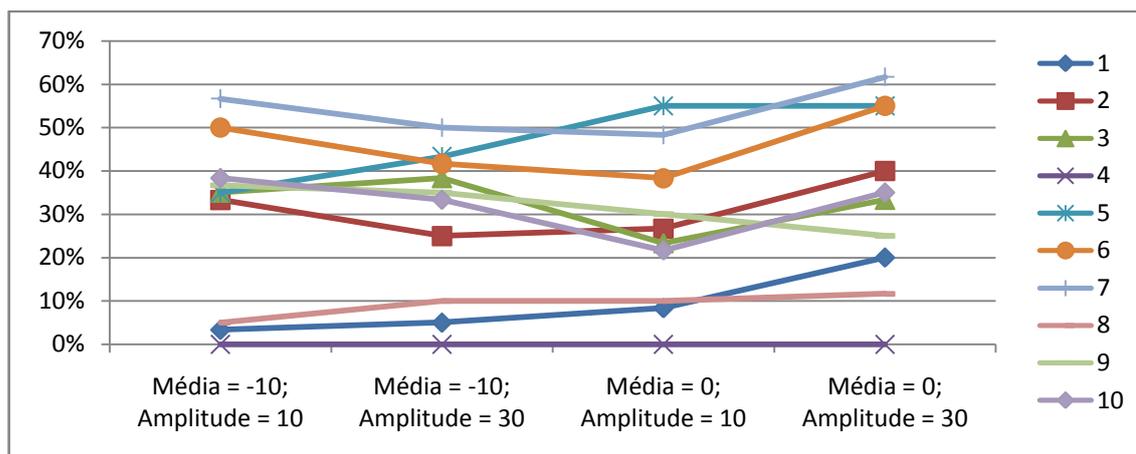


Gráfico 5.16 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)

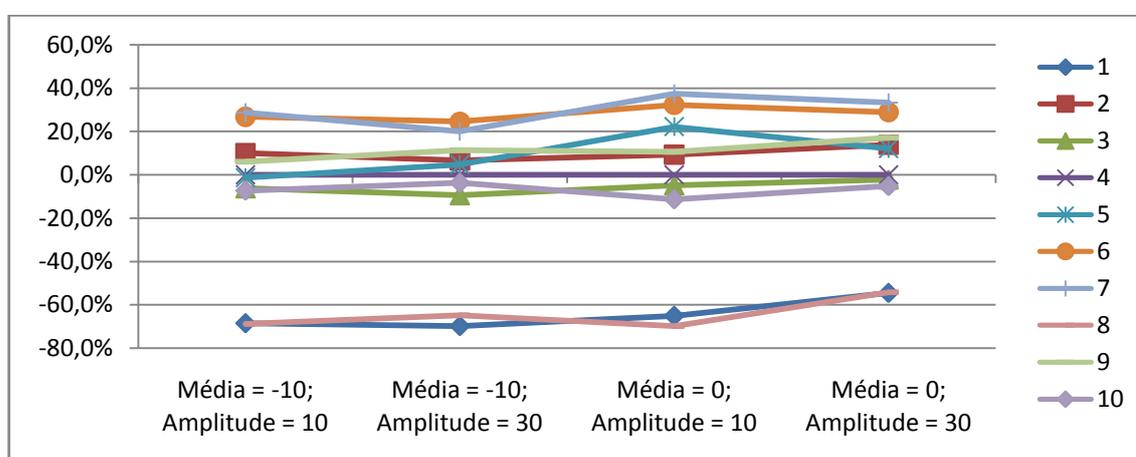


Gráfico 5.17 – Variação percentual média do critério ponderado em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)

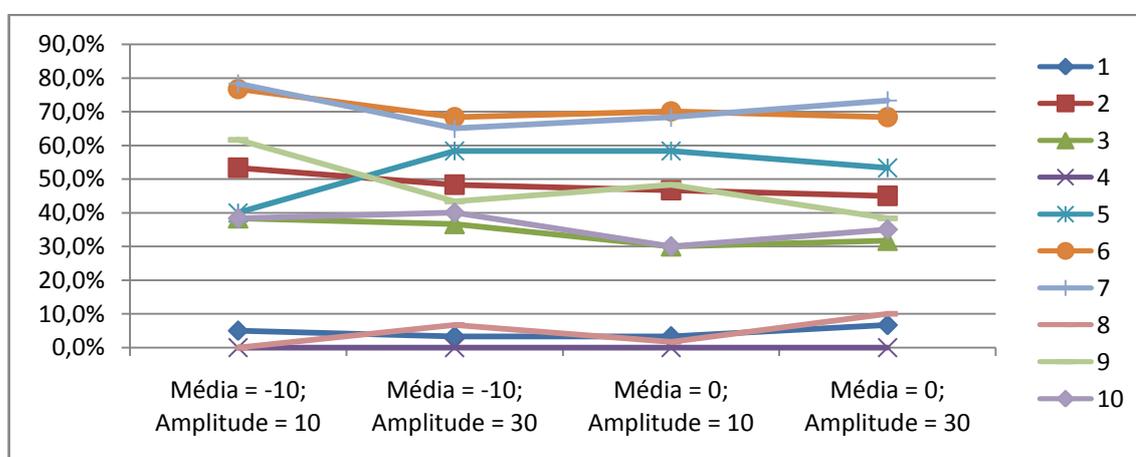


Gráfico 5.18 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DG)

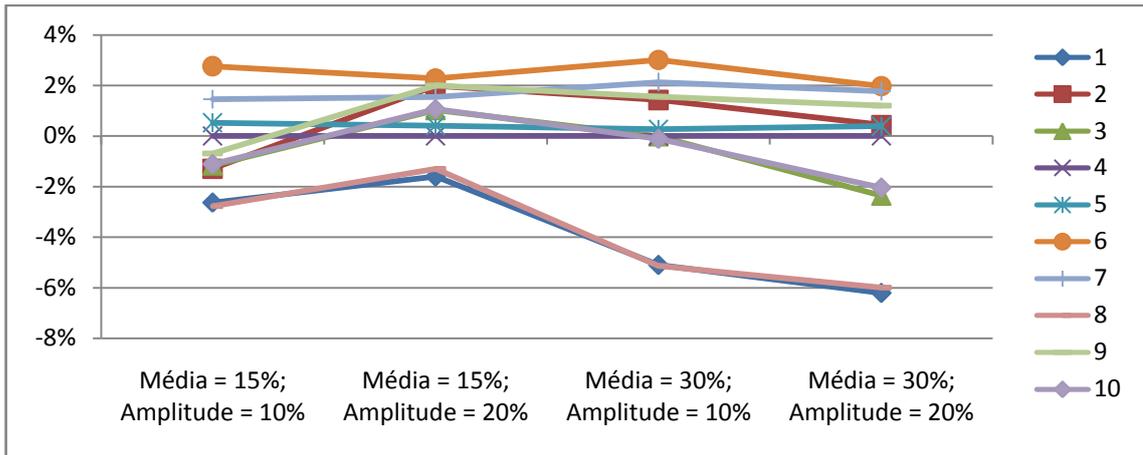


Gráfico 5.19 – Variação percentual média da ocupação em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)

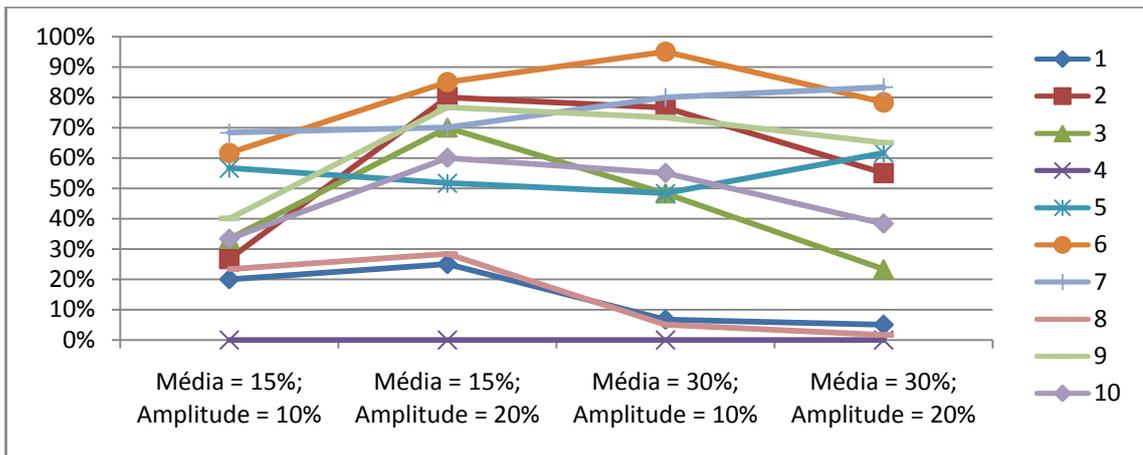


Gráfico 5.20 – Porcentagem de sucesso em ocupação em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)

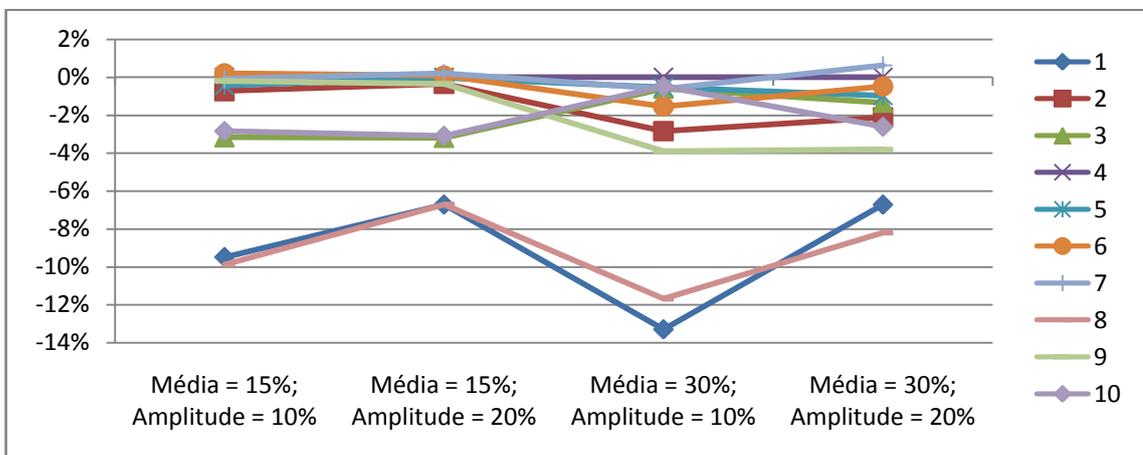


Gráfico 5.21 – Variação percentual média de atendimento de datas em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)

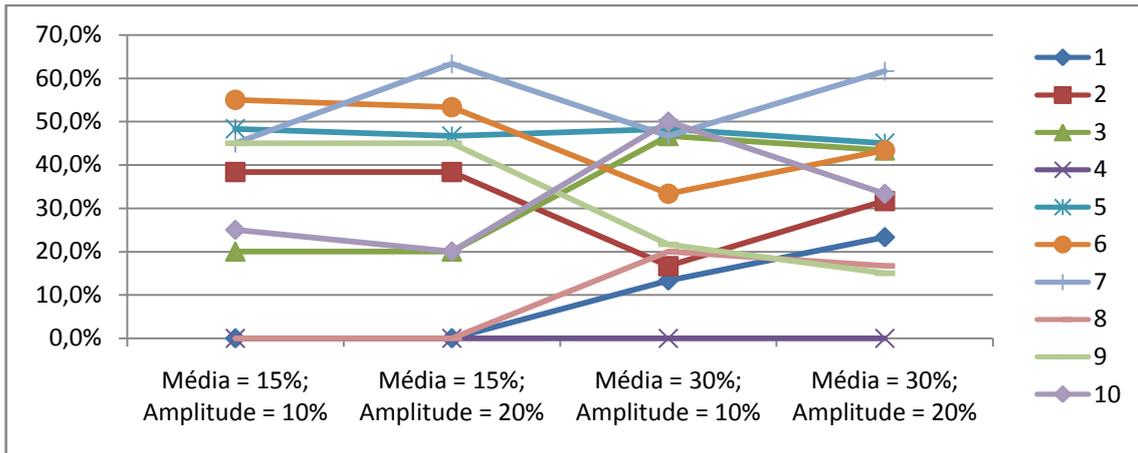


Gráfico 5.22 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)

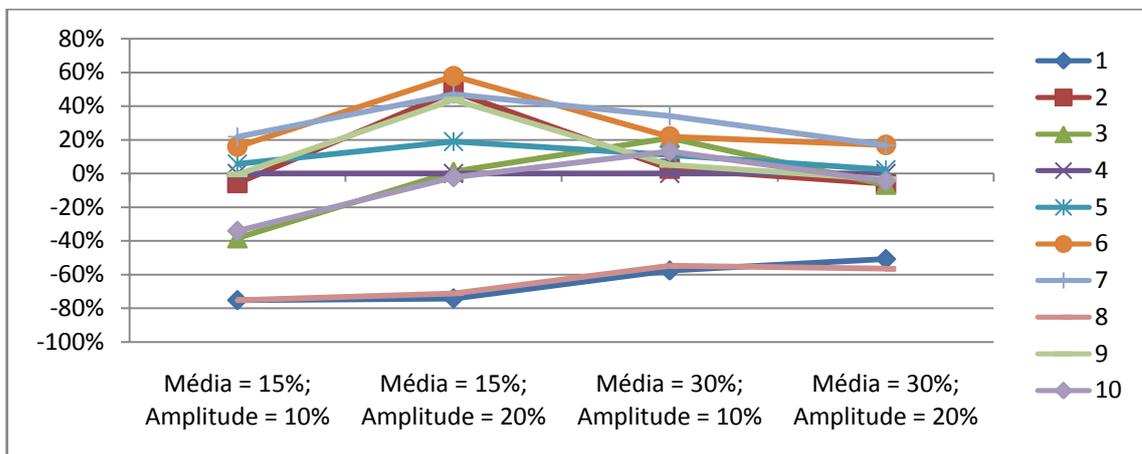


Gráfico 5.23 – Variação percentual média em critério ponderado em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)

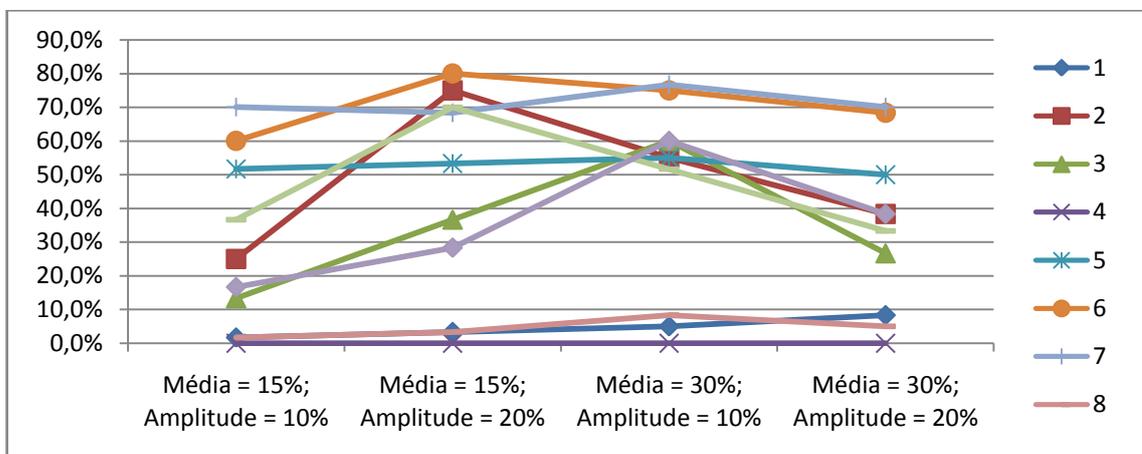


Gráfico 5.24 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DG)

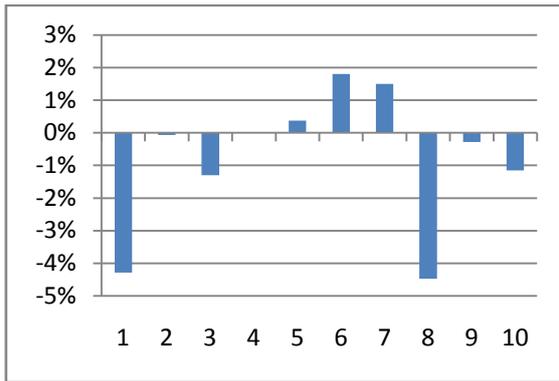


Gráfico 5.25 – Variação percentual média da ocupação em relação à heurística 4 (2DNG)

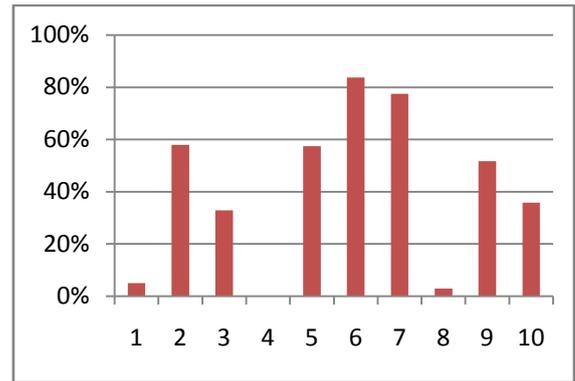


Gráfico 5.26 – Porcentagem de sucesso em ocupação em relação à heurística 4 (2DNG)

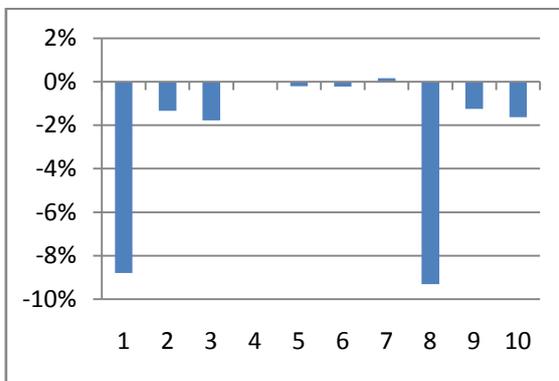


Gráfico 5.27 – Variação percentual média de atendimento de datas em relação à heurística 4 (2DNG)

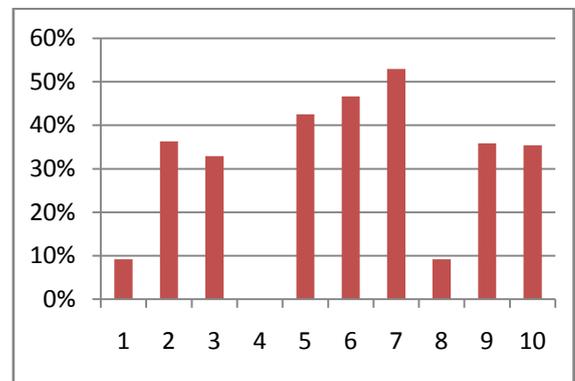


Gráfico 5.28 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 (2DNG)

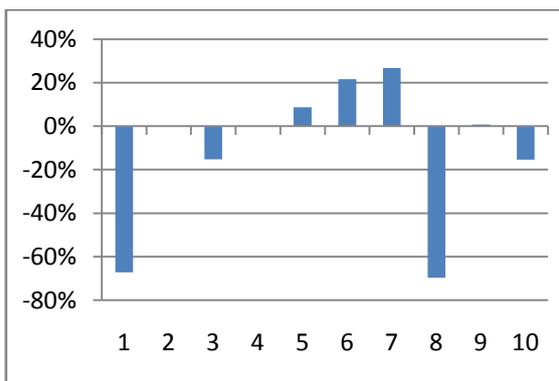


Gráfico 5.29 – Variação percentual média do critério ponderado em relação à heurística 4 (2DNG)

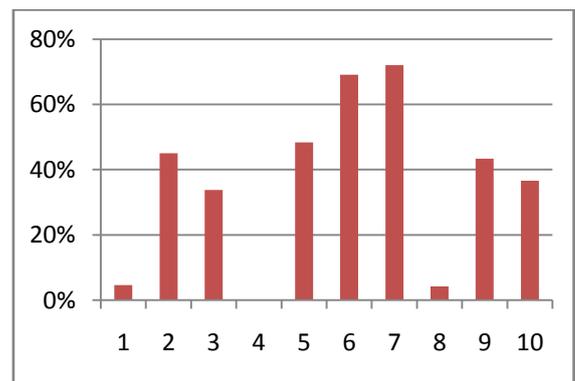


Gráfico 5.30 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 (2DNG)

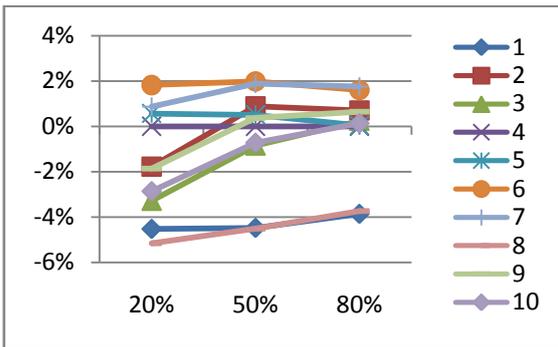


Gráfico 5.31 – Variação percentual média da ocupação em relação à heurística 4 por peso de ocupação (2DNG)

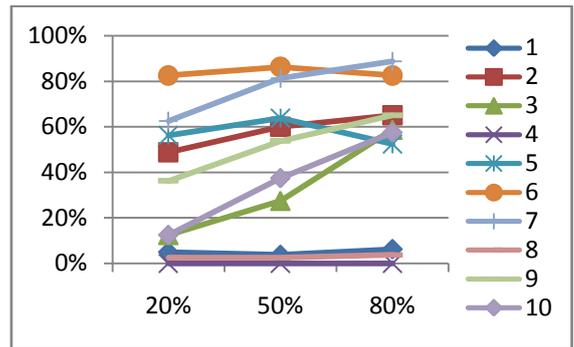


Gráfico 5.32 – Porcentagem de sucesso em ocupação em relação à heurística 4 por peso de ocupação (2DNG)

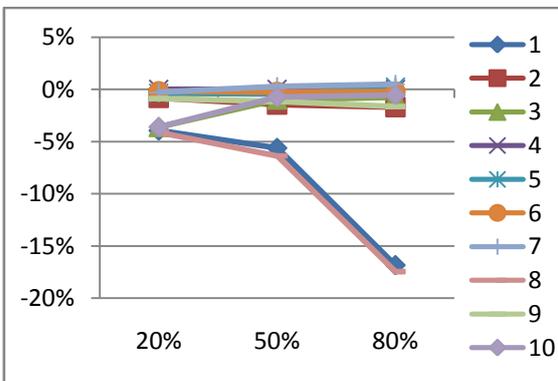


Gráfico 5.33 – Variação percentual média de atendimento de datas em relação à heurística 4 por peso de ocupação (2DNG)

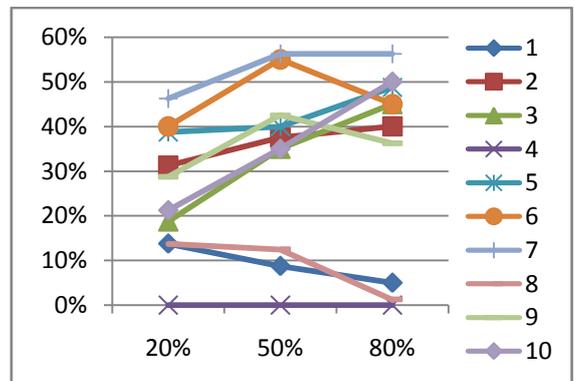


Gráfico 5.34 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 por peso de ocupação (2DNG)

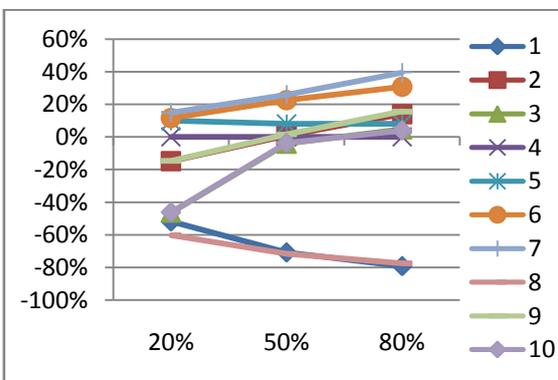


Gráfico 5.35 – Variação percentual média do critério ponderado em relação à heurística 4 por peso de ocupação (2DNG)

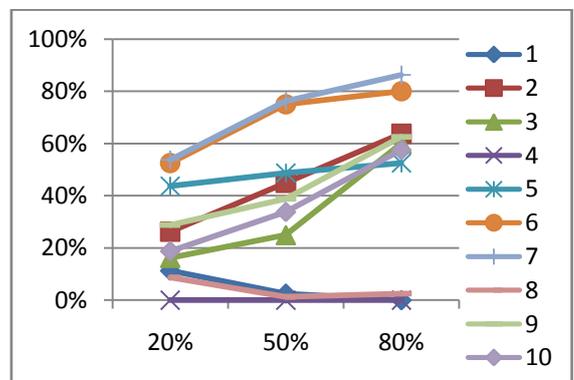


Gráfico 5.36 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 por peso de ocupação (2DNG)

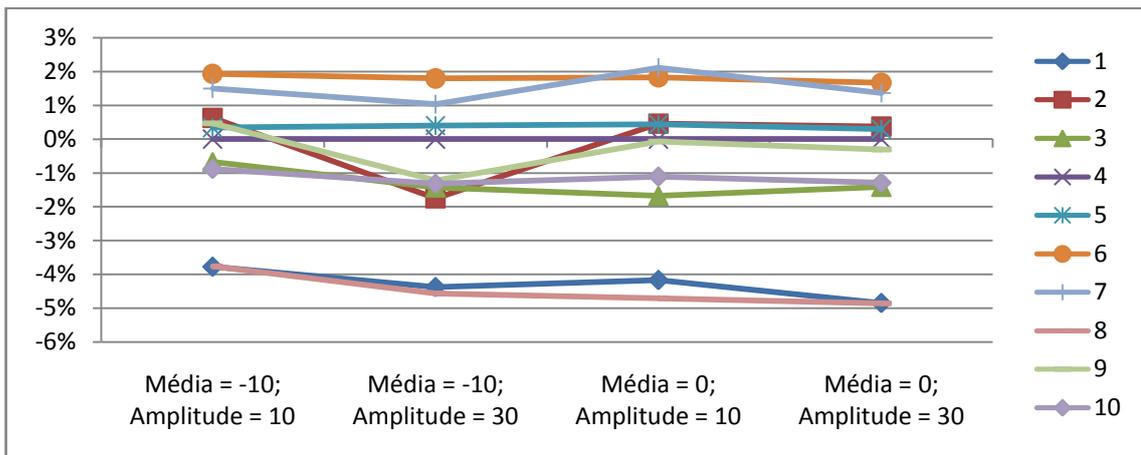


Gráfico 5.37 – Variação percentual média da ocupação em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)

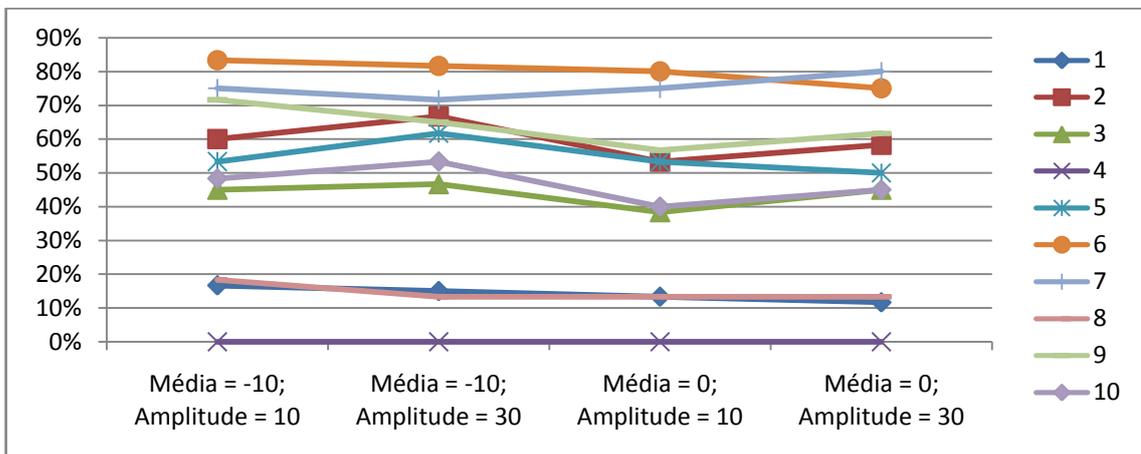


Gráfico 5.38 – Porcentagem de sucesso em ocupação em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)

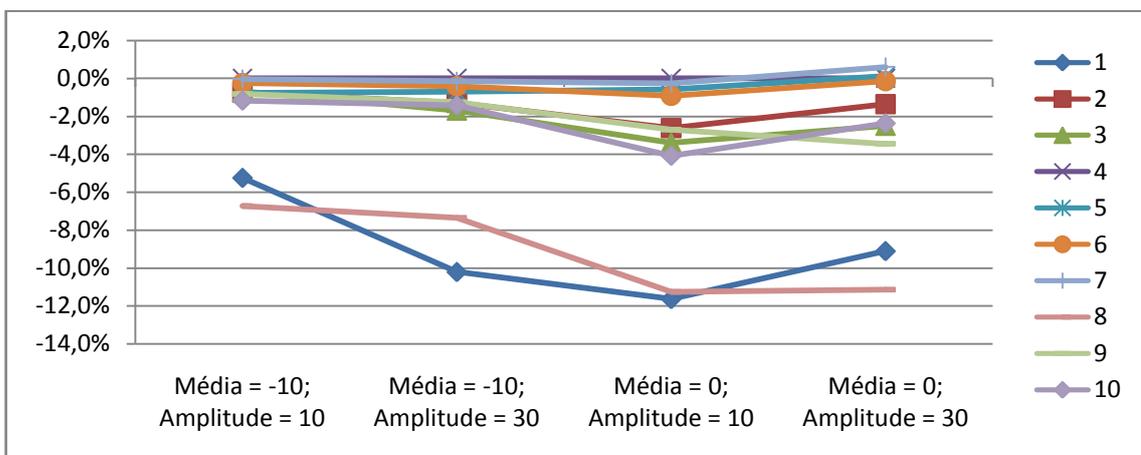


Gráfico 5.39 – Variação percentual média de atendimento de datas em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)

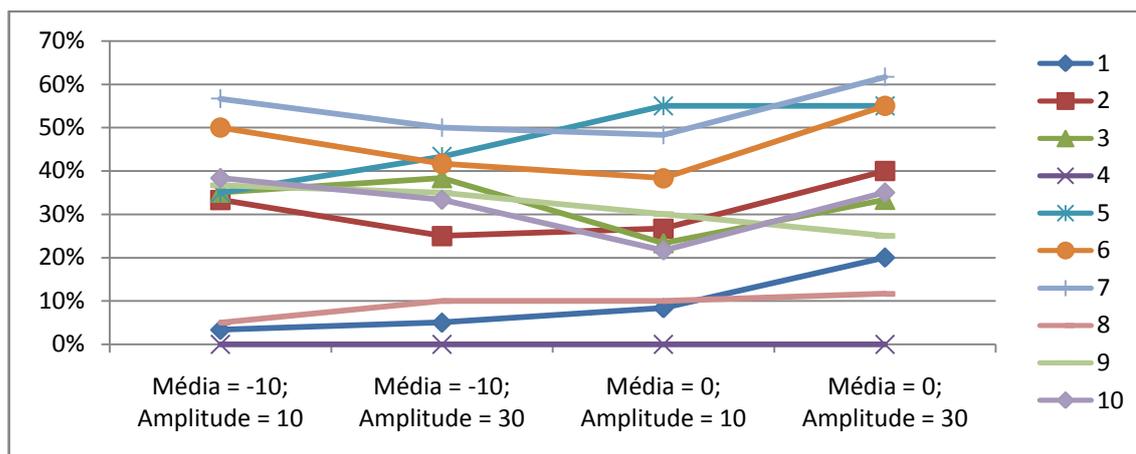


Gráfico 5.40 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)

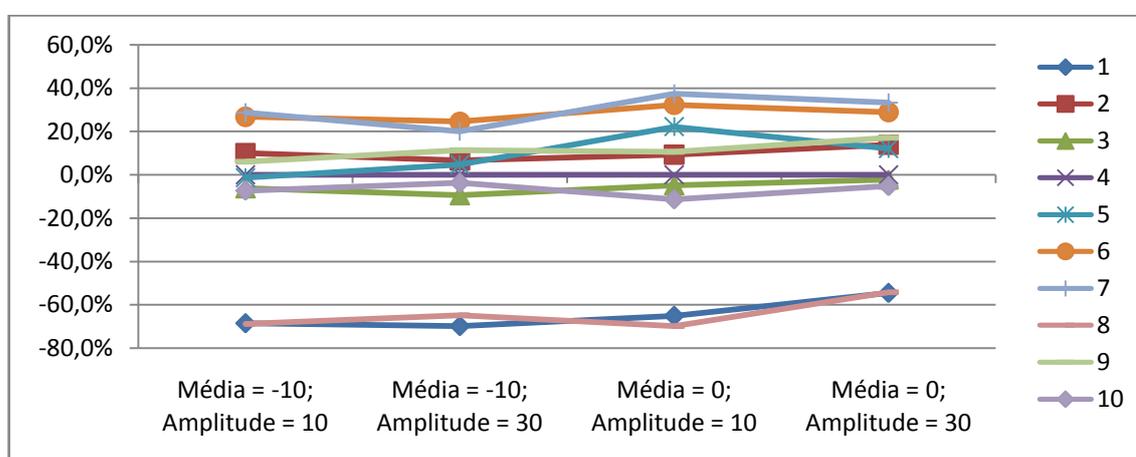


Gráfico 5.41 – Variação percentual média do critério ponderado em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)

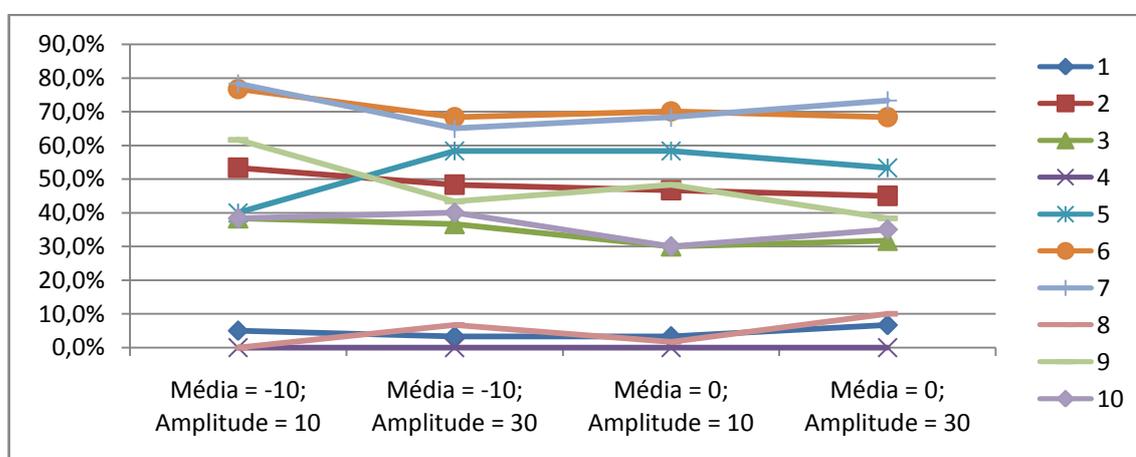


Gráfico 5.42 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 de acordo com a distribuição das folgas dinâmicas (2DNG)

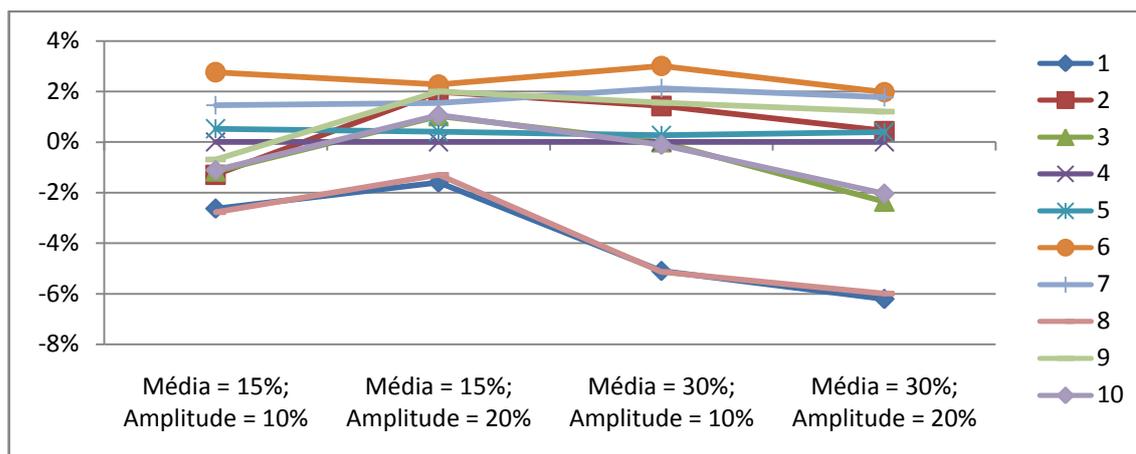


Gráfico 5.43 – Variação percentual média da ocupação em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)

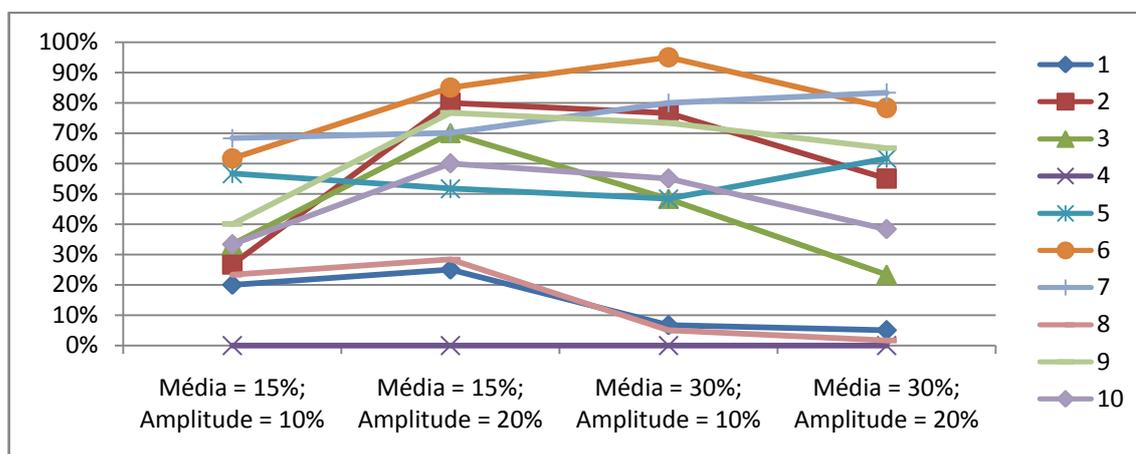


Gráfico 5.44 – Porcentagem de sucesso em ocupação em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)

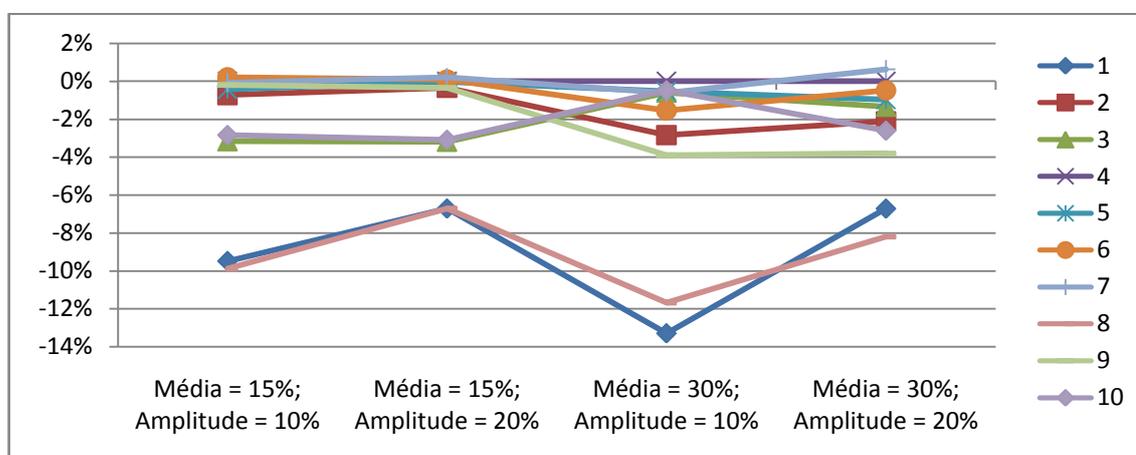


Gráfico 5.45 – Variação percentual média de atendimento de datas em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)

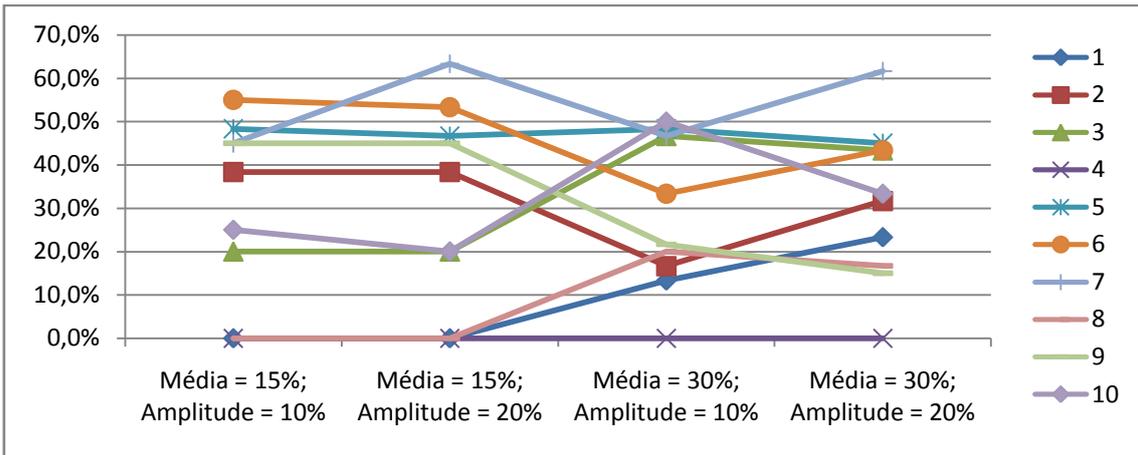


Gráfico 5.46 – Porcentagem de sucesso em atendimento de datas em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)

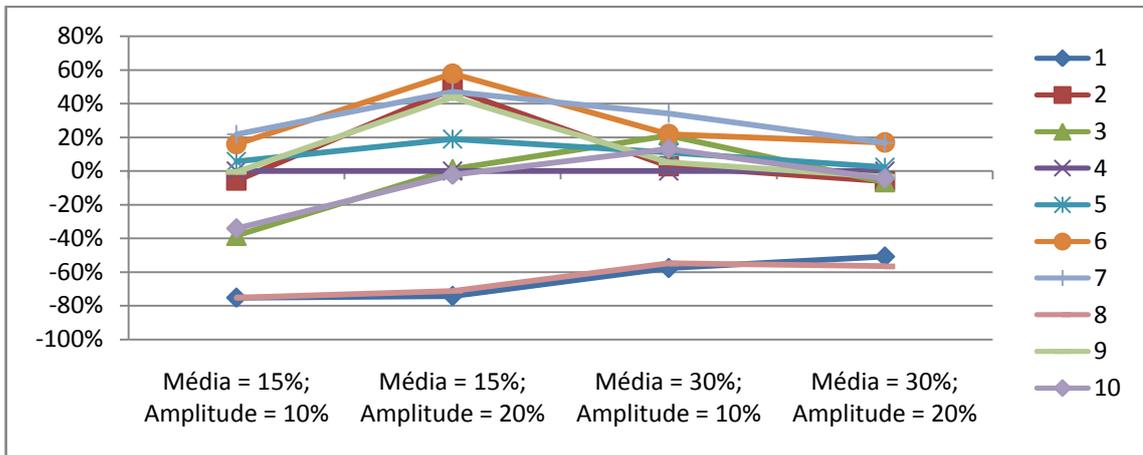


Gráfico 5.47 – Variação percentual média em critério ponderado em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)

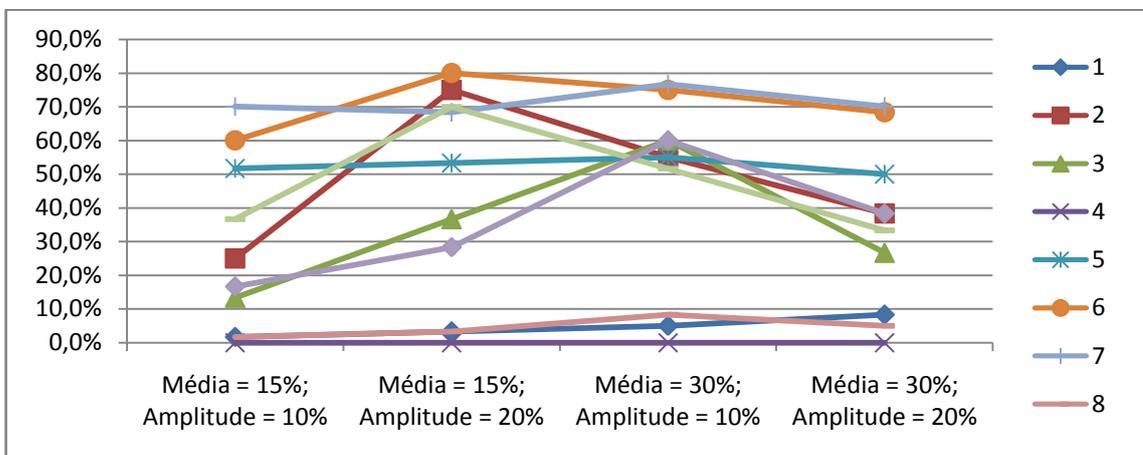


Gráfico 5.48 – Porcentagem de sucesso em critério ponderado em relação à heurística 4 de acordo com a distribuição das dimensões dos itens (2DNG)

APÊNDICE F – INTERFACES DO SISTEMA DESENVOLVIDO

Figura F.1 – Interface de parametrização

	A	B	C	D	E	F	G	H	I
1	SEQUEDad								
2	OcupPeso	DataPeso	CritDataMaxFila	NumIterMax	TpoCpuMax	NumIterSemGanhoMax	SeparaOPSN	TipoProb	CodHeur
3	50%	50%	150	500	60	100	VERDADEIRO	1D	3
4									
5									
6	HEURDad								
7	CodHeur	DescrHeur	PriorHeur	OrdIniHeur	ViesHeur	ProcGeral			
8	1	Determinística ordenada por fc CritData			Determinística	Determinística			
9	2	Determinística ordenada por ta CritOcup			Determinística	Determinística			
10	3	Determinística ordenada por bi CritPonderado			Determinística	Determinística			
11	4	Probabilística sem vies ordenac CritData			Sem	Probabilística			
12	5	Probabilística com vies linear or CritData			Linear	Probabilística			
13	6	Probabilística com vies linear or CritOcup			Linear	Probabilística			
14	7	Probabilística com vies linear or CritPonderado			Linear	Probabilística			
15	8	Probabilística com vies desprop: CritData			Desproporcional	Probabilística			
16	9	Probabilística com vies desprop: CritOcup			Desproporcional	Probabilística			
17	10	Probabilística com vies desprop: CritPonderado			Desproporcional	Probabilística			
18									
19									
20									
21									
22									
23									
24									
25									
26									
27									

Rodar

Figura F.2 – Interface de *input* de dados dos objetos

	K	L	M	N	P	Q	R	S
	OBJETODad							
	CodObj	CritAgrup	Car1	Car2	ObjQtde	PadraoSN	PriorObjAgr	AtivoSN
	1	A	100,0	100,0	1	VERDADEIRO	1	VERDADEIRO
	2	B	100,0	100,0	1	VERDADEIRO	1	VERDADEIRO
	3	A	50,0	70,0	1	VERDADEIRO	1	VERDADEIRO
	4	B	50,0	70,0	1	VERDADEIRO	1	VERDADEIRO
	5	A	30,0	30,0	1	VERDADEIRO	1	VERDADEIRO
	6	B	30,0	30,0	1	VERDADEIRO	1	VERDADEIRO

Figura F.3 – Interface de *input* de dados das ordens

ORDEMDad	CodOrdem	CodEmbr	CritAgrup	Car1	Car2	QtyMinTec	QtyMultTec	OrdQtde	FDin	OrientSN
1	100	A	22	17	0	1	1	14	VERDADEIRO	
2	200	A	20	17	0	1	1	0	FALSO	
3	300	A	20	22	0	1	1	-2	VERDADEIRO	
4	400	A	19	20	0	1	1	11	FALSO	
5	500	A	20	24	0	1	1	7	VERDADEIRO	
6	600	A	23	19	0	1	1	-11	FALSO	
7	700	A	21	16	0	1	1	-11	VERDADEIRO	
8	800	A	24	20	0	1	1	-11	FALSO	
9	900	A	22	18	0	1	1	-11	VERDADEIRO	
10	1000	A	19	15	0	1	1	-3	FALSO	
11	1100	A	25	15	0	1	1	-8	VERDADEIRO	
12	1200	A	23	23	0	1	1	0	FALSO	
13	1300	A	25	16	0	1	1	8	VERDADEIRO	
14	1400	A	25	21	0	1	1	0	FALSO	
15	1500	A	19	22	0	1	1	8	VERDADEIRO	
16	1600	A	18	25	0	1	1	-14	FALSO	
17	1700	A	19	24	0	1	1	-6	VERDADEIRO	
18	1800	A	19	20	0	1	1	-8	FALSO	
19	1900	A	17	24	0	1	1	13	VERDADEIRO	
20	2000	A	19	19	0	1	1	14	FALSO	
21	2100	A	18	23	0	1	1	5	VERDADEIRO	
22	2200	A	22	17	0	1	1	-7	FALSO	
23	2300	A	15	16	0	1	1	-6	VERDADEIRO	
24	2400	A	18	17	0	1	1	-15	FALSO	
25	2500	A	15	25	0	1	1	10	VERDADEIRO	
26	2600	A	18	25	0	1	1	-4	FALSO	

Figura F.4 – Interface de *output* de dados dos padrões de corte

PadCorte	CodObj	CodItem	CritAgrup	PadCortePrior	CritAgrup	CodObj	CritOcup	CritData	CritOcupNorm	CritDataNorm	CritPond	NumItens
6	1	169	A	6	A	1	99%	3,68	92,86	99,68	96,27	5
6	1	41	A	7	A	1	100%	3,39	100,00	91,70	95,85	5
6	1	172	A	20	A	1	100%	2,68	100,00	72,41	86,20	5
6	1	111	A	5	A	1	96%	3,69	71,43	100,00	85,71	5
6	1	157	A	15	A	1	99%	2,63	92,86	71,10	81,98	5
7	1	195	A	19	A	1	96%	2,91	71,43	78,74	75,08	5
7	1	184	A	1	A	1	93%	3,67	50,00	99,38	74,69	4
7	1	161	A	14	A	1	94%	3,21	57,14	86,86	72,00	5
7	1	22	A	22	A	1	96%	2,65	71,43	71,54	71,49	5
7	1	145	A	31	A	1	99%	1,74	92,86	46,72	69,79	6
20	1	190	A	26	A	1	100%	1,41	100,00	37,91	68,96	5
20	1	17	A	3	A	1	94%	2,92	57,14	79,06	68,10	4
20	1	125	A	2	A	1	93%	3,19	50,00	86,15	68,08	4
20	1	126	A	27	A	1	99%	1,60	92,86	43,14	68,00	5
20	1	75	A	24	A	1	100%	0,99	100,00	26,54	63,27	5
5	1	24	A	37	A	1	100%	0,65	100,00	17,16	58,58	6
5	1	16	A	12	A	1	90%	3,00	28,57	81,22	54,90	4
5	1	43	A	8	A	1	93%	2,14	50,00	57,70	53,85	4
5	1	31	A	21	A	1	93%	2,14	50,00	57,69	53,84	5
5	1	7	A	29	A	1	97%	1,01	78,57	27,05	52,81	5
15	1	105	A	38	A	1	100%	0,21	100,00	5,28	52,64	6
15	1	50	A	9	A	1	95%	1,48	64,29	39,89	52,09	4
15	1	18	A	10	A	1	100%	0,04	100,00	0,64	50,32	4
15	1	46	A	39	A	1	100%	0,03	100,00	0,32	50,16	6
15	1	164	A	11	A	1	100%	0,02	100,00	0,14	50,07	4
19	1	117	A	40	A	1	100%	0,02	100,00	0,07	50,04	6

APÊNDICE G – CÓDIGO-FONTE DO SISTEMA DESENVOLVIDO EM LINGUAGEM VBA

Option Explicit

'Matriz do sequenciamento

Type rSEQUEDad

OcupPeso As Double

DataPeso As Double

CritDataMaxFila As Long

NumIterMax As Long

TpoCpuMax As Long

NumIterSemGanhoMax As Long

SeparaOPSN As Boolean

TipoProb As String

CodHeur As Long

End Type

Public mSEQUEDad() As rSEQUEDad

'Matriz das heurísticas

Type rHEURDad

CodHeur As Long

DescrHeur As String

ProcGer As String

SeqHeur As String

ViesHeur As String

PriorHeur As Long

End Type

Public mHEURDad() As rHEURDad

'Matriz dos dados dos itens

Type rITEMDad

CodItem As Long

CritAgrup As String

Car1 As Double

Car2 As Double

CodOP As Long

FDin As Double

CritOcup As Double

CritData As Double

CritPond As Double

OrientSN As Boolean

AtivoSN As Boolean

KeyStr As String

End Type

Public mITEMDad() As rITEMDad

Public mITEMLocal() As rITEMDad

'Matriz dos objetos

```

Type rOBJETODad
  CodObj As Long
  CritAgrup As String
  Car1 As Double
  Car2 As Double
  CTAlts As Long
  ObjQtde As Long
  PadraoSN As Boolean
  PriorObjAgr As String
  AtivoSN As Boolean
  KeyStr As String
End Type
Public mOBJETODad() As rOBJETODad

```

'Matriz dos padroes de corte

```

Type rPadCorteDesemp
  CodPad As Long
  CritAgrup As String
  CodObj As Long
  ContObj As Long
  mITEMPadCorte() As rITEMDad
  CritOcup As Double
  CritData As Double
  CritOcupNorm As Double
  CritDataNorm As Double
  CritPond As Double
  MelhorObjSN As Boolean
End Type
Public mPadCorteDesemp() As rPadCorteDesemp
Public mPadCorteDesempTemp() As rPadCorteDesemp

```

'Matriz das ordens

```

Type rORDEMDad
  CodOrdem As Long
  CodEmbr As Long
  CritAgrup As String
  Car1 As Double
  Car2 As Double
  QtdMinTec As Long
  QtdMultTec As Long
  OrdQtde As Long
  FDin As Double
  OrientSN As Boolean
  PriorExtSN As Boolean
End Type
Public mORDEMDad() As rORDEMDad

```

```

Public glQtdHeur As Long, glQtdItem As Long, glQtdObjeto As Long, glQtdItemLocal As Long

```

```
Public glQtdOrdem As Long, glQtdPadCorteDesemp As Long, glQtdPadCorteDesempTemp  
As Long
```

```
Public mITEMPadCorte() As rITEMDad
```

```
Public glQtdITEMPadCorte As Long
```

```
Type rSubObjeto
```

```
Car1 As Double
```

```
Car2 As Double
```

```
AtivoSN As Boolean
```

```
End Type
```

```
Public mSubObjeto() As rSubObjeto
```

```
Public glQtdSubObjeto As Long
```

```
Type rCantos
```

```
X As Double
```

```
Y As Double
```

```
tipo As String
```

```
Ordem As Long
```

```
arestaX As Double
```

```
arestaY As Double
```

```
KeyXY As String
```

```
End Type
```

```
Public mCantos() As rCantos
```

```
Public glQtdCantos As Long
```

```
Type rCOPMs
```

```
CodItem As Long
```

```
xp As Double
```

```
yp As Double
```

```
xc As Double
```

```
yc As Double
```

```
FFV As Double
```

```
tipo As String
```

```
canto As Long
```

```
ig As String
```

```
End Type
```

```
Public mCOPMs() As rCOPMs
```

```
Public glQtdCOPMs As Long
```

```
Public maiorCOPM As Long
```

```
Type rItensPos
```

```
CodItem As Long
```

```
X As Double
```

```
Y As Double
```

```
xi As Double
```

```
yi As Double
```

```
CodOP As Long
```

```
FDin As Long
```

```
End Type
```

```
Public mItensPos() As rItensPos
```

```
Public glQtdItensPos As Long
```

```
Sub AASeque()
```

```
'Rotina principal
```

```
    AuxLimpaSaidas
```

```
    CarregaDados
```

```
    SequenciaPadCorte
```

```
    GravamITEMDad
```

```
End Sub
```

```
Sub AuxLimpaSaidas()
```

```
'limpa as saídas
```

```
Dim rg As Range
```

```
Set rg = Worksheets("SequeSai").Range("PadCorte")
```

```
If rg.Cells(3, 1) <> "" Then
```

```
    With rg
```

```
        Range(.Offset(2, 0), .Offset(1, 0).End(xlDown).End(xlToRight)).ClearContents
```

```
    End With
```

```
End If
```

```
Set rg = Worksheets("SequeSai").Range("PadCortePrior")
```

```
If rg.Cells(3, 1) <> "" Then
```

```
    With rg
```

```
        Range(.Offset(2, 0), .Offset(1, 0).End(xlDown).End(xlToRight)).ClearContents
```

```
    End With
```

```
End If
```

```
End Sub
```

```
Sub CarregaDados()
```

```
'carrega dados
```

```
Dim i As Long, j As Long, k As Long
```

```
Dim maxOcup As Double, maxData As Double, auxCar As Double
```

```
Dim minOcup As Double, minData As Double
```

```
Dim rg As Range
```

```
'Carrega parâmetros gerais do programa
```

```
ReDim mSEQUEDad(1)
```

```
i = 1
```

```
Set rg = Worksheets("SequeDad").Range("SEQUEDad")
```

```
    mSEQUEDad(i).OcupPeso = rg.Cells(3, 1)
```

```
    mSEQUEDad(i).DataPeso = rg.Cells(3, 2)
```

```
    mSEQUEDad(i).CritDataMaxFila = rg.Cells(3, 3)
```

```
    mSEQUEDad(i).NumIterMax = rg.Cells(3, 4)
```

```
    mSEQUEDad(i).TpoCpuMax = rg.Cells(3, 5)
```

```

mSEQUEDad(i).NumIterSemGanhoMax = rg.Cells(3, 6)
mSEQUEDad(i).SeparaOPSN = rg.Cells(3, 7)
mSEQUEDad(i).TipoProb = rg.Cells(3, 8)
mSEQUEDad(i).CodHeur = rg.Cells(3, 9)

```

```

'Carrega matriz de heurísticas
Set rg = Worksheets("SequeDad").Range("HEURDad")
i = 0
Do While rg.Cells(i + 3, 1) <> ""
    i = i + 1
Loop
glQtdHeur = i
ReDim mHEURDad(glQtdHeur)
For i = 1 To glQtdHeur
    mHEURDad(i).CodHeur = rg.Cells(i + 2, 1)
    mHEURDad(i).DescrHeur = rg.Cells(i + 2, 2)
    mHEURDad(i).PriorHeur = rg.Cells(i + 2, 3)
    mHEURDad(i).SeqHeur = rg.Cells(i + 2, 4)
    mHEURDad(i).ViesHeur = rg.Cells(i + 2, 5)
    mHEURDad(i).ProcGer = rg.Cells(i + 2, 6)
Next

```

```

'Carrega matriz mORDEMDad
Set rg = Worksheets("SequeDad").Range("ORDEMDad")
i = 0
Do While rg.Cells(i + 3, 1) <> ""
    i = i + 1
Loop
glQtdOrdem = i
ReDim mORDEMDad(glQtdOrdem)
For i = 1 To glQtdOrdem
    mORDEMDad(i).CodOrdem = rg.Cells(i + 2, 1)
    mORDEMDad(i).CodEmbr = rg.Cells(i + 2, 2)
    mORDEMDad(i).CritAgrup = rg.Cells(i + 2, 3)
    mORDEMDad(i).Car1 = rg.Cells(i + 2, 4)
    mORDEMDad(i).Car2 = rg.Cells(i + 2, 5)
    mORDEMDad(i).QtdMinTec = rg.Cells(i + 2, 6)
    mORDEMDad(i).QtdMultTec = rg.Cells(i + 2, 7)
    mORDEMDad(i).OrdQtde = rg.Cells(i + 2, 8)
    mORDEMDad(i).FDin = rg.Cells(i + 2, 9)
    mORDEMDad(i).OrientSN = rg.Cells(i + 2, 10)
    mORDEMDad(i).PriorExtSN = rg.Cells(i + 2, 11)
Next
QSmORDEMDadCodOrdem 1, glQtdOrdem

```

```

'Carrega matriz mITEMDad
glQtdItem = 0
For i = 1 To glQtdOrdem
    glQtdItem = glQtdItem + mORDEMDad(i).OrdQtde / mORDEMDad(i).QtdMultTec
Next

```

```

ReDim mITEMDad(glQtdItem)
k = 0: maxOcup = 0: maxData = 0
For i = 1 To glQtdOrdem
  For j = 1 To mORDEMDad(i).OrdQtde / mORDEMDad(i).QtdMultTec
    mITEMDad(i).CodItem = k + j
    mITEMDad(i).CritAgrup = mORDEMDad(i).CritAgrup
    mITEMDad(i).Car1 = mORDEMDad(i).Car1
    mITEMDad(i).Car2 = mORDEMDad(i).Car2
    mITEMDad(i).CodOP = mORDEMDad(i).CodOrdem
    mITEMDad(i).FDin = mORDEMDad(i).FDin
    mITEMDad(i).OrientSN = mORDEMDad(i).OrientSN
    'Inativa se ultrapassa a FDin máxima
    If mITEMDad(i).FDin > mSEQUEDad(1).CritDataMaxFila Then
      mITEMDad(i).AtivoSN = False
    Else
      mITEMDad(i).AtivoSN = True
    End If
    'Grava em CritOcup a dimensão que ditará a fila futuramente
    If (mITEMDad(i).OrientSN = False And mITEMDad(i).Car2 < mITEMDad(i).Car1) Or _
      mSEQUEDad(1).TipoProb = "1D" Then
      mITEMDad(i).CritOcup = mITEMDad(i).Car1
    Else
      mITEMDad(i).CritOcup = mITEMDad(i).Car2
    End If
    If mITEMDad(i).CritOcup > maxOcup Then
      maxOcup = mITEMDad(i).CritOcup
    End If
    If i = 1 Then
      minOcup = mITEMDad(i).CritOcup
    ElseIf mITEMDad(i).CritOcup < minOcup Then
      minOcup = mITEMDad(i).CritOcup
    End If
    'Grava em CritData a "folga dinâmica corrigida"
    If mITEMDad(i).FDin < 0 Then
      mITEMDad(i).CritData = -mITEMDad(i).FDin
    Else
      mITEMDad(i).CritData = 1 / (mITEMDad(i).FDin + 2)
    End If
    If mITEMDad(i).CritData > maxData Then
      maxData = mITEMDad(i).CritData
    End If
    If i = 1 Then
      minData = mITEMDad(i).CritData
    ElseIf mITEMDad(i).CritData < minData Then
      minData = mITEMDad(i).CritData
    End If
  Next
  k = k + mORDEMDad(i).OrdQtde
Next

```

```

For i = 1 To glQtdItem
  If minOcup <> maxOcup Then
    If minData <> maxData Then
      mITEMDad(i).CritPond = mSEQUEDad(1).OcupPeso * ((mITEMDad(i).CritOcup -
minOcup) / (maxOcup - minOcup)) + _
      mSEQUEDad(1).DataPeso * ((mITEMDad(i).CritData - minData) /
(maxData - minData))
    Else
      mITEMDad(i).CritPond = mITEMDad(i).CritOcup
    End If
  Else
    If minData <> maxData Then
      mITEMDad(i).CritPond = mITEMDad(i).CritData
    Else
      mITEMDad(i).CritPond = 1
    End If
  End If
  mITEMDad(i).KeyStr = ObtChCompITEM(i, mSEQUEDad(1).CodHeur)
Next

'Carrega matriz mOBJETODad
Set rg = Worksheets("SequeDad").Range("OBJETODad")
i = 0
Do While rg.Cells(i + 3, 1) <> ""
  i = i + 1
Loop
glQtdObjeto = i
ReDim mOBJETODad(glQtdObjeto)
For i = 1 To glQtdObjeto
  mOBJETODad(i).CodObj = rg.Cells(i + 2, 1)
  mOBJETODad(i).CritAgrup = rg.Cells(i + 2, 2)
  mOBJETODad(i).Car1 = rg.Cells(i + 2, 3)
  mOBJETODad(i).Car2 = rg.Cells(i + 2, 4)
  mOBJETODad(i).CTAlts = rg.Cells(i + 2, 5)
  mOBJETODad(i).ObjQtde = rg.Cells(i + 2, 6)
  mOBJETODad(i).PadraoSN = rg.Cells(i + 2, 7)
  mOBJETODad(i).PriorObjAgr = rg.Cells(i + 2, 8)
  mOBJETODad(i).AtivoSN = rg.Cells(i + 2, 9)
Next

End Sub

Sub SequenciaPadCorte()
'Rotina que forma os padrões de corte

'Auxiliares para manipular faixas
Dim iFx1 As Long, fFx1 As Long, iFx2 As Long, fFx2 As Long
Dim iFx3 As Long, fFx3 As Long, iFx4 As Long, fFx4 As Long
Dim cont As Long, minimoOcup As Double, minimoData As Double

```

'Contadores

Dim i As Long, j As Long, cod As Long, ContObj As Long

'Auxiliares na normalização e avaliação

Dim MelhorOBJ As Double, melhorCritAgrup As Double

Dim codMelhorOBJ As Long, codmelhorCritAgrup As Long, codObjmelhorCritAgrup As Long

Dim maximoOcup As Double, maximoData As Double

ReDim mPadCorteDesemp(UBound(mITEMDad))

cont = 1: glQtdPadCorteDesemp = 0

Do

'Sort inicial por CritAgrup, AtivoSN, CritOrd (segundo heurística) e OP

QSmITEMDadKeyStr 1, UBound(mITEMDad)

'Encontro faixa de mITEMDad para o critério de agrupamento sendo analisado

ObtFaixaCritAgrupmITEMDad mITEMDad(cont).CritAgrup, iFx1, fFx1

Do While mITEMDad(iFx1).AtivoSN = True

For i = 1 To glQtdObjeto

mOBJETODad(i).KeyStr = ObtChCompOBJETO(i)

Next

'Acha a faixa de objetos do CritAgrup, Ativos, de maior PriorObjAgr

QSmOBJETODadKeyStr 1, glQtdObjeto

cod = BSCritAgrupmOBJETODad(mITEMDad(iFx1).CritAgrup)

j = cod

If j > 1 Then

Do Until (mOBJETODad(j).AtivoSN = True) Or j = 1

j = j - 1

Loop

If j > 1 Then

Do While mOBJETODad(j).AtivoSN = True And mOBJETODad(j).CritAgrup = mOBJETODad(j - 1).CritAgrup

j = j - 1

Loop

End If

Else

j = 1

End If

ObtFaixaKeyStrmOBJETODad mOBJETODad(j).KeyStr, iFx2, fFx2

'Obtém a faixa de itens ativos no critério de agrupamento analisado

iFx3 = iFx1

j = 1

Do While iFx1 + j <= fFx1

If mITEMDad(iFx1 + j).AtivoSN = True Then

```

    j = j + 1
Else
    Exit Do
End If
Loop
fFx3 = iFx3 + j - 1

'Acha melhor padrão de corte para o agrupamento
CriaPadCorte iFx2, fFx2, iFx3, fFx3, mSEQUEDad(1).CodHeur

QSmITEMDadCodItem 1, UBound(mITEMDad)
For i = 1 To UBound(mPadCorteDesemp(glQtdPadCorteDesemp).mITEMPadCorte)
    cod =
BSmITEMDadCodItem(mPadCorteDesemp(glQtdPadCorteDesemp).mITEMPadCorte(i).Cod
Item)
    mITEMDad(cod).AtivoSN = False
    mITEMDad(cod).KeyStr = ObtChCompITEM(cod, mSEQUEDad(1).CodHeur)
Next

'Reordeno mITEMDad como estava antes da avaliação
QSmITEMDadKeyStr 1, UBound(mITEMDad)
glQtdItem = glQtdItem -
UBound(mPadCorteDesemp(glQtdPadCorteDesemp).mITEMPadCorte)

'Atualiza cadastro de objetos
If mOBJETODad(mPadCorteDesemp(glQtdPadCorteDesemp).ContObj).PadraoSN =
False Then
    mOBJETODad(mPadCorteDesemp(glQtdPadCorteDesemp).ContObj).ObjQtde = _
    mOBJETODad(mPadCorteDesemp(glQtdPadCorteDesemp).ContObj).ObjQtde - 1
    If mOBJETODad(mPadCorteDesemp(glQtdPadCorteDesemp).ContObj).ObjQtde = 0
Then
        mOBJETODad(mPadCorteDesemp(glQtdPadCorteDesemp).ContObj).AtivoSN =
False
    End If
End If

Loop

'Atualizo o contador para pular para o início da próxima faixa de critério de agrupamento
cont = cont + fFx1 - iFx1 + 1

Loop Until cont > UBound(mITEMDad)

'Sequenciamento de mPadCorteDesemp()
ReDim Preserve mPadCorteDesemp(glQtdPadCorteDesemp)
maximoOcup = 0: maximoData = 0: minimoOcup = 1

'Cálculo valores normalizados
For i = 1 To glQtdPadCorteDesemp
    If mPadCorteDesemp(i).CritOcup > maximoOcup Then

```

```

        maximoOcup = mPadCorteDesemp(i).CritOcup
    End If
    If mPadCorteDesemp(i).CritData > maximoData Then
        maximoData = mPadCorteDesemp(i).CritData
    End If
    If mPadCorteDesemp(i).CritOcup < minimoOcup And mPadCorteDesemp(i).CritOcup >
0.5 Then
        minimoOcup = mPadCorteDesemp(i).CritOcup
    End If
    If i = 1 Then
        minimoData = mPadCorteDesemp(i).CritData
    ElseIf mPadCorteDesemp(i).CritData < minimoData Then
        minimoData = mPadCorteDesemp(i).CritData
    End If
Next
For i = 1 To glQtdPadCorteDesemp
    If maximoData <> minimoData Then
        mPadCorteDesemp(i).CritDataNorm = 100 * (mPadCorteDesemp(i).CritData -
minimoData) / (maximoData - minimoData)
    Else
        mPadCorteDesemp(i).CritDataNorm = 100
    End If
    If maximoOcup <> minimoOcup Then
        mPadCorteDesemp(i).CritOcupNorm = 100 * (mPadCorteDesemp(i).CritOcup -
minimoOcup) / (maximoOcup - minimoOcup)
    Else
        mPadCorteDesemp(i).CritOcupNorm = 100
    End If
    mPadCorteDesemp(i).CritPond = mSEQUEDad(1).DataPeso *
mPadCorteDesemp(i).CritDataNorm _
+ mSEQUEDad(1).OcupPeso *
mPadCorteDesemp(i).CritOcupNorm
Next
    QSmPadCorteDesempCritPond 1, glQtdPadCorteDesemp

```

End Sub

```

Sub CriaPadCorte(ByVal iFxOBJ As Long, ByVal fFxOBJ As Long, _
    ByVal iFxITEM As Long, ByVal fFxITEM As Long, ByVal CodHeur As Long)
'Procedimento que recebe a faixa de Objetos candidatos e a faixa de itens possíveis,
'preenchendo as matrizes mPadCorteDetTemp e mPadCorteDesempTemp
'Entradas: - Faixa de objetos ativos e prioritários do Critério de Agrupamento
'          - Faixa de itens ativos do critério de agrupamento
'          - Heurística
'Saída: insere o melhor padrão de corte do critério de agrupamento na mPadCorteDesemp

```

Dim i As Long

Dim iFx As Long, fFx As Long

'Auxiliares na normalização e avaliação

Dim MelhorOBJ As Double, melhorCritAgrup As Double, melhorDATA As Double, melhorOCUP As Double

Dim codMelhorOBJ As Long, codmelhorCritAgrup As Long, codObjmelhorCritAgrup As Long

Dim maximoOcup As Double, maximoData As Double, minimoOcup As Double, minimoData As Double

'Auxiliares para determinar parada das probabilísticas

Dim iter As Long, iterSemGanho As Long, tempoMax As Variant, ParadaSN As Boolean

ReDim mPadCorteDesempTemp(fFxOBJ - iFxOBJ + 1)

glQtdPadCorteDesempTemp = 0: codMelhorOBJ = 0: MelhorOBJ = 0

'Faz o melhor lote para cada um dos objetos candidatos

For ContObj = iFxOBJ To fFxOBJ

'Seleciona o tipo de formação de padrão que será usado

Select Case mHEURDad(CodHeur).ProcGer

Case "Determinística"

Select Case mSEQUEDad(1).TipoProb

Case "1D"

Unidimensional mOBJETODad(ContObj).Car1, iFxITEM, fFxITEM, CodHeur

Case "2DG"

BidimensionalGuilhotinado mOBJETODad(ContObj).Car1,
mOBJETODad(ContObj).Car2, iFxITEM, fFxITEM, CodHeur

Case "2DNG"

BidimensionalNaoGuilhotinado mOBJETODad(ContObj).Car1,
mOBJETODad(ContObj).Car2, iFxITEM, fFxITEM, CodHeur

Case "3D"

Unidimensional mOBJETODad(ContObj).Car1, iFxITEM, fFxITEM, CodHeur

End Select

'Atualização de mPadCorteDesempTemp

If glQtdITEMPadCorte > 0 Then

glQtdPadCorteDesempTemp = glQtdPadCorteDesempTemp + 1

mPadCorteDesempTemp(glQtdPadCorteDesempTemp).CodObj =
mOBJETODad(ContObj).CodObj

mPadCorteDesempTemp(glQtdPadCorteDesempTemp).ContObj = ContObj

mPadCorteDesempTemp(glQtdPadCorteDesempTemp).CritAgrup =
mOBJETODad(ContObj).CritAgrup

ReDim

mPadCorteDesempTemp(glQtdPadCorteDesempTemp).mITEMPadCorte(glQtdITEMPadCorte)
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).mITEMPadCorte =

mITEMPadCorte

mPadCorteDesempTemp(glQtdPadCorteDesempTemp).MelhorObjSN = True

End If

'Caso Probabilístico
Case Else

```

glQtdObjetoCand = 0: maximoOcup = 0: maximoData = 0: MelhorOBJ = 0:
codMelhorOBJ = 0
iter = 0: iterSemGanho = 0: ParadaSN = False: melhorDATA = 0: melhorOCUP = 0
tempoMax = Now + TimeSerial(0, 0, mSEQUEDad(1).TpoCpuMax): minimoOcup =
1

```

Do

```

'Chama heurística de formação de lote
Select Case mSEQUEDad(1).TipoProb
Case "1D"
Unidimensional mOBJETODad(ContObj).Car1, iFxITEM, fFxITEM,
CodHeur
Case "2DG"
BidimensionalGuilhotinado mOBJETODad(ContObj).Car1,
mOBJETODad(ContObj).Car2, iFxITEM, fFxITEM, CodHeur
Case "2DNG"
BidimensionalNaoGuilhotinado mOBJETODad(ContObj).Car1,
mOBJETODad(ContObj).Car2, iFxITEM, fFxITEM, CodHeur
Case "3D"
Unidimensional mOBJETODad(ContObj).Car1, iFxITEM, fFxITEM,
CodHeur
End Select

```

```

'Atualização de mPadCorteDesempTemp
If glQtdITEMPadCorte > 0 Then
glQtdPadCorteDesempTemp = glQtdPadCorteDesempTemp + 1
If UBound(mPadCorteDesempTemp) < glQtdPadCorteDesempTemp Then
ReDim Preserve mPadCorteDesempTemp(glQtdPadCorteDesempTemp +
mSEQUEDad(1).NumIterSemGanhoMax)
End If
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).CodObj =
mOBJETODad(ContObj).CodObj
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).ContObj = ContObj
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).CritAgrup =
mOBJETODad(ContObj).CritAgrup
ReDim
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).mITEMPadCorte(glQtdITEMPadCorte)
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).mITEMPadCorte =
mITEMPadCorte
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).MelhorObjSN = False
End If

```

iter = iter + 1

'Só avalia os padrões quando há potencial de parada por falta de melhora

```

If iter Mod mSEQUEDad(1).NumIterSemGanhoMax = 0 Then

    If mPadCorteDesempTemp(glQtdPadCorteDesempTemp).ContObj = ContObj
Then
        maximoOcup = 0: maximoData = 0

        For j = glQtdPadCorteDesempTemp -
mSEQUEDad(1).NumIterSemGanhoMax + 1 To glQtdPadCorteDesempTemp
            'Calcula valor do candidato
            CalculaValor
mPadCorteDesempTemp(glQtdPadCorteDesempTemp).ContObj,
mPadCorteDesempTemp(j).CritData, mPadCorteDesempTemp(j).CritOcup, j
        Next

        'Reordeno mPadCorteDesempTemp para normalização
        QSmPadCorteDesempTempCodObj 1, glQtdPadCorteDesempTemp
        ObtFaixaCodObjmPadCorteDesempTemp mOBJETODad(ContObj).CodObj,
iFx, fFx

        'Máximos de Ocup e Data para normalização
        For i = iFx To fFx
            If mPadCorteDesempTemp(i).CritOcup > maximoOcup Then
                maximoOcup = mPadCorteDesempTemp(i).CritOcup
            End If
            If mPadCorteDesempTemp(i).CritData > maximoData Then
                maximoData = mPadCorteDesempTemp(i).CritData
            End If
            If mPadCorteDesempTemp(i).CritOcup < minimoOcup And
mPadCorteDesempTemp(i).CritOcup > 0.5 Then
                minimoOcup = mPadCorteDesempTemp(i).CritOcup
            End If
            If i = 1 Then
                minimoData = mPadCorteDesempTemp(i).CritData
            ElseIf mPadCorteDesempTemp(i).CritData < minimoData Then
                minimoData = mPadCorteDesempTemp(i).CritData
            End If
        Next
        'Normalização dos valores
        For i = iFx To fFx
            If maximoData <> minimoData Then
                mPadCorteDesempTemp(i).CritDataNorm = 100 *
(mPadCorteDesempTemp(i).CritData - minimoData) / (maximoData - minimoData)
            Else
                mPadCorteDesempTemp(i).CritDataNorm = 100
            End If
            If maximoOcup <> minimoOcup Then
                mPadCorteDesempTemp(i).CritOcupNorm = 100 *
(mPadCorteDesempTemp(i).CritOcup - minimoOcup) / (maximoOcup - minimoOcup)
            Else
                mPadCorteDesempTemp(i).CritOcupNorm = 100
        Next
    End If
End If

```

```

        End If
        mPadCorteDesempTemp(i).CritPond = mSEQUEDad(1).DataPeso *
mPadCorteDesempTemp(i).CritDataNorm _
        + mSEQUEDad(1).OcupPeso *
mPadCorteDesempTemp(i).CritOcupNorm
    Next

    'Verifico a incumbente
    For i = iFx To fFx
        If mPadCorteDesempTemp(i).CritPond > MelhorOBJ Then
            MelhorOBJ = mPadCorteDesempTemp(i).CritPond
            codMelhorOBJ = i
        End If
    Next
    If melhorDATA >= mPadCorteDesempTemp(codMelhorOBJ).CritData And _
    melhorOCUP >= mPadCorteDesempTemp(codMelhorOBJ).CritOcup Then
        iterSemGanho = iterSemGanho + mSEQUEDad(1).NumIterSemGanhoMax
    Else
        melhorDATA = mPadCorteDesempTemp(codMelhorOBJ).CritData
        melhorOCUP = mPadCorteDesempTemp(codMelhorOBJ).CritOcup
    End If
    Else
        iterSemGanho = iterSemGanho + mSEQUEDad(1).NumIterSemGanhoMax
    End If
End If

If iter >= mSEQUEDad(1).NumIterMax Or _
iterSemGanho >= mSEQUEDad(1).NumIterSemGanhoMax Or _
Now >= tempoMax Then
    mPadCorteDesempTemp(codMelhorOBJ).MelhorObjSN = True
    ParadaSN = True
End If

Loop Until ParadaSN = True

End Select

Next 'Próximo Objeto ativo do critério de agrupamento

'Avaliação dos melhores padrões gerados

'Obtém faixa de mPadCorteDesempTemp com MelhorOBJSN=True
QSmPadCorteDesempTempMelhorObjSN 1, glQtdPadCorteDesempTemp
ObtFaixaMelhorOBJSNmPadCorteDesempTemp iFx, fFx

For j = iFx To fFx
    'Calcula valor do candidato
    CalculaValor mPadCorteDesempTemp(j).ContObj, mPadCorteDesempTemp(j).CritData,
mPadCorteDesempTemp(j).CritOcup, j

```

```

Next

melhorCritAgrup = 0: codmelhorCritAgrup = 0: maximoOcup = 0: maximoData = 0:
minimoOcup = 1

'Cálculo valores normalizados
For i = iFx To fFx
  If mPadCorteDesempTemp(i).CritOcup > maximoOcup Then
    maximoOcup = mPadCorteDesempTemp(i).CritOcup
  End If
  If mPadCorteDesempTemp(i).CritData > maximoData Then
    maximoData = mPadCorteDesempTemp(i).CritData
  End If
  If mPadCorteDesempTemp(i).CritOcup < minimoOcup And
mPadCorteDesempTemp(i).CritOcup > 0.5 Then
    minimoOcup = mPadCorteDesempTemp(i).CritOcup
  End If
  If i = 1 Then
    minimoData = mPadCorteDesempTemp(i).CritData
  ElseIf mPadCorteDesempTemp(i).CritData < minimoData Then
    minimoData = mPadCorteDesempTemp(i).CritData
  End If
Next
For i = iFx To fFx
  If maximoData <> minimoData Then
    mPadCorteDesempTemp(i).CritDataNorm = 100 *
(mPadCorteDesempTemp(i).CritData - minimoData) / (maximoData - minimoData)
  Else
    mPadCorteDesempTemp(i).CritDataNorm = 100
  End If
  If maximoOcup <> minimoOcup Then
    mPadCorteDesempTemp(i).CritOcupNorm = 100 *
(mPadCorteDesempTemp(i).CritOcup - minimoOcup) / (maximoOcup - minimoOcup)
  Else
    mPadCorteDesempTemp(i).CritOcupNorm = 100
  End If
  mPadCorteDesempTemp(i).CritPond = mSEQUEDad(1).DataPeso *
mPadCorteDesempTemp(i).CritDataNorm +
mSEQUEDad(1).OcupPeso *
mPadCorteDesempTemp(i).CritOcupNorm
Next
'Escolhe melhor do critério de agrupamento
For i = iFx To fFx
  If mPadCorteDesempTemp(i).CritPond > melhorCritAgrup Then
    melhorCritAgrup = mPadCorteDesempTemp(i).CritPond
    codmelhorCritAgrup = i
  End If
Next

glQtdPadCorteDesemp = glQtdPadCorteDesemp + 1

```

```

If UBound(mPadCorteDesemp) < glQtdPadCorteDesemp Then
  ReDim Preserve mPadCorteDesemp(glQtdPadCorteDesemp + 10)
End If
mPadCorteDesemp(glQtdPadCorteDesemp) =
mPadCorteDesempTemp(codmelhorCritAgrup)
mPadCorteDesemp(glQtdPadCorteDesemp).CodPad = glQtdPadCorteDesemp

```

```
End Sub
```

```

Sub CalculaValor(ByVal ContObj As Long, ValorDat As Double, ValorOcup As Double,
ByVal j As Long)
'calcula avaliação de ocupação e avaliação de atendimento de datas

```

```
Dim i As Long
```

```
ValorOcup = 0: ValorDat = 0
```

```

QSmITEMPadCorteFDin 1, UBound(mPadCorteDesempTemp(j).mITEMPadCorte), j
For i = 1 To UBound(mPadCorteDesempTemp(j).mITEMPadCorte)

```

```

  Select Case mSEQUEDad(1).TipoProb
    Case "2DNG", "2DG"
      ValorOcup = ValorOcup + mPadCorteDesempTemp(j).mITEMPadCorte(i).Car1 *
mPadCorteDesempTemp(j).mITEMPadCorte(i).Car2
    Case "1D", "3D"
      ValorOcup = ValorOcup + mPadCorteDesempTemp(j).mITEMPadCorte(i).Car1
  End Select

```

```

  If mPadCorteDesempTemp(j).mITEMPadCorte(i).FDin < 0 Then
    ValorDat = ValorDat - mPadCorteDesempTemp(j).mITEMPadCorte(i).FDin * (0.2 *
((1 - 0.8) ^ (i - 1)))
  Else
    ValorDat = ValorDat + (1 / (2 +
mPadCorteDesempTemp(j).mITEMPadCorte(i).FDin)) * (0.2 * ((1 - 0.8) ^ (i - 1)))
  End If
Next

```

```

  Select Case mSEQUEDad(1).TipoProb
    Case "2DNG", "2DG"
      ValorOcup = ValorOcup / (mOBJETODad(ContObj).Car1 *
mOBJETODad(ContObj).Car2)
    Case "1D", "3D"
      ValorOcup = ValorOcup / (mOBJETODad(ContObj).Car1)
  End Select

```

```
End Sub
```

```
Sub GravamITEMDad()
```

Dim i, j, k As Long
Dim rg As Range

```
Set rg = Worksheets("SequeSai").Range("PadCortePrior")
For i = 1 To glQtdPadCorteDesemp
    rg.Cells(i + 2, 1) = mPadCorteDesemp(i).CodPad
    rg.Cells(i + 2, 2) = mPadCorteDesemp(i).CritAgrup
    rg.Cells(i + 2, 3) = mPadCorteDesemp(i).CodObj
    rg.Cells(i + 2, 4) = mPadCorteDesemp(i).CritOcup
    rg.Cells(i + 2, 5) = mPadCorteDesemp(i).CritData
    rg.Cells(i + 2, 6) = mPadCorteDesemp(i).CritOcupNorm
    rg.Cells(i + 2, 7) = mPadCorteDesemp(i).CritDataNorm
    rg.Cells(i + 2, 8) = mPadCorteDesemp(i).CritPond
    rg.Cells(i + 2, 9) = UBound(mPadCorteDesemp(i).mITEMPadCorte)
Next
```

```
Set rg = Worksheets("SequeSai").Range("PadCorte")
For i = 1 To glQtdPadCorteDesemp
    For j = 1 To UBound(mPadCorteDesemp(i).mITEMPadCorte)
        rg.Cells(k + j + 2, 1) = mPadCorteDesemp(i).CodPad
        rg.Cells(k + j + 2, 2) = mPadCorteDesemp(i).CodObj
        rg.Cells(k + j + 2, 3) = mPadCorteDesemp(i).mITEMPadCorte(j).CodItem
        rg.Cells(k + j + 2, 4) = mPadCorteDesemp(i).CritAgrup
    Next
    k = k + UBound(mPadCorteDesemp(i).mITEMPadCorte)
Next
```

End Sub

Sub Unidimensional(ByVal maxX As Double, iFx As Long, fFx As Long, r As Long)

'Dados de entrada:

- ' - tamanho do objeto unidimensional (maxX);
- ' - extremos da lista de itens ativos de um dado critério de agrupamento (iFx, fFx);
- ' - heurística que será utilizada (r).

'Saída: preenche a matriz mITEMPadCorte() com os CodItens alocados

```
Dim sorteio As Double, comp As Double, inativos As Long
Dim i As Long, j As Long, aux As Long, k As Long, auxReord As rITEMDad
```

```
glQtdITEMPadCorte = 0
inativos = 0
ReDim mITEMPadCorte(fFx - iFx + 1)
```

```
'Redimensiona mITEMLocal de acordo com o número de itens ativos
'de um determinado agrupamento
glQtdItemLocal = fFx - iFx + 1
ReDim mITEMLocal(glQtdItemLocal)
```

```
'Carrega mITEMLocal e já elimina itens que não cabem no objeto
For i = 1 To glQtdItemLocal
```

```

mITEMLocal(i) = mITEMDad(iFx + i - 1)
Next

Do While glQtdItemLocal > 0

  'Toma primeiro item conforme a heurística
  Randomize
  sorteio = Rnd()
  inativos = 0
  Select Case mHEURDad(r).ViesHeur
    'Heurística determinística, pega o primeiro
    Case "Determinística"
      j = 1
    'Heurística probabilística sem viés
    Case "Sem"
      j = 1
      comp = 1 / (glQtdItemLocal)
      Do While sorteio > comp
        j = j + 1
        comp = comp + 1 / glQtdItemLocal
      Loop
    'Heurística probabilística com viés linear
    Case "Linear"
      aux = 0
      For j = 1 To glQtdItemLocal - 1
        aux = aux + j
      Next
      j = 1
      comp = glQtdItemLocal / ((glQtdItemLocal ^ 2) - aux)
      Do While sorteio > comp
        j = j + 1
        comp = comp + (glQtdItemLocal - 1) / ((glQtdItemLocal ^ 2) - aux)
      Loop
    'Heurística probabilística com desproporcional
    Case "Desproporcional"
      j = 1
      If glQtdItemLocal > 1 Then
        comp = ((0.5) ^ (j - 1)) + ((0.5) ^ (glQtdItemLocal - 1)) * (((0.5) ^ (j - 1))) / (1 -
(0.5) ^ (glQtdItemLocal - 1))
      Else
        comp = 1
      End If
      Do While sorteio > comp
        j = j + 1
        comp = ((0.5) ^ (j - 1)) + ((0.5) ^ (glQtdItemLocal - 1)) * (((0.5) ^ (j - 1))) / (1 -
(0.5) ^ (glQtdItemLocal - 1))
      Loop
  End Select

  If mITEMLocal(j).Car1 <= maxX Then

```

```

'Coloca item escolhido em na matriz mITEMPadCorte
'Atualiza CritOcup da barra
maxX = maxX - mITEMLocal(j).Car1
glQtdITEMPadCorte = glQtdITEMPadCorte + 1
mITEMPadCorte(glQtdITEMPadCorte).CodItem = mITEMLocal(j).CodItem
mITEMPadCorte(glQtdITEMPadCorte).Car1 = mITEMLocal(j).Car1
mITEMPadCorte(glQtdITEMPadCorte).CodOP = mITEMLocal(j).CodOP
mITEMPadCorte(glQtdITEMPadCorte).FDin = mITEMLocal(j).FDin
mITEMLocal(j).AtivoSN = False
inativos = inativos + 1
auxReord = mITEMLocal(j)
For k = j To glQtdItemLocal - 1
    mITEMLocal(k) = mITEMLocal(k + 1)
Next
mITEMLocal(glQtdItemLocal) = auxReord
Else
'Atualiza status dos itens e elimina quem já não cabe mais
i = 1
Do While i <= glQtdItemLocal
    If mITEMLocal(i).AtivoSN = False Then
        Exit Do
    End If
    If mITEMLocal(i).Car1 > maxX Then
        mITEMLocal(i).AtivoSN = False
        inativos = inativos + 1
        auxReord = mITEMLocal(i)
        For k = i To glQtdItemLocal - 1
            mITEMLocal(k) = mITEMLocal(k + 1)
        Next
        mITEMLocal(glQtdItemLocal) = auxReord
    Else
        i = i + 1
    End If
Loop
End If
glQtdItemLocal = glQtdItemLocal - inativos
Loop

```

```

ReDim Preserve mITEMPadCorte(glQtdITEMPadCorte)

```

```

End Sub

```

```

Sub BidimensionalGuilhotinado(ByVal maxX As Double, ByVal maxY As Double, iFx As
Long, fFx As Long, r As Long)

```

```

'Dados de entrada:

```

```

' - CritOcup do objeto bidimensional (maxX, maxY);

```

```

' - extremos da lista de itens ativos de um dado critério de agrupamento (iFx, fFx);

```

```

' - heurística que será utilizada (r).

```

```

'Saída: preenche a matriz mPadCorte() com os CodItens alocados

```

```
'Auxiliares na escolha dos SubObjetos
Dim intervalo1 As Double, intervalo2 As Double
Dim Intervalo As Double, area As Double
Dim menorSubObjeto As Long, menorIntervalo As Double, menorArea As Double
```

```
'Auxiliares na atualização de mSubObjeto e mITEMDad
Dim inativos As Long, ok As Boolean, auxReord As rITEMDad
Dim rotL As Boolean, rotG As Boolean
```

```
'Auxiliares no sorteio probabilístico
Dim sorteio As Double, j As Long, aux As Long, comp As Double
```

```
'Auxiliar para eliminar subobjetos
Dim menorCar1 As Double, menorCar2 As Double, inativosSO As Long, auxSubObj As
rSubObjeto
```

```
'Contadores
Dim i, k As Long
```

```
glQtdITEMPadCorte = 0
inativos = 0
ReDim mITEMPadCorte(fFx - iFx + 1)
```

```
'Redimensiona mITEMLocal de acordo com o número de itens ativos
'de um determinado agrupamento
glQtdItemLocal = fFx - iFx + 1
ReDim mITEMLocal(glQtdItemLocal)
```

```
'Carrega mITEMLocal
For i = 1 To fFx - iFx + 1
    mITEMLocal(i) = mITEMDad(iFx + i - 1)
Next
```

```
'Carrega SubObjeto inicial que é o objeto todo
glQtdSubObjeto = 1
ReDim mSubObjeto(glQtdSubObjeto)
mSubObjeto(glQtdSubObjeto).Car1 = maxX
mSubObjeto(glQtdSubObjeto).Car2 = maxY
mSubObjeto(glQtdSubObjeto).AtivoSN = True
```

```
Do While glQtdItemLocal > 0 And glQtdSubObjeto > 0
```

```
    'Toma primeiro item conforme a heurística
    Randomize
    sorteio = Rnd()
    inativos = 0
    Select Case mHEURDad(r).ViesHeur
        'Heurística determinística, pega o primeiro
        Case "Determinística"
            j = 1
```

```

'Heurística probabilística sem viés
Case "Sem"
  j = 1
  comp = 1 / (glQtdItemLocal)
  Do While sorteio > comp
    j = j + 1
    comp = comp + 1 / glQtdItemLocal
  Loop
'Heurística probabilística com viés linear
Case "Linear"
  aux = 0
  For j = 1 To glQtdItemLocal - 1
    aux = aux + j
  Next
  j = 1
  comp = glQtdItemLocal / ((glQtdItemLocal ^ 2) - aux)
  Do While sorteio > comp
    j = j + 1
    comp = comp + (glQtdItemLocal - 1) / ((glQtdItemLocal ^ 2) - aux)
  Loop
'Heurística probabilística com desproporcional
Case "Desproporcional"
  j = 1
  If glQtdItemLocal > 1 Then
    comp = ((0.5) ^ (j - 1)) + ((0.5) ^ (glQtdItemLocal - 1)) * (((0.5) ^ (j - 1))) / (1 -
(0.5) ^ (glQtdItemLocal - 1)))
  Else
    comp = 1
  End If
  Do While sorteio > comp
    j = j + 1
    comp = ((0.5) ^ (j - 1)) + ((0.5) ^ (glQtdItemLocal - 1)) * (((0.5) ^ (j - 1))) / (1 -
(0.5) ^ (glQtdItemLocal - 1)))
  Loop
End Select

menorSubObjeto = 0
For i = 1 To glQtdSubObjeto
  'Só considera SubObjeto se o item couber nele, levando em conta que pode rotacionar
  If ((mITEMLocal(j).OrientSN = True) And (mITEMLocal(j).Car1 <=
mSubObjeto(i).Car1) And _
(mITEMLocal(j).Car2 <= mSubObjeto(i).Car2)) Or _
((mITEMLocal(j).OrientSN = False) And (((mITEMLocal(j).Car1 <=
mSubObjeto(i).Car1) And _
(mITEMLocal(j).Car2 <= mSubObjeto(i).Car2)) Or (mITEMLocal(j).Car2 <=
mSubObjeto(i).Car1) And _
(mITEMLocal(j).Car1 <= mSubObjeto(i).Car2)))) Then
    Intervalo = maxX + maxY

```

```

    If (mITEMLocal(j).Car1 <= mSubObjeto(i).Car1) And (mITEMLocal(j).Car2 <=
mSubObjeto(i).Car2) Then
        intervalo1 = mSubObjeto(i).Car1 - mITEMLocal(j).Car1
        intervalo2 = mSubObjeto(i).Car2 - mITEMLocal(j).Car2
        If intervalo1 < intervalo2 Then
            Intervalo = intervalo1
        Else
            Intervalo = intervalo2
        End If
        rotL = False
    End If
    If (mITEMLocal(j).OrientSN = False) And ((mITEMLocal(j).Car2 <=
mSubObjeto(i).Car1) And (mITEMLocal(j).Car1 <= mSubObjeto(i).Car2)) And _
        ((mSubObjeto(i).Car1 - mITEMLocal(j).Car2) < Intervalo Or
(mSubObjeto(i).Car2 - mITEMLocal(j).Car1) < Intervalo) Then
        intervalo1 = mSubObjeto(i).Car2 - mITEMLocal(j).Car1
        intervalo2 = mSubObjeto(i).Car1 - mITEMLocal(j).Car2
        If intervalo1 < intervalo2 Then
            Intervalo = intervalo1
        Else
            Intervalo = intervalo2
        End If
        rotL = True
    End If

    area = mSubObjeto(i).Car1 * mSubObjeto(i).Car2
    If menorSubObjeto = 0 Or Intervalo < menorIntervalo Or _
        (Intervalo = menorIntervalo And area < menorArea) Then
        menorSubObjeto = i
        menorIntervalo = Intervalo
        menorArea = area
        rotG = rotL
    End If

End If
Next

If menorSubObjeto <> 0 Then
'Coloca item escolhido em na matriz mITEMPadCorte
glQtdITEMPadCorte = glQtdITEMPadCorte + 1
mITEMPadCorte(glQtdITEMPadCorte).CodItem = mITEMLocal(j).CodItem
mITEMPadCorte(glQtdITEMPadCorte).Car1 = mITEMLocal(j).Car1
mITEMPadCorte(glQtdITEMPadCorte).Car2 = mITEMLocal(j).Car2
mITEMPadCorte(glQtdITEMPadCorte).CodOP = mITEMLocal(j).CodOP
mITEMPadCorte(glQtdITEMPadCorte).FDin = mITEMLocal(j).FDin

AtualizaSubObjetos menorSubObjeto, rotG, j

mITEMLocal(j).AtivoSN = False
inativos = inativos + 1

```

```

auxReord = mITEMLocal(j)
For k = j To glQtdItemLocal - 1
  mITEMLocal(k) = mITEMLocal(k + 1)
Next
mITEMLocal(glQtdItemLocal) = auxReord
Else
'Marcar itens inativos
i = 1
Do Until i > glQtdItemLocal Or mITEMLocal(i).AtivoSN = False
  ok = VerifAtivo(i)
  If ok = False Then
    mITEMLocal(i).AtivoSN = False
    inativos = inativos + 1
    auxReord = mITEMLocal(i)
    For k = i To glQtdItemLocal - 1
      mITEMLocal(k) = mITEMLocal(k + 1)
    Next
    mITEMLocal(glQtdItemLocal) = auxReord
    If i = j Then
      j = glQtdItemLocal + 1
    End If
  Else
    i = i + 1
  End If
Loop

'Elimina subobjetos em que não cabe nem mesmo o menor item
menoresDim menorCar1, menorCar2
inativosSO = 0
i = 1
Do Until i > glQtdSubObjeto Or mSubObjeto(i).AtivoSN = False
  If mSubObjeto(i).Car1 < menorCar1 Or mSubObjeto(i).Car2 < menorCar2 Then
    mSubObjeto(i).AtivoSN = False
    auxSubObj = mSubObjeto(i)
    mSubObjeto(i) = mSubObjeto(glQtdSubObjeto)
    mSubObjeto(glQtdSubObjeto) = auxSubObj
    glQtdSubObjeto = glQtdSubObjeto - 1
  Else
    i = i + 1
  End If
Loop

End If
glQtdItemLocal = glQtdItemLocal - inativos

Loop

ReDim Preserve mITEMPadCorte(glQtdITEMPadCorte)

End Sub

```

Sub AtualizaSubObjetos(ByVal i As Long, ByVal rot As Boolean, ByVal j As Long)
'Atualiza matriz mSubObjetos

Dim desig1 As Double, desig2 As Double, auxITEM As Double

If rot = True Then

auxITEM = mITEMLocal(j).Car1
mITEMLocal(j).Car1 = mITEMLocal(j).Car2
mITEMLocal(j).Car2 = auxITEM

End If

If mSubObjeto(i).Car1 = mITEMLocal(j).Car1 Then

If mSubObjeto(i).Car2 = mITEMLocal(j).Car2 Then
mSubObjeto(i) = mSubObjeto(glQtdSubObjeto)
glQtdSubObjeto = glQtdSubObjeto - 1
mSubObjeto(glQtdSubObjeto).AtivoSN = False

Else

mSubObjeto(i).Car2 = mSubObjeto(i).Car2 - mITEMLocal(j).Car2

End If

Else

If mSubObjeto(i).Car2 = mITEMLocal(j).Car2 Then

mSubObjeto(i).Car1 = mSubObjeto(i).Car1 - mITEMLocal(j).Car1

Else

$$\text{desig1} = ((\text{mSubObjeto}(i).\text{Car1} - \text{mITEMLocal}(j).\text{Car1}) * (\text{mITEMLocal}(j).\text{Car2})) ^ 2 _ \\ + ((\text{mSubObjeto}(i).\text{Car2} - \text{mITEMLocal}(j).\text{Car2}) * (\text{mSubObjeto}(i).\text{Car1})) ^ 2$$

$$\text{desig2} = ((\text{mSubObjeto}(i).\text{Car1} - \text{mITEMLocal}(j).\text{Car1}) * (\text{mSubObjeto}(i).\text{Car2})) ^ 2 _ \\ + ((\text{mSubObjeto}(i).\text{Car2} - \text{mITEMLocal}(j).\text{Car2}) * (\text{mITEMLocal}(j).\text{Car1})) ^ 2$$

glQtdSubObjeto = glQtdSubObjeto + 1

If UBound(mSubObjeto) < glQtdSubObjeto Then

ReDim Preserve mSubObjeto(glQtdSubObjeto + 20)

End If

If desig1 > desig2 Then

mSubObjeto(glQtdSubObjeto).Car1 = mSubObjeto(i).Car1
mSubObjeto(glQtdSubObjeto).Car2 = mSubObjeto(i).Car2 - mITEMLocal(j).Car2
mSubObjeto(glQtdSubObjeto).AtivoSN = True

mSubObjeto(i).Car1 = mSubObjeto(i).Car1 - mITEMLocal(j).Car1

mSubObjeto(i).Car2 = mITEMLocal(j).Car2

Else

mSubObjeto(glQtdSubObjeto).Car1 = mITEMLocal(j).Car1

mSubObjeto(glQtdSubObjeto).Car2 = mSubObjeto(i).Car2 - mITEMLocal(j).Car2

mSubObjeto(glQtdSubObjeto).AtivoSN = True

mSubObjeto(i).Car1 = mSubObjeto(i).Car1 - mITEMLocal(j).Car1

```

        mSubObjeto(i).Car2 = mSubObjeto(i).Car2
    End If
End If
End If

```

```
End Sub
```

```
Function VerifAtivo(ByVal i As Long) As Boolean
'Verifica se um item ainda cabe em algum subobjeto
```

```
Dim k As Long
```

```

    VerifAtivo = False
    k = 1
    Do While k <= glQtdSubObjeto
        If ((mITEMLocal(i).OrientSN = True And mITEMLocal(i).Car1 <=
mSubObjeto(k).Car1 And mITEMLocal(i).Car2 <= mSubObjeto(k).Car2) Or _
        (mITEMLocal(i).OrientSN = False And _
        ((mITEMLocal(i).Car1 <= mSubObjeto(k).Car1 And mITEMLocal(i).Car2 <=
mSubObjeto(k).Car2) Or _
        (mITEMLocal(i).Car2 <= mSubObjeto(k).Car1 And mITEMLocal(i).Car1 <=
mSubObjeto(k).Car2)))) Then
            VerifAtivo = True
            Exit Do
        End If
        k = k + 1
    Loop

```

```
End Function
```

```
Sub menoresDim(menorCar1 As Double, menorCar2 As Double)
'Verifica as menores dimensões em x e y
```

```
Dim i As Long
```

```

For i = 1 To glQtdItemLocal
    If i = 1 Then
        If mITEMLocal(i).OrientSN = True Then
            menorCar1 = mITEMLocal(i).Car1
            menorCar2 = mITEMLocal(i).Car2
        Else
            If mITEMLocal(i).Car1 < mITEMLocal(i).Car2 Then
                menorCar1 = mITEMLocal(i).Car1
                menorCar2 = mITEMLocal(i).Car1
            Else
                menorCar1 = mITEMLocal(i).Car2
                menorCar2 = mITEMLocal(i).Car2
            End If
        End If
    End If
Else

```

```

If mITEMLocal(i).OrientSN = True Then
  If mITEMLocal(i).Car1 < menorCar1 Then
    menorCar1 = mITEMLocal(i).Car1
  End If
  If mITEMLocal(i).Car2 < menorCar2 Then
    menorCar2 = mITEMLocal(i).Car2
  End If
Else
  If mITEMLocal(i).Car1 < mITEMLocal(i).Car2 Then
    If mITEMLocal(i).Car1 < menorCar1 Then
      menorCar1 = mITEMLocal(i).Car1
    End If
    If mITEMLocal(i).Car1 < menorCar2 Then
      menorCar2 = mITEMLocal(i).Car1
    End If
  Else
    If mITEMLocal(i).Car2 < menorCar1 Then
      menorCar1 = mITEMLocal(i).Car2
    End If
    If mITEMLocal(i).Car2 < menorCar2 Then
      menorCar2 = mITEMLocal(i).Car2
    End If
  End If
End If
End If
End If
Next

```

End Sub

```

Sub BidimensionalNaoGuilhotinado(ByVal maxX As Double, ByVal maxY As Double, iFx
As Long, fFx As Long, r As Long)

```

```

  Dim inativos As Long

```

```

  Dim j As Long, i As Long, auxReord As rITEMDad

```

```

  glQtdItensPos = 0: glQtdITEMPadCorte = 0: inativos = 0

```

```

  glQtdItemLocal = fFx - iFx + 1

```

```

  ReDim mITEMLocal(glQtdItemLocal)

```

```

  ReDim mItensPos(fFx - iFx + 1)

```

```

  'Carrega mITEMLocal

```

```

  For i = 1 To fFx - iFx + 1

```

```

    mITEMLocal(i) = mITEMDad(iFx + i - 1)

```

```

  Next

```

```

  'Carrega matriz de cantos iniciais

```

```

  glQtdCantos = 4

```

```

  ReDim mCantos(4)

```

```

  mCantos(1).X = 0

```

```

mCantos(1).Y = 0
mCantos(1).tipo = "EB"
mCantos(1).Ordem = 1
mCantos(1).arestaX = maxX
mCantos(1).arestaY = maxY

```

```

mCantos(2).X = maxX
mCantos(2).Y = 0
mCantos(2).tipo = "DB"
mCantos(2).Ordem = 2
mCantos(2).arestaX = maxX
mCantos(2).arestaY = maxY

```

```

mCantos(3).X = 0
mCantos(3).Y = maxY
mCantos(3).tipo = "EA"
mCantos(3).Ordem = 3
mCantos(3).arestaX = maxX
mCantos(3).arestaY = maxY

```

```

mCantos(4).X = maxX
mCantos(4).Y = maxY
mCantos(4).tipo = "DA"
mCantos(4).Ordem = 4
mCantos(4).arestaX = maxX
mCantos(4).arestaY = maxY

```

Do While glQtdItemLocal > 0

```

'Toma primeiro item conforme a heurística
inativos = 0
Randomize
sorteio = Rnd()
Select Case mHEURDad(r).ViesHeur
  'Heurística determinística, pega o primeiro
  Case "Determinística"
    j = 1
  'Heurística probabilística sem viés
  Case "Sem"
    j = 1
    comp = 1 / (glQtdItemLocal)
    Do While sorteio > comp
      j = j + 1
      comp = comp + 1 / glQtdItemLocal
    Loop
  'Heurística probabilística com viés linear
  Case "Linear"
    aux = 0
    For j = 1 To glQtdItemLocal - 1
      aux = aux + j
    Next j
  End Select

```

```

Next
j = 1
comp = glQtdItemLocal / ((glQtdItemLocal ^ 2) - aux)
Do While sorteio > comp
  j = j + 1
  comp = comp + (glQtdItemLocal - 1) / ((glQtdItemLocal ^ 2) - aux)
Loop
'Heurística probabilística com desproporcional
Case "Desproporcional"
  j = 1
  If glQtdItemLocal > 1 Then
    comp = ((0.5) ^ (j - 1)) + ((0.5) ^ (glQtdItemLocal - 1)) * (((0.5) ^ (j - 1))) / (1 -
(0.5) ^ (glQtdItemLocal - 1)))
  Else
    comp = 1
  End If
  Do While sorteio > comp
    j = j + 1
    comp = ((0.5) ^ (j - 1)) + ((0.5) ^ (glQtdItemLocal - 1)) * (((0.5) ^ (j - 1))) / (1 -
(0.5) ^ (glQtdItemLocal - 1)))
  Loop
End Select
If j > glQtdItemLocal Then j = 1

glQtdCOPMs = PossibilidadeEncaixe(j, maxX, maxY)

If glQtdCOPMs > 0 Then
  For k = 1 To glQtdCOPMs

    mCOPMs(k).FFV = 0
    If mCantos(mCOPMs(k).canto).arestaX > mCOPMs(k).xp Then
      mCOPMs(k).FFV = mCOPMs(k).FFV + mCOPMs(k).xp
    Else
      mCOPMs(k).FFV = mCOPMs(k).FFV + mCantos(mCOPMs(k).canto).arestaX
    End If
    If mCantos(mCOPMs(k).canto).arestaY > mCOPMs(k).yp Then
      mCOPMs(k).FFV = mCOPMs(k).FFV + mCOPMs(k).yp
    Else
      mCOPMs(k).FFV = mCOPMs(k).FFV + mCantos(mCOPMs(k).canto).arestaY
    End If
    mCOPMs(k).FFV = mCOPMs(k).FFV / (mCOPMs(k).yp + mCOPMs(k).xp)

    If k = 1 Then
      maiorCOPM = k
    Else
      If mCOPMs(k).FFV >= mCOPMs(maiorCOPM).FFV Then
        maiorCOPM = k
      End If
    End If
  End If
Next

```

```

glQtdItensPos = glQtdItensPos + 1
If UBound(mItensPos) < glQtdItensPos Then
  ReDim Preserve mItensPos(glQtdItensPos + 10)
End If
mItensPos(glQtdItensPos).CodItem = mITEMLocal(j).CodItem
mItensPos(glQtdItensPos).CodOP = mITEMLocal(j).CodOP
mItensPos(glQtdItensPos).FDin = mITEMLocal(j).FDin
mItensPos(glQtdItensPos).xi = mCOPMs(maiorCOPM).xc
mItensPos(glQtdItensPos).X = mCOPMs(maiorCOPM).xp
mItensPos(glQtdItensPos).yi = mCOPMs(maiorCOPM).yc
mItensPos(glQtdItensPos).Y = mCOPMs(maiorCOPM).yp

atualizaCantos maxX, maxY

mITEMLocal(j).AtivoSN = False
inativos = inativos + 1
auxReord = mITEMLocal(j)
For k = j To glQtdItemLocal - 1
  mITEMLocal(k) = mITEMLocal(k + 1)
Next
mITEMLocal(glQtdItemLocal) = auxReord

Else
  'Marcar itens inativos
  i = 1
  Do Until i > glQtdItemLocal Or mITEMLocal(i).AtivoSN = False
    glQtdCOPMs = PossibilidadeEncaixe(i, maxX, maxY)
    If glQtdCOPMs = 0 Then
      mITEMLocal(i).AtivoSN = False
      inativos = inativos + 1
      auxReord = mITEMLocal(i)
      For k = i To glQtdItemLocal - 1
        mITEMLocal(k) = mITEMLocal(k + 1)
      Next
      mITEMLocal(glQtdItemLocal) = auxReord
    Else
      i = i + 1
    End If
  Loop
End If
glQtdItemLocal = glQtdItemLocal - inativos
Loop

ReDim mITEMPadCorte(glQtdItensPos)
glQtdITEMPadCorte = glQtdItensPos
For i = 1 To glQtdItensPos
  mITEMPadCorte(i).CodItem = mItensPos(i).CodItem
  mITEMPadCorte(i).Car1 = mItensPos(i).X
  mITEMPadCorte(i).Car2 = mItensPos(i).Y

```

```

    mITEMPadCorte(i).CodOP = mItensPos(i).CodOP
    mITEMPadCorte(i).FDin = mItensPos(i).FDin
Next

End Sub

Sub atualizaCantos(ByVal maxX As Double, ByVal maxY As Double)

Dim cod As Long, i As Long, menorX As Double, menorY As Double
Dim CantosProv() As rCantos
Dim m, aux, aux2, comp As Long
Dim glQtdCantosProv As Long

glQtdCantosProv = 0
ReDim CantosProv(glQtdCantosProv + 20)
mCantos(mCOPMs(maiorCOPM).canto).tipo = "ZP"

For m = 1 To glQtdItensPos - 1

    'Caso de peças com aresta coincidente por baixo do novo item posicionado
    If (mItensPos(m).yi + mItensPos(m).Y) = mCOPMs(maiorCOPM).yc Then
        If (mItensPos(m).xi > mCOPMs(maiorCOPM).xc) And _
            (mItensPos(m).xi <= mCOPMs(maiorCOPM).xc + mCOPMs(maiorCOPM).xp)
Then
            glQtdCantosProv = glQtdCantosProv + 1
            If UBound(CantosProv) < glQtdCantosProv Then
                ReDim Preserve CantosProv(glQtdCantosProv + 20)
            End If
            CantosProv(glQtdCantosProv).X = mItensPos(m).xi
            CantosProv(glQtdCantosProv).Y = mItensPos(m).yi + mItensPos(m).Y
            CantosProv(glQtdCantosProv).tipo = "DA"
        End If
        If (mItensPos(m).xi + mItensPos(m).X < mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp) And _
            (mItensPos(m).xi + mItensPos(m).X >= mCOPMs(maiorCOPM).xc) Then
            glQtdCantosProv = glQtdCantosProv + 1
            If UBound(CantosProv) < glQtdCantosProv Then
                ReDim Preserve CantosProv(glQtdCantosProv + 20)
            End If
            CantosProv(glQtdCantosProv).X = mItensPos(m).xi + mItensPos(m).X
            CantosProv(glQtdCantosProv).Y = mItensPos(m).yi + mItensPos(m).Y
            CantosProv(glQtdCantosProv).tipo = "EA"
        End If
        If (mItensPos(m).xi < mCOPMs(maiorCOPM).xc) And _
            (mItensPos(m).xi + mItensPos(m).X >= mCOPMs(maiorCOPM).xc) Then
            glQtdCantosProv = glQtdCantosProv + 1
            If UBound(CantosProv) < glQtdCantosProv Then
                ReDim Preserve CantosProv(glQtdCantosProv + 20)
            End If
            CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc

```

```

    CantosProv(glQtdCantosProv).Y = mItensPos(m).yi + mItensPos(m).Y
    CantosProv(glQtdCantosProv).tipo = "DB"
  End If
  If (mItensPos(m).xi + mItensPos(m).X > mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp) And _
    (mItensPos(m).xi <= mCOPMs(maiorCOPM).xc + mCOPMs(maiorCOPM).xp)
  Then
    glQtdCantosProv = glQtdCantosProv + 1
    If UBound(CantosProv) < glQtdCantosProv Then
      ReDim Preserve CantosProv(glQtdCantosProv + 20)
    End If
    CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp
    CantosProv(glQtdCantosProv).Y = mItensPos(m).yi + mItensPos(m).Y
    CantosProv(glQtdCantosProv).tipo = "EB"
  End If
End If

'Caso de peças com aresta coincidente por cima do novo item posicionado
If (mItensPos(m).yi) = mCOPMs(maiorCOPM).yc + mCOPMs(maiorCOPM).yp Then
  If (mItensPos(m).xi > mCOPMs(maiorCOPM).xc) And _
    (mItensPos(m).xi <= mCOPMs(maiorCOPM).xc + mCOPMs(maiorCOPM).xp)
  Then
    glQtdCantosProv = glQtdCantosProv + 1
    If UBound(CantosProv) < glQtdCantosProv Then
      ReDim Preserve CantosProv(glQtdCantosProv + 20)
    End If
    CantosProv(glQtdCantosProv).X = mItensPos(m).xi
    CantosProv(glQtdCantosProv).Y = mItensPos(m).yi
    CantosProv(glQtdCantosProv).tipo = "DB"
  End If
  If (mItensPos(m).xi + mItensPos(m).X < mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp) And _
    (mItensPos(m).xi + mItensPos(m).X >= mCOPMs(maiorCOPM).xc) Then
    glQtdCantosProv = glQtdCantosProv + 1
    If UBound(CantosProv) < glQtdCantosProv Then
      ReDim Preserve CantosProv(glQtdCantosProv + 20)
    End If
    CantosProv(glQtdCantosProv).X = mItensPos(m).xi + mItensPos(m).X
    CantosProv(glQtdCantosProv).Y = mItensPos(m).yi
    CantosProv(glQtdCantosProv).tipo = "EB"
  End If
  If (mItensPos(m).xi < mCOPMs(maiorCOPM).xc) And _
    (mItensPos(m).xi + mItensPos(m).X >= mCOPMs(maiorCOPM).xc) Then
    glQtdCantosProv = glQtdCantosProv + 1
    If UBound(CantosProv) < glQtdCantosProv Then
      ReDim Preserve CantosProv(glQtdCantosProv + 20)
    End If
    CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc
    CantosProv(glQtdCantosProv).Y = mItensPos(m).yi
  End If

```

```

        CantosProv(glQtdCantosProv).tipo = "DA"
    End If
    If (mItensPos(m).xi + mItensPos(m).X > mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp) And _
        (mItensPos(m).xi <= mCOPMs(maiorCOPM).xc + mCOPMs(maiorCOPM).xp)
    Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp
        CantosProv(glQtdCantosProv).Y = mItensPos(m).yi
        CantosProv(glQtdCantosProv).tipo = "EA"
    End If
End If

' Caso de peças com aresta coincidente pela esquerda do novo item posicionado
If (mItensPos(m).xi) = mCOPMs(maiorCOPM).xc + mCOPMs(maiorCOPM).xp Then
    If (mItensPos(m).yi > mCOPMs(maiorCOPM).yc) And _
        (mItensPos(m).yi <= mCOPMs(maiorCOPM).yc + mCOPMs(maiorCOPM).yp)
    Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mItensPos(m).xi
        CantosProv(glQtdCantosProv).Y = mItensPos(m).yi
        CantosProv(glQtdCantosProv).tipo = "EA"
    End If
    If (mItensPos(m).yi + mItensPos(m).Y < mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp) And _
        (mItensPos(m).yi + mItensPos(m).Y >= mCOPMs(maiorCOPM).yc) Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mItensPos(m).xi
        CantosProv(glQtdCantosProv).Y = mItensPos(m).yi + mItensPos(m).Y
        CantosProv(glQtdCantosProv).tipo = "EB"
    End If
    If (mItensPos(m).yi + mItensPos(m).Y > mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp) And _
        (mItensPos(m).yi <= mCOPMs(maiorCOPM).yc + mCOPMs(maiorCOPM).yp)
    Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mItensPos(m).xi

```

```

        CantosProv(glQtdCantosProv).Y      =      mCOPMs(maiorCOPM).yc      +
mCOPMs(maiorCOPM).yp
        CantosProv(glQtdCantosProv).tipo = "DB"
    End If
    If (mItensPos(m).yi < mCOPMs(maiorCOPM).yc) And _
        (mItensPos(m).yi + mItensPos(m).Y >= mCOPMs(maiorCOPM).yc) Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mItensPos(m).xi
        CantosProv(glQtdCantosProv).Y = mCOPMs(maiorCOPM).yc
        CantosProv(glQtdCantosProv).tipo = "DA"
    End If
End If

'Caso de peças com aresta coincidente pela direita novo item posicionado
If (mItensPos(m).xi + mItensPos(m).X) = mCOPMs(maiorCOPM).xc Then
    If (mItensPos(m).yi > mCOPMs(maiorCOPM).yc) And _
        (mItensPos(m).yi <= mCOPMs(maiorCOPM).yc + mCOPMs(maiorCOPM).yp)
Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mItensPos(m).xi + mItensPos(m).X
        CantosProv(glQtdCantosProv).Y = mItensPos(m).yi
        CantosProv(glQtdCantosProv).tipo = "DA"
    End If
    If (mItensPos(m).yi + mItensPos(m).Y < mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp) And _
        (mItensPos(m).yi + mItensPos(m).Y >= mCOPMs(maiorCOPM).yc) Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mItensPos(m).xi + mItensPos(m).X
        CantosProv(glQtdCantosProv).Y = mItensPos(m).yi + mItensPos(m).Y
        CantosProv(glQtdCantosProv).tipo = "DB"
    End If
    If (mItensPos(m).yi < mCOPMs(maiorCOPM).yc) And _
        (mItensPos(m).Y + mItensPos(m).yi >= mCOPMs(maiorCOPM).yc) Then
        glQtdCantosProv = glQtdCantosProv + 1
        If UBound(CantosProv) < glQtdCantosProv Then
            ReDim Preserve CantosProv(glQtdCantosProv + 20)
        End If
        CantosProv(glQtdCantosProv).X = mItensPos(m).xi + mItensPos(m).X
        CantosProv(glQtdCantosProv).Y = mCOPMs(maiorCOPM).yc
        CantosProv(glQtdCantosProv).tipo = "EA"
    End If

```

```

    If (mItensPos(m).yi + mItensPos(m).Y > mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp) And _
        (mItensPos(m).yi <= mCOPMs(maiorCOPM).yc + mCOPMs(maiorCOPM).yp)
Then
    glQtdCantosProv = glQtdCantosProv + 1
    If UBound(CantosProv) < glQtdCantosProv Then
        ReDim Preserve CantosProv(glQtdCantosProv + 20)
    End If
    CantosProv(glQtdCantosProv).X = mItensPos(m).xi + mItensPos(m).X
    CantosProv(glQtdCantosProv).Y = mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp
    CantosProv(glQtdCantosProv).tipo = "EB"
    End If
End If
Next

' Caso de novos itens posicionados em arestas
'Aresta de horizontal de baixo
If mCOPMs(maiorCOPM).yc = 0 Then
    glQtdCantosProv = glQtdCantosProv + 2
    If UBound(CantosProv) < glQtdCantosProv Then
        ReDim Preserve CantosProv(glQtdCantosProv + 20)
    End If
    CantosProv(glQtdCantosProv - 1).X = mCOPMs(maiorCOPM).xc
    CantosProv(glQtdCantosProv - 1).Y = mCOPMs(maiorCOPM).yc
    CantosProv(glQtdCantosProv - 1).tipo = "DB"
    CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp
    CantosProv(glQtdCantosProv).Y = mCOPMs(maiorCOPM).yc
    CantosProv(glQtdCantosProv).tipo = "EB"
End If
'Aresta de horizontal de cima
If mCOPMs(maiorCOPM).yc + mCOPMs(maiorCOPM).yp = maxY Then
    glQtdCantosProv = glQtdCantosProv + 2
    If UBound(CantosProv) < glQtdCantosProv Then
        ReDim Preserve CantosProv(glQtdCantosProv + 20)
    End If
    CantosProv(glQtdCantosProv - 1).X = mCOPMs(maiorCOPM).xc
    CantosProv(glQtdCantosProv - 1).Y = mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp
    CantosProv(glQtdCantosProv - 1).tipo = "DA"
    CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp
    CantosProv(glQtdCantosProv).Y = mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp
    CantosProv(glQtdCantosProv).tipo = "EA"
End If
'Aresta de vertical da esquerda
If mCOPMs(maiorCOPM).xc = 0 Then
    glQtdCantosProv = glQtdCantosProv + 2

```

```

If UBound(CantosProv) < glQtdCantosProv Then
  ReDim Preserve CantosProv(glQtdCantosProv + 20)
End If
CantosProv(glQtdCantosProv - 1).X = mCOPMs(maiorCOPM).xc
CantosProv(glQtdCantosProv - 1).Y = mCOPMs(maiorCOPM).yc
CantosProv(glQtdCantosProv - 1).tipo = "EA"
CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc
CantosProv(glQtdCantosProv).Y      =      mCOPMs(maiorCOPM).yc      +
mCOPMs(maiorCOPM).yp
CantosProv(glQtdCantosProv).tipo = "EB"
End If
'Aresta de vertical da direita
If mCOPMs(maiorCOPM).xc + mCOPMs(maiorCOPM).xp = maxX Then
  glQtdCantosProv = glQtdCantosProv + 2
  If UBound(CantosProv) < glQtdCantosProv Then
    ReDim Preserve CantosProv(glQtdCantosProv + 20)
  End If
  CantosProv(glQtdCantosProv - 1).X = mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp
  CantosProv(glQtdCantosProv - 1).Y = mCOPMs(maiorCOPM).yc
  CantosProv(glQtdCantosProv - 1).tipo = "DA"
  CantosProv(glQtdCantosProv).X = mCOPMs(maiorCOPM).xc +
mCOPMs(maiorCOPM).xp
  CantosProv(glQtdCantosProv).Y = mCOPMs(maiorCOPM).yc +
mCOPMs(maiorCOPM).yp
  CantosProv(glQtdCantosProv).tipo = "DB"
End If

glQtdCantos = glQtdCantos + glQtdCantosProv
ReDim Preserve mCantos(glQtdCantos)
If glQtdCantosProv > 0 Then
  For m = 1 To glQtdCantosProv
    CantosProv(m).Ordem = glQtdCantos - glQtdCantosProv + m
    mCantos(glQtdCantos - glQtdCantosProv + m) = CantosProv(m)
  Next
End If

'Ordena em ordem crescente de x e depois de y
QSChvCompCantosXY 1, glQtdCantos

'Elimina cantos inexistentes para limpar o sistema
For m = 1 To glQtdCantos
  Select Case mCantos(m).tipo
    Case "EB"
      If mCantos(m).Y = maxY Then
        mCantos(m).tipo = "ZF"
      Else
        If mCantos(m).X = maxX Then
          mCantos(m).tipo = "ZF"
        End If
      End If
    End Select
  Next m
End For

```

```

    End If
  Case "EA"
    If mCantos(m).Y = 0 Then
      mCantos(m).tipo = "ZF"
    Else
      If mCantos(m).X = maxX Then
        mCantos(m).tipo = "ZF"
      End If
    End If
  Case "DB"
    If mCantos(m).Y = maxY Then
      mCantos(m).tipo = "ZF"
    Else
      If mCantos(m).X = 0 Then
        mCantos(m).tipo = "ZF"
      End If
    End If
  Case "DA"
    If mCantos(m).Y = 0 Then
      mCantos(m).tipo = "ZF"
    Else
      If mCantos(m).X = 0 Then
        mCantos(m).tipo = "ZF"
      End If
    End If
  End Select
Next

For m = 1 To glQtdCantos - 1
  If m < glQtdCantos Then
    If mCantos(m).X = mCantos(m + 1).X And mCantos(m).Y = mCantos(m + 1).Y Then
      Select Case mCantos(m).tipo
        Case "EA"
          Select Case mCantos(m + 1).tipo
            Case "EA"
              mCantos(m).tipo = "ZR"
            Case "EB", "DA", "ZP"
              mCantos(m).tipo = "ZA"
              mCantos(m + 1).tipo = "ZA"
          End Select
        Case "EB"
          Select Case mCantos(m + 1).tipo
            Case "EB"
              mCantos(m).tipo = "ZR"
            Case "EA", "DB", "ZP"
              mCantos(m).tipo = "ZA"
              mCantos(m + 1).tipo = "ZA"
          End Select
        Case "DA"
          Select Case mCantos(m + 1).tipo

```

```

        Case "DA"
            mCantos(m).tipo = "ZR"
        Case "DB", "EA", "ZP"
            mCantos(m).tipo = "ZA"
            mCantos(m + 1).tipo = "ZA"
        End Select
    Case "DB"
        Select Case mCantos(m + 1).tipo
            Case "DB"
                mCantos(m).tipo = "ZR"
            Case "EB", "DA", "ZP"
                mCantos(m).tipo = "ZA"
                mCantos(m + 1).tipo = "ZA"
        End Select
    Case "ZP"
        mCantos(m + 1).tipo = "ZR"
    Case "ZA"
        mCantos(m + 1).tipo = "ZA"
    End Select
End If
End If
Next

aux = 0
For m = 1 To glQtdCantos
    If mCantos(m).tipo = "ZP" Or mCantos(m).tipo = "ZF" Or _
        mCantos(m).tipo = "ZR" Or mCantos(m).tipo = "ZA" Then
        aux = aux + 1
    End If
Next

QSmCantosTipoA 1, glQtdCantos

glQtdCantos = glQtdCantos - aux
ReDim Preserve mCantos(glQtdCantos)

'Calcula aresta livre do canto
For m = 1 To glQtdCantos
    mCantos(m).Ordem = m
    Select Case mCantos(m).tipo
        Case "EB"
            menorX = maxX
            menorY = maxY
            For i = 1 To glQtdItensPos
                If ((mItensPos(i).yi + mItensPos(i).Y) >= mCantos(m).Y) And _
                    (mItensPos(i).yi <= mCantos(m).Y) And _
                    ((mItensPos(i).xi + mItensPos(i).X) > mCantos(m).X) Then
                    If mItensPos(i).xi < menorX And mItensPos(i).xi > mCantos(m).X Then
                        menorX = mItensPos(i).xi
                    End If
                End If
            Next
        End Select
    Next
End For

```

```

ElseIf ((mItensPos(i).xi + mItensPos(i).X) > mCantos(m).X) And (mItensPos(i).xi
< menorX) Then
    menorX = mItensPos(i).xi + mItensPos(i).X
End If
End If
If ((mItensPos(i).xi + mItensPos(i).X) >= mCantos(m).X) And _
(mItensPos(i).xi <= mCantos(m).X) And _
((mItensPos(i).yi + mItensPos(i).Y) > mCantos(m).Y) Then
    If mItensPos(i).yi < menorY And mItensPos(i).yi > mCantos(m).Y Then
        menorY = mItensPos(i).yi
    ElseIf ((mItensPos(i).yi + mItensPos(i).Y) > mCantos(m).Y) And (mItensPos(i).yi
< menorY) Then
        menorY = mItensPos(i).yi + mItensPos(i).Y
    End If
End If
Next
mCantos(m).arestaX = menorX - mCantos(m).X
mCantos(m).arestaY = menorY - mCantos(m).Y
Case "EA"
menorX = maxX
menorY = 0
For i = 1 To glQtdItensPos
    If ((mItensPos(i).yi + mItensPos(i).Y) >= mCantos(m).Y) And _
(mItensPos(i).yi <= mCantos(m).Y) And _
((mItensPos(i).xi + mItensPos(i).X) > mCantos(m).X) Then
        If mItensPos(i).xi < menorX And mItensPos(i).xi > mCantos(m).X Then
            menorX = mItensPos(i).xi
        ElseIf ((mItensPos(i).xi + mItensPos(i).X) > mCantos(m).X) And (mItensPos(i).xi
< menorX) Then
            menorX = mItensPos(i).xi + mItensPos(i).X
        End If
    End If
    If ((mItensPos(i).xi + mItensPos(i).X) >= mCantos(m).X) And _
(mItensPos(i).xi <= mCantos(m).X) And _
(mItensPos(i).yi < mCantos(m).Y) Then
        If ((mItensPos(i).yi + mItensPos(i).Y) < mCantos(m).Y) And ((mItensPos(i).yi +
mItensPos(i).Y) > menorY) Then
            menorY = mItensPos(i).yi + mItensPos(i).Y
        ElseIf (mItensPos(i).yi < mCantos(m).Y) And (mItensPos(i).yi > menorY) Then
            menorY = mItensPos(i).yi
        End If
    End If
Next
mCantos(m).arestaX = menorX - mCantos(m).X
mCantos(m).arestaY = mCantos(m).Y - menorY
Case "DB"
menorX = 0
menorY = maxY
For i = 1 To glQtdItensPos
    If ((mItensPos(i).yi + mItensPos(i).Y) >= mCantos(m).Y) And _

```

```

(mItensPos(i).yi <= mCantos(m).Y) And _
(mItensPos(i).xi < mCantos(m).X) Then
  If ((mItensPos(i).xi + mItensPos(i).X) < mCantos(m).X) And ((mItensPos(i).xi +
mItensPos(i).X) > menorX) Then
    menorX = mItensPos(i).xi + mItensPos(i).X
  ElseIf (mItensPos(i).xi < mCantos(m).X) And (mItensPos(i).xi > menorX) Then
    menorX = mItensPos(i).xi
  End If
End If
If ((mItensPos(i).xi + mItensPos(i).X) >= mCantos(m).X) And _
(mItensPos(i).xi <= mCantos(m).X) And _
((mItensPos(i).yi + mItensPos(i).Y) > mCantos(m).Y) Then
  If mItensPos(i).yi < menorY And mItensPos(i).yi > mCantos(m).Y Then
    menorY = mItensPos(i).yi
  ElseIf ((mItensPos(i).yi + mItensPos(i).Y) > mCantos(m).Y) And (mItensPos(i).yi
< menorY) Then
    menorY = mItensPos(i).yi + mItensPos(i).Y
  End If
End If
Next
mCantos(m).arestaX = mCantos(m).X - menorX
mCantos(m).arestaY = menorY - mCantos(m).Y
Case "DA"
  menorX = 0
  menorY = 0
  For i = 1 To glQtdItensPos
    If ((mItensPos(i).yi + mItensPos(i).Y) >= mCantos(m).Y) And _
(mItensPos(i).yi <= mCantos(m).Y) And _
(mItensPos(i).xi < mCantos(m).X) Then
      If ((mItensPos(i).xi + mItensPos(i).X) < mCantos(m).X) And ((mItensPos(i).xi +
mItensPos(i).X) > menorX) Then
        menorX = mItensPos(i).xi + mItensPos(i).X
      ElseIf (mItensPos(i).xi < mCantos(m).X) And (mItensPos(i).xi > menorX) Then
        menorX = mItensPos(i).xi
      End If
    End If
    If ((mItensPos(i).xi + mItensPos(i).X) >= mCantos(m).X) And _
(mItensPos(i).xi <= mCantos(m).X) And _
(mItensPos(i).yi < mCantos(m).Y) Then
      If ((mItensPos(i).yi + mItensPos(i).Y) < mCantos(m).Y) And ((mItensPos(i).yi +
mItensPos(i).Y) > menorY) Then
        menorY = mItensPos(i).yi + mItensPos(i).Y
      ElseIf (mItensPos(i).yi < mCantos(m).Y) And (mItensPos(i).yi > menorY) Then
        menorY = mItensPos(i).yi
      End If
    End If
  Next
  mCantos(m).arestaX = mCantos(m).X - menorX
  mCantos(m).arestaY = mCantos(m).Y - menorY
End Select

```

Next

End Sub

Function sCOPMsA(ByVal k As Long, ByVal a As Double, ByVal b As Double, ByVal c As Double, ByVal d As Double) As Long

Dim m As Long

If k <> 1 Then

For m = 1 To k - 1

If (mCOPMs(m).xp = a) And (mCOPMs(m).yp = b) _
And (mCOPMs(m).xc = c) And (mCOPMs(m).yc = d) Then

sCOPMsA = 0

Exit For

Else

sCOPMsA = -1

End If

Next

Else

sCOPMsA = -1

End If

End Function

Function PossibilidadeEncaixe(ByVal j As Long, ByVal maxX As Double, ByVal maxY As Double) As Long

Dim k As Long, i As Long

Dim cod As Long

Dim pos As Long

Dim ok As Boolean

glQtdCOPMs = glQtdCantos * 2

ReDim mCOPMs(glQtdCOPMs)

k = 1

For i = 1 To glQtdCantos

mCOPMs(k).CodItem = mITEMLocal(j).CodItem

mCOPMs(k).xp = mITEMLocal(j).Car1

mCOPMs(k).yp = mITEMLocal(j).Car2

mCOPMs(k).canto = mCantos(i).Ordem

Select Case mCantos(i).tipo

Case "EB"

mCOPMs(k).xc = mCantos(i).X

mCOPMs(k).yc = mCantos(i).Y

Case "EA"

mCOPMs(k).xc = mCantos(i).X

mCOPMs(k).yc = mCantos(i).Y - mITEMLocal(j).Car2

Case "DA"

```

        mCOPMs(k).xc = mCantos(i).X - mITEMLocal(j).Car1
        mCOPMs(k).yc = mCantos(i).Y - mITEMLocal(j).Car2
    Case "DB"
        mCOPMs(k).xc = mCantos(i).X - mITEMLocal(j).Car1
        mCOPMs(k).yc = mCantos(i).Y
    End Select
    ok = False
    If (mCOPMs(k).xc >= 0) And (mCOPMs(k).yc >= 0) And _
        (mCOPMs(k).xc + mCOPMs(k).xp <= maxX) And _
        (mCOPMs(k).yc + mCOPMs(k).yp <= maxY) Then
        cod = sCOPMsA(k, mCOPMs(k).xp, mCOPMs(k).yp, mCOPMs(k).xc,
mCOPMs(k).yc)
        If cod = -1 Then
            ok = True
            For pos = 1 To glQtdItensPos
                If (mCOPMs(k).xc < mItensPos(pos).X + mItensPos(pos).xi) And _
                    (mCOPMs(k).xc + mCOPMs(k).xp > mItensPos(pos).xi) Then
                    If (mCOPMs(k).yc < mItensPos(pos).Y + mItensPos(pos).yi) And _
                        (mCOPMs(k).yc + mCOPMs(k).yp > mItensPos(pos).yi) Then
                        ok = False
                        Exit For
                    End If
                End If
            Next
        End If
    End If

    If ok = True Then
        k = k + 1
    End If
Next

If mITEMLocal(j).OrientSN = False Then
    For i = 1 To glQtdCantos
        mCOPMs(k).CodItem = mITEMLocal(j).CodItem
        mCOPMs(k).xp = mITEMLocal(j).Car2
        mCOPMs(k).yp = mITEMLocal(j).Car1
        mCOPMs(k).canto = mCantos(i).Ordem
        Select Case mCantos(i).tipo
            Case "EB"
                mCOPMs(k).xc = mCantos(i).X
                mCOPMs(k).yc = mCantos(i).Y
            Case "EA"
                mCOPMs(k).xc = mCantos(i).X
                mCOPMs(k).yc = mCantos(i).Y - mITEMLocal(j).Car1
            Case "DA"
                mCOPMs(k).xc = mCantos(i).X - mITEMLocal(j).Car2
                mCOPMs(k).yc = mCantos(i).Y - mITEMLocal(j).Car1
            Case "DB"
                mCOPMs(k).xc = mCantos(i).X - mITEMLocal(j).Car2

```

```

        mCOPMs(k).yc = mCantos(i).Y
    End Select
    ok = False
    If (mCOPMs(k).xc >= 0) And (mCOPMs(k).yc >= 0) And _
        (mCOPMs(k).xc + mCOPMs(k).xp <= maxX) And _
        (mCOPMs(k).yc + mCOPMs(k).yp <= maxY) Then
        cod = sCOPMsA(k, mCOPMs(k).xp, mCOPMs(k).yp, mCOPMs(k).xc,
mCOPMs(k).yc)
        If cod = -1 Then
            ok = True
            For pos = 1 To glQtdItensPos
                If (mCOPMs(k).xc < mItensPos(pos).X + mItensPos(pos).xi) And _
                    (mCOPMs(k).xc + mCOPMs(k).xp > mItensPos(pos).xi) Then
                    If (mCOPMs(k).yc < mItensPos(pos).Y + mItensPos(pos).yi) And _
                        (mCOPMs(k).yc + mCOPMs(k).yp > mItensPos(pos).yi) Then
                        ok = False
                        Exit For
                    End If
                End If
            End If
        Next
    End If
    End If
    End If

    If ok = True Then
        k = k + 1
    End If
Next
End If
PossibilidadeEncaixe = k - 1

```

End Function

Sub QSmCantosTipoA(ByVal imin As Long, ByVal imax As Long)

```

    Dim t() As rCantos
    Dim i As Long, j As Long, Lmin As Long, Lmax As Long
    Dim Meio As String

```

```

    Lmin = imin
    Lmax = imax
    i = imin
    j = imax
    Meio = mCantos((Lmin + Lmax) / 2).tipo
    ReDim t(1)

```

```

Do
    Do While mCantos(i).tipo < Meio
        i = i + 1
    Loop
    Do While Meio < mCantos(j).tipo

```

```

    j = j - 1
Loop
If i < j Then
    t(1) = mCantos(i)
    mCantos(i) = mCantos(j)
    mCantos(j) = t(1)
End If
If i <= j Then
    i = i + 1
    j = j - 1
End If
Loop Until i > j

If imin < j Then QSmCantosTipoA imin, j
If i < imax Then QSmCantosTipoA i, imax

```

End Sub

Sub QSmCantosXa(ByVal imin As Long, ByVal imax As Long)

```

Dim t() As rCantos
Dim i As Long, j As Long, Lmin As Long, Lmax As Long
Dim Meio As String

```

```

Lmin = imin
Lmax = imax
i = imin
j = imax
Meio = mCantos((Lmin + Lmax) / 2).X
ReDim t(1)

```

```

Do
    Do While mCantos(i).X < Meio
        i = i + 1
    Loop
    Do While Meio < mCantos(j).X
        j = j - 1
    Loop
    If i < j Then
        t(1) = mCantos(i)
        mCantos(i) = mCantos(j)
        mCantos(j) = t(1)
    End If
    If i <= j Then
        i = i + 1
        j = j - 1
    End If
Loop Until i > j

```

If imin < j Then QSmCantosXa imin, j

```
If i < imax Then QSmCantosXa i, imax
```

```
End Sub
```

```
Sub QSmCantosYa(ByVal imin As Long, ByVal imax As Long)
```

```
Dim t() As rCantos
```

```
Dim i As Long, j As Long, Lmin As Long, Lmax As Long
```

```
Dim Meio As String
```

```
Lmin = imin
```

```
Lmax = imax
```

```
i = imin
```

```
j = imax
```

```
Meio = mCantos((Lmin + Lmax) / 2).Y
```

```
ReDim t(1)
```

```
Do
```

```
Do While mCantos(i).Y < Meio
```

```
    i = i + 1
```

```
Loop
```

```
Do While Meio < mCantos(j).Y
```

```
    j = j - 1
```

```
Loop
```

```
If i < j Then
```

```
    t(1) = mCantos(i)
```

```
    mCantos(i) = mCantos(j)
```

```
    mCantos(j) = t(1)
```

```
End If
```

```
If i <= j Then
```

```
    i = i + 1
```

```
    j = j - 1
```

```
End If
```

```
Loop Until i > j
```

```
If imin < j Then QSmCantosYa imin, j
```

```
If i < imax Then QSmCantosYa i, imax
```

```
End Sub
```

```
Sub ObtFaixaCodObjmPadCorteDesempTemp(ByVal cod As Long, iFx As Long, fFx As Long)
```

```
Dim i As Long
```

```
iQQ = BSCodObjmPadCorteDesempTemp(cod)
```

```
If iQQ = -1 Then
```

```
    iFx = -1: fFx = -1
```

```
Exit Sub
```

```

End If

'descobre o início da faixa
For i = iQQ To 0 Step -1
  If mPadCorteDesempTemp(i).CodObj <> mPadCorteDesempTemp(iQQ).CodObj Then
    Exit For
  End If
Next
iFx = i + 1

'descobre o fim da faixa
For i = iQQ To UBound(mPadCorteDesempTemp)
  If mPadCorteDesempTemp(i).CodObj <> mPadCorteDesempTemp(iQQ).CodObj Then
    Exit For
  End If
Next
fFx = i - 1

End Sub
Sub ObtFaixaKeyStrmOBJETODad(ByVal cod As String, iFx As Long, fFx As Long)

  Dim i As Long

  iQQ = BSKKeyStrmOBJETODad(cod)

  If iQQ = -1 Then
    iFx = -1: fFx = -1
    Exit Sub
  End If

  'descobre o início da faixa
  For i = iQQ To 0 Step -1
    If mOBJETODad(i).KeyStr <> mOBJETODad(iQQ).KeyStr Then
      Exit For
    End If
  Next
  iFx = i + 1

  'descobre o fim da faixa
  For i = iQQ To UBound(mOBJETODad)
    If mOBJETODad(i).KeyStr <> mOBJETODad(iQQ).KeyStr Then
      Exit For
    End If
  Next
  fFx = i - 1

End Sub

Sub ObtFaixaCodItemmITEMLocal(ByVal cod As Long, iFx As Long, fFx As Long)

```

```
Dim i As Long
```

```
iQQ = BScodItemmITEMLocal(cod)
```

```
If iQQ = -1 Then
```

```
    iFx = -1: fFx = -1
```

```
    Exit Sub
```

```
End If
```

```
'descobre o início da faixa
```

```
For i = iQQ To 0 Step -1
```

```
    If mITEMLocal(i).CodItem <> mITEMLocal(iQQ).CodItem Then
```

```
        Exit For
```

```
    End If
```

```
Next
```

```
iFx = i + 1
```

```
'descobre o fim da faixa
```

```
For i = iQQ To UBound(mITEMLocal)
```

```
    If mITEMLocal(i).CodItem <> mITEMLocal(iQQ).CodItem Then
```

```
        Exit For
```

```
    End If
```

```
Next
```

```
fFx = i - 1
```

```
End Sub
```

```
Sub ObtFaixaCritAgrupmITEMDad(ByVal cod As String, iFx As Long, fFx As Long)
```

```
Dim i As Long
```

```
iQQ = BScritAgrupmITEMDad(cod)
```

```
If iQQ = -1 Then
```

```
    iFx = -1: fFx = -1
```

```
    Exit Sub
```

```
End If
```

```
'descobre o início da faixa
```

```
For i = iQQ To 0 Step -1
```

```
    If mITEMDad(i).CritAgrup <> mITEMDad(iQQ).CritAgrup Then
```

```
        Exit For
```

```
    End If
```

```
Next
```

```
iFx = i + 1
```

```
'descobre o fim da faixa
```

```
For i = iQQ To UBound(mITEMDad)
```

```
    If mITEMDad(i).CritAgrup <> mITEMDad(iQQ).CritAgrup Then
```

```

        Exit For
    End If
Next
fFx = i - 1

End Sub

Sub ObtFaixaCritAgrupmOBJETODad(ByVal cod As String, iFx As Long, fFx As Long)

    Dim i As Long

    iQQ = BSCritAgrupmOBJETODad(cod)

    If iQQ = -1 Then
        iFx = -1: fFx = -1
        Exit Sub
    End If

    'descobre o início da faixa
    For i = iQQ To 0 Step -1
        If mOBJETODad(i).CritAgrup <> mOBJETODad(iQQ).CritAgrup Then
            Exit For
        End If
    Next
    iFx = i + 1

    'descobre o fim da faixa
    For i = iQQ To UBound(mOBJETODad)
        If mOBJETODad(i).CritAgrup <> mOBJETODad(iQQ).CritAgrup Then
            Exit For
        End If
    Next
    fFx = i - 1

End Sub

Sub ObtFaixaMelhorObjSNmPadCorteDesempTemp(iFx As Long, fFx As Long)

    Dim i As Long

    'Caso específico em que iQQ será sempre 1
    iFx = 1

    'descobre o fim da faixa
    For i = iFx To glQtdPadCorteDesempTemp
        If
            mPadCorteDesempTemp(i).MelhorObjSN
            mPadCorteDesempTemp(iFx).MelhorObjSN Then
            Exit For
        End If
    Next

```

◊

```

Next
fFx = i - 1

```

```

End Sub

```

```

Function BSKeyStrmOBJETODad(ByVal cod As String) As Long
'binary search em mOBJETODad

```

```

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

```

```

BSKeyStrmOBJETODad = -1
Hi = UBound(mOBJETODad)
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0
Do While Tent < 30
  If cod = mOBJETODad(Meio).KeyStr Then
    BSKeyStrmOBJETODad = Meio
    Exit Do
  Else
    If cod < mOBJETODad(Meio).KeyStr Then
      Hi = Meio - 1
    Else
      Lo = Meio + 1
    End If
  End If
  Meio = (Hi + Lo) \ 2
  Tent = Tent + 1
Loop
End Function

```

```

Function BSCodObjmOBJETODad(ByVal cod As Long) As Long
'binary search em mOBJETODad

```

```

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

```

```

BSCodObjmOBJETODad = -1
Hi = UBound(mOBJETODad)
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0
Do While Tent < 30
  If cod = mOBJETODad(Meio).CodObj Then
    BSCodObjmOBJETODad = Meio
    Exit Do
  Else
    If cod < mOBJETODad(Meio).CodObj Then
      Hi = Meio - 1
    Else

```

```

        Lo = Meio + 1
    End If
End If
Meio = (Hi + Lo) \ 2
Tent = Tent + 1
Loop
End Function

```

Function BSCodObjmPadCorteDesempTemp(ByVal cod As Long) As Long
'binary search em mPadCorteDesempTemp

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

```

BSCodObjmPadCorteDesempTemp = -1
Hi = glQtdPadCorteDesempTemp
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0
Do While Tent < 30
    If cod = mPadCorteDesempTemp(Meio).CodObj Then
        BSCodObjmPadCorteDesempTemp = Meio
        Exit Do
    Else
        If cod < mPadCorteDesempTemp(Meio).CodObj Then
            Hi = Meio - 1
        Else
            Lo = Meio + 1
        End If
    End If
    Meio = (Hi + Lo) \ 2
    Tent = Tent + 1
Loop
End Function

```

Function BSmORDEMDadCodOrdem(ByVal cod As Long) As Long
'binary search em mORDEMDad

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

```

BSmORDEMDadCodOrdem = -1
Hi = UBound(mORDEMDad)
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0
Do While Tent < 30
    If cod = mORDEMDad(Meio).CodOrdem Then
        BSmORDEMDadCodOrdem = Meio
        Exit Do
    Else

```

```

    If cod < mORDEMDad(Meio).CodOrdem Then
        Hi = Meio - 1
    Else
        Lo = Meio + 1
    End If
End If
Meio = (Hi + Lo) \ 2
Tent = Tent + 1
Loop
End Function

```

```

Function BSmITEMDadCodItem(ByVal cod As Long) As Long
'binary search em mITEMDad

```

```

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

```

```

BSmITEMDadCodItem = -1
Hi = UBound(mITEMDad)
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0
Do While Tent < 30
    If cod = mITEMDad(Meio).CodItem Then
        BSmITEMDadCodItem = Meio
        Exit Do
    Else
        If cod < mITEMDad(Meio).CodItem Then
            Hi = Meio - 1
        Else
            Lo = Meio + 1
        End If
    End If
    Meio = (Hi + Lo) \ 2
    Tent = Tent + 1
Loop
End Function

```

```

Function BSCritAgrupmOBJETODad(cod As String) As Long
'binary search em mOBJETODad

```

```

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

```

```

BSCritAgrupmOBJETODad = -1
Hi = UBound(mOBJETODad)
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0
Do While Tent < 30
    If cod = mOBJETODad(Meio).CritAgrup Then

```

```

    BScritAgrupmOBJETODad = Meio
    Exit Do
Else
    If cod < mOBJETODad(Meio).CritAgrup Then
        Hi = Meio - 1
    Else
        Lo = Meio + 1
    End If
End If
Meio = (Hi + Lo) \ 2
Tent = Tent + 1
Loop
End Function

Function BScritAgrupmITEMDad(cod As String) As Long
'binary search em mITEMDad

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

BScritAgrupmITEMDad = -1
Hi = UBound(mITEMDad)
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0
Do While Tent < 30
    If cod = mITEMDad(Meio).CritAgrup Then
        BScritAgrupmITEMDad = Meio
        Exit Do
    Else
        If cod < mITEMDad(Meio).CritAgrup Then
            Hi = Meio - 1
        Else
            Lo = Meio + 1
        End If
    End If
    Meio = (Hi + Lo) \ 2
    Tent = Tent + 1
Loop
End Function

Function BScodItemmITEMLocal(cod As Long) As Long
'binary search em mITEMLocal

Dim Hi As Long, Lo As Long, Meio As Long, Tent As Long

BScodItemmITEMLocal = -1
Hi = UBound(mITEMLocal)
Lo = 1
Meio = (Hi + Lo) \ 2
Tent = 0

```

```

Do While Tent < 30
  If cod = mITEMLocal(Meio).CodItem Then
    BScodItemmITEMLocal = Meio
    Exit Do
  Else
    If cod < mITEMLocal(Meio).CodItem Then
      Hi = Meio - 1
    Else
      Lo = Meio + 1
    End If
  End If
  Meio = (Hi + Lo) \ 2
  Tent = Tent + 1
Loop
End Function

```

```

Sub QSmORDEMDadCodOrdem(imin As Long, imax As Long)
'ordena a matriz mORDEMDad em ordem crescente pela CodOrdem
Dim tReg As rORDEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Long, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mORDEMDad((Lmin + Lmax) \ 2).CodOrdem
Do
  Do While mORDEMDad(k).CodOrdem < Meio
    k = k + 1
  Loop
  Do While Meio < mORDEMDad(j).CodOrdem
    j = j - 1
  Loop
  If k < j Then
    tReg = mORDEMDad(k)
    mORDEMDad(k) = mORDEMDad(j)
    mORDEMDad(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmORDEMDadCodOrdem imin, j
If k < imax Then QSmORDEMDadCodOrdem k, imax
End Sub

```

```

Sub QSmITEMDadCodItem(imin As Long, imax As Long)
'ordena a matriz mITEMDad em ordem crescente pelo CodItem
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Long, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMDad((Lmin + Lmax) \ 2).CodItem
Do
  Do While mITEMDad(k).CodItem < Meio
    k = k + 1
  Loop
  Do While Meio < mITEMDad(j).CodItem
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMDad(k)
    mITEMDad(k) = mITEMDad(j)
    mITEMDad(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmITEMDadCodItem imin, j
If k < imax Then QSmITEMDadCodItem k, imax
End Sub

```

```

Sub QSmITEMDadAtivo(imin As Long, imax As Long)
'ordena a matriz mITEMDad em ordem crescente pelo AtivoSN
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Boolean, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMDad((Lmin + Lmax) \ 2).AtivoSN
Do
  Do While mITEMDad(k).AtivoSN < Meio
    k = k + 1
  Loop
  Do While Meio < mITEMDad(j).AtivoSN
    j = j - 1
  Loop

```

```

Loop
If k < j Then
  tReg = mITEMDad(k)
  mITEMDad(k) = mITEMDad(j)
  mITEMDad(j) = tReg
End If
If k <= j Then
  k = k + 1
  j = j - 1
End If
Loop Until k > j
If imin < j Then QSmITEMDadAtivo imin, j
If k < imax Then QSmITEMDadAtivo k, imax
End Sub

Sub QSmITEMDadCritAgrup(imin As Long, imax As Long)
'ordena a matriz mITEMDad em ordem crescente pelo CritAgrup
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As String, Lmin As Long, Lmax As Long

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMDad((Lmin + Lmax) \ 2).CritAgrup
Do
  Do While mITEMDad(k).CritAgrup < Meio
    k = k + 1
  Loop
  Do While Meio < mITEMDad(j).CritAgrup
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMDad(k)
    mITEMDad(k) = mITEMDad(j)
    mITEMDad(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmITEMDadCritAgrup imin, j
If k < imax Then QSmITEMDadCritAgrup k, imax
End Sub

Sub QSmITEMDadCritData(imin As Long, imax As Long)
'ordena a matriz mITEMDad em ordem crescente pelo CritData
Dim tReg As rITEMDad

```

```
Dim j As Long, k As Long, L As Long
Dim Meio As Double, Lmin As Long, Lmax As Long
```

```
Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMDad((Lmin + Lmax) \ 2).CritData
Do
  Do While mITEMDad(k).CritData < Meio
    k = k + 1
  Loop
  Do While Meio < mITEMDad(j).CritData
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMDad(k)
    mITEMDad(k) = mITEMDad(j)
    mITEMDad(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmITEMDadCritData imin, j
If k < imax Then QSmITEMDadCritData k, imax
End Sub
```

```
Sub QSmPadCorteDesempTempCodObj(imin As Long, imax As Long)
'ordena a matriz mPadCorteDesempTemp em ordem crescente pelo CodObj
Dim tReg As rPadCorteDesemp
Dim j As Long, k As Long, L As Long
Dim Meio As Long, Lmin As Long, Lmax As Long
```

```
Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mPadCorteDesempTemp((Lmin + Lmax) \ 2).CodObj
Do
  Do While mPadCorteDesempTemp(k).CodObj < Meio
    k = k + 1
  Loop
  Do While Meio < mPadCorteDesempTemp(j).CodObj
    j = j - 1
  Loop
  If k < j Then
    tReg = mPadCorteDesempTemp(k)
```

```

    mPadCorteDesempTemp(k) = mPadCorteDesempTemp(j)
    mPadCorteDesempTemp(j) = tReg
End If
If k <= j Then
    k = k + 1
    j = j - 1
End If
Loop Until k > j
If imin < j Then QSmPadCorteDesempTempCodObj imin, j
If k < imax Then QSmPadCorteDesempTempCodObj k, imax
End Sub

Sub QSmPadCorteDesempTempMelhorObjSN(imin As Long, imax As Long)
'ordena a matriz mPadCorteDesempTemp em ordem crescente pelo MelhorObjSN
Dim tReg As rPadCorteDesemp
Dim j As Long, k As Long, L As Long
Dim Meio As Long, Lmin As Long, Lmax As Long

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mPadCorteDesempTemp((Lmin + Lmax) \ 2).MelhorObjSN
Do
    Do While mPadCorteDesempTemp(k).MelhorObjSN < Meio
        k = k + 1
    Loop
    Do While Meio < mPadCorteDesempTemp(j).MelhorObjSN
        j = j - 1
    Loop
    If k < j Then
        tReg = mPadCorteDesempTemp(k)
        mPadCorteDesempTemp(k) = mPadCorteDesempTemp(j)
        mPadCorteDesempTemp(j) = tReg
    End If
    If k <= j Then
        k = k + 1
        j = j - 1
    End If
Loop Until k > j
If imin < j Then QSmPadCorteDesempTempMelhorObjSN imin, j
If k < imax Then QSmPadCorteDesempTempMelhorObjSN k, imax
End Sub

Sub QSmITEMLocalCritData(imin As Long, imax As Long)
'ordena a matriz mITEMLocal em ordem crescente pela FDin
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Double, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMLocal((Lmin + Lmax) \ 2).CritData
Do
  Do While mITEMLocal(k).CritData < Meio
    k = k + 1
  Loop
  Do While Meio < mITEMLocal(j).CritData
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMLocal(k)
    mITEMLocal(k) = mITEMLocal(j)
    mITEMLocal(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmITEMLocalCritData imin, j
If k < imax Then QSmITEMLocalCritData k, imax
End Sub

```

```

Sub QSmITEMLocalCar1(imin As Long, imax As Long)
'ordena a matriz mITEMLocal em ordem crescente pela Car1
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Double, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMLocal((Lmin + Lmax) \ 2).Car1
Do
  Do While mITEMLocal(k).Car1 > Meio
    k = k + 1
  Loop
  Do While Meio > mITEMLocal(j).Car1
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMLocal(k)
    mITEMLocal(k) = mITEMLocal(j)
    mITEMLocal(j) = tReg
  End If

```

```

If k <= j Then
  k = k + 1
  j = j - 1
End If
Loop Until k > j
If imin < j Then QSmITEMLocalCar1 imin, j
If k < imax Then QSmITEMLocalCar1 k, imax
End Sub

```

```

Sub QSmITEMLocalAtivo(imin As Long, imax As Long)
'ordena a matriz mITEMLocal em ordem crescente pelo AtivoSN
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Boolean, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMLocal((Lmin + Lmax) \ 2).AtivoSN
Do
  Do While mITEMLocal(k).AtivoSN < Meio
    k = k + 1
  Loop
  Do While Meio < mITEMLocal(j).AtivoSN
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMLocal(k)
    mITEMLocal(k) = mITEMLocal(j)
    mITEMLocal(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmITEMLocalAtivo imin, j
If k < imax Then QSmITEMLocalAtivo k, imax
End Sub

```

```

Sub QSmITEMLocalCodItem(imin As Long, imax As Long)
'ordena a matriz mITEMLocal em ordem crescente pelo CodItem
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Long, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax

```

```

k = imin
j = imax
Meio = mITEMLocal((Lmin + Lmax) \ 2).CodItem
Do
  Do While mITEMLocal(k).CodItem < Meio
    k = k + 1
  Loop
  Do While Meio < mITEMLocal(j).CodItem
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMLocal(k)
    mITEMLocal(k) = mITEMLocal(j)
    mITEMLocal(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmITEMLocalCodItem imin, j
If k < imax Then QSmITEMLocalCodItem k, imax
End Sub

```

```

Sub QSmITEMLocalCritPond(imin As Long, imax As Long)
'ordena a matriz mITEMLocal em ordem crescente pelo CritPond
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Double, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mITEMLocal((Lmin + Lmax) \ 2).CritPond
Do
  Do While mITEMLocal(k).CritPond > Meio
    k = k + 1
  Loop
  Do While Meio > mITEMLocal(j).CritPond
    j = j - 1
  Loop
  If k < j Then
    tReg = mITEMLocal(k)
    mITEMLocal(k) = mITEMLocal(j)
    mITEMLocal(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j

```

```

End If
Loop Until k > j
If imin < j Then QSmITEMLocalCritPond imin, j
If k < imax Then QSmITEMLocalCritPond k, imax
End Sub

Sub QSmITEMDadKeyStr(imin As Long, imax As Long)
'Ordena mITEMDad por uma chave composta de:
'CritAgrup, AtivoSN, CritOrd e OP, nesta ordem

Dim Temp As rITEMDad, Meio As String
Dim i As Long, j As Long, Lmin As Long, Lmax As Long

Lmin = imin
Lmax = imax
i = imin
j = imax
Meio = mITEMDad((Lmin + Lmax) / 2).KeyStr

Do
Do While mITEMDad(i).KeyStr < Meio
i = i + 1
Loop
Do While Meio < mITEMDad(j).KeyStr
j = j - 1
Loop
If i < j Then
Temp = mITEMDad(i)
mITEMDad(i) = mITEMDad(j)
mITEMDad(j) = Temp
End If
If i <= j Then
i = i + 1
j = j - 1
End If
Loop Until i > j

If imin < j Then QSmITEMDadKeyStr imin, j
If i < imax Then QSmITEMDadKeyStr i, imax

End Sub

```

```

Function ObtChCompITEM(i As Long, heur As Long) As String
'retorna uma chave composta para ser usada na rotina de QS acima

```

```

Dim CritAgrup As String
Dim AtivoSN As String
Dim OP As String
Dim CritOrd As String
Dim aux As String

```

```

'ajusta o tamanho do campo CritAgrup para 12 caracteres
CritAgrup = Right$("000000000000" & Trim$(CStr(mITEMDad(i).CritAgrup)), 12)
'ajusta o tamanho do campo AtivoSN para 1 caracter, com True=0 e False=1
If mITEMDad(i).AtivoSN = True Then
    AtivoSN = "0"
Else
    AtivoSN = "1"
End If
'ajusta o tamanho do campo CritOrd para 6 caracteres
Select Case mHEURDad(heur).SeqHeur
    Case "CritData"
        CritOrd = Right$("000000" & Trim$(100000 - Round(mITEMDad(i).CritData * 1000,
0)), 6)
    Case "CritOcup"
        CritOrd = Right$("000000" & Trim$(100000 - Round(mITEMDad(i).CritOcup * 1000,
0)), 6)
    Case "CritPonderado"
        CritOrd = Right$("000000" & Trim$(100000 - Round(mITEMDad(i).CritPond * 1000,
0)), 6)
End Select
'ajusta o tamanho do campo OP para 10 caracteres
OP = Right$("0000000000" & Trim$(CStr(mITEMDad(i).CodOP)), 10)

'monta e retorna a chave composta
ObtChCompITEM = CritAgrup & AtivoSN & CritOrd & OP

End Function

```

```

Sub QSChvCompmCantosXY(imin As Long, imax As Long)
'esta rotina ordena a matriz de strings mCantos() em ordem crescente

```

```

    Dim Temp As rCantos, Meio As String
    Dim i As Long, j As Long, Lmin As Long, Lmax As Long

```

```

    Lmin = imin
    Lmax = imax
    i = imin
    j = imax
    Meio = ObtChvCompXY((Lmin + Lmax) / 2)

```

```

Do
    Do While StrComp(ObtChvCompXY(i), Meio) = -1
        i = i + 1
    Loop
    Do While StrComp(Meio, ObtChvCompXY(j)) = -1
        j = j - 1
    Loop
    If i < j Then
        Temp = mCantos(i)

```

```

        mCantos(i) = mCantos(j)
        mCantos(j) = Temp
    End If
    If i <= j Then
        i = i + 1
        j = j - 1
    End If
Loop Until i > j

If imin < j Then QSChvCompCantosXY imin, j
If i < imax Then QSChvCompCantosXY i, imax

```

End Sub

Function ObtChvCompXY(i As Long) As String

'retorna uma chave composta para ser usada na rotina de quicksort acima

```
Dim X As String, Y As String
```

'ajusta o tamanho do campo X para 5 caracteres

```
X = Right$("000000" & Trim$(CStr(Round(100 * mCantos(i).X, 0))), 6)
```

'ajusta o tamanho do campo duracao para 6 caracteres

```
Y = Right$("0000000" & Trim$(CStr(Round(100 * mCantos(i).Y, 0))), 6)
```

'monta e retorna a chave composta

```
ObtemChaveComposta = X & Y
```

End Function

Sub QSmOBJETODadKeyStr(imin As Long, imax As Long)

'Ordena mOBJETODad por uma chave composta de:

'CritAgrup, AtivoSN, PriorObjAgr

```
Dim Temp As rOBJETODad, Meio As String
```

```
Dim i As Long, j As Long, Lmin As Long, Lmax As Long
```

```
Lmin = imin
```

```
Lmax = imax
```

```
i = imin
```

```
j = imax
```

```
Meio = mOBJETODad((Lmin + Lmax) / 2).KeyStr
```

Do

```
Do While mOBJETODad(i).KeyStr < Meio
```

```
    i = i + 1
```

```
Loop
```

```
Do While Meio < mOBJETODad(j).KeyStr
```

```
    j = j - 1
```

```
Loop
```

```
If i < j Then
```

```

    Temp = mOBJETODad(i)
    mOBJETODad(i) = mOBJETODad(j)
    mOBJETODad(j) = Temp
End If
If i <= j Then
    i = i + 1
    j = j - 1
End If
Loop Until i > j

If imin < j Then QSmOBJETODadKeyStr imin, j
If i < imax Then QSmOBJETODadKeyStr i, imax

```

End Sub

Function ObtChCompOBJETO(i As Long) As String
'returna uma chave composta para ser usada na rotina de QS acima

```

Dim CritAgrup As String
Dim AtivoSN As String
Dim PriorObjAgr As String

'ajusta o tamanho do campo CritAgrup para 12 caracteres
CritAgrup = Right$("000000000000" & Trim$(CStr(mOBJETODad(i).CritAgrup)), 12)
'ajusta o tamanho do campo AtivoSN para 1 caracter, com True=0 e False=1
If mOBJETODad(i).AtivoSN = True Then
    AtivoSN = "0"
Else
    AtivoSN = "1"
End If
'ajusta o tamanho do campo OP para 2 caracteres
PriorObjAgr = Right$("00" & Trim$(CStr(mOBJETODad(i).PriorObjAgr)), 2)

'monta e retorna a chave composta
ObtChCompOBJETO = CritAgrup & AtivoSN & PriorObjAgr

```

End Function

```

Sub QSmPadCorteDesempCritPond(imin As Long, imax As Long)
'ordena a matriz mPadCorteDesemp em ordem decrescente de CritPond
Dim tReg As rPadCorteDesemp
Dim j As Long, k As Long, L As Long
Dim Meio As Double, Lmin As Long, Lmax As Long

```

```

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mPadCorteDesemp((Lmin + Lmax) \ 2).CritPond
Do

```

```

Do While mPadCorteDesemp(k).CritPond > Meio
  k = k + 1
Loop
Do While Meio > mPadCorteDesemp(j).CritPond
  j = j - 1
Loop
If k < j Then
  tReg = mPadCorteDesemp(k)
  mPadCorteDesemp(k) = mPadCorteDesemp(j)
  mPadCorteDesemp(j) = tReg
End If
If k <= j Then
  k = k + 1
  j = j - 1
End If
Loop Until k > j
If imin < j Then QSmPadCorteDesempCritPond imin, j
If k < imax Then QSmPadCorteDesempCritPond k, imax
End Sub

Sub QSmSubObjetoAtivoSN(imin As Long, imax As Long)
'ordena a matriz mSubObjeto em ordem decrescente de AtivoSN
Dim tReg As rSubObjeto
Dim j As Long, k As Long, L As Long
Dim Meio As Double, Lmin As Long, Lmax As Long

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mSubObjeto((Lmin + Lmax) \ 2).AtivoSN
Do
  Do While mSubObjeto(k).AtivoSN < Meio
    k = k + 1
  Loop
  Do While Meio < mSubObjeto(j).AtivoSN
    j = j - 1
  Loop
  If k < j Then
    tReg = mSubObjeto(k)
    mSubObjeto(k) = mSubObjeto(j)
    mSubObjeto(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmSubObjetoAtivoSN imin, j
If k < imax Then QSmSubObjetoAtivoSN k, imax

```

End Sub

```

Sub QSmITEMPadCorteFDin(imin As Long, imax As Long, ByVal cont As Long)
'ordena a matriz mPadCorteDesempTemp(j).mITEMPadCorte em ordem decrescente de FDin
Dim tReg As rITEMDad
Dim j As Long, k As Long, L As Long
Dim Meio As Long, Lmin As Long, Lmax As Long

Lmin = imin
Lmax = imax
k = imin
j = imax
Meio = mPadCorteDesempTemp(cont).mITEMPadCorte((Lmin + Lmax) \ 2).FDin
Do
  Do While mPadCorteDesempTemp(cont).mITEMPadCorte(k).FDin < Meio
    k = k + 1
  Loop
  Do While Meio < mPadCorteDesempTemp(cont).mITEMPadCorte(j).FDin
    j = j - 1
  Loop
  If k < j Then
    tReg = mPadCorteDesempTemp(cont).mITEMPadCorte(k)
    mPadCorteDesempTemp(cont).mITEMPadCorte(k) =
mPadCorteDesempTemp(cont).mITEMPadCorte(j)
    mPadCorteDesempTemp(cont).mITEMPadCorte(j) = tReg
  End If
  If k <= j Then
    k = k + 1
    j = j - 1
  End If
Loop Until k > j
If imin < j Then QSmITEMPadCorteFDin imin, j, cont
If k < imax Then QSmITEMPadCorteFDin k, imax, cont
End Sub

```