

UNIVERSIDADE TÉCNICA DE LISBOA

INSTITUTO SUPERIOR DE ECONOMIA E GESTÃO

Mestrado em Decisão Económica e Empresarial

Problema de corte bidimensional
Aplicação a um caso real

Joel Alexandre Roda Gomes

Júri

Orientação: Maria Cândida Vergueiro Monteiro Cidade Mourão

Presidente: Margarida Maria Gonçalves Vaz Pato

Vogais: Leonor Almeida Leite Santiago Pinto

Novembro de 2011

Joel Gomes

joelargomes@gmail.com

Agradecimentos

Na realização deste trabalho contei com a ajuda de várias pessoas que me encaminharam e apoiaram na escrita, suporte e revisão.

Deixo um grande apreço e agradecimento à minha orientadora, Professora Cândida Mourão, pelo suporte directo e activo que me deu, concedendo-me também a oportunidade de apresentar parte deste trabalho em congressos, o que para mim foi uma grande mais valia. Destaco também o apoio concedido pelo Centro de Investigação Operacional (CIO) sediado na Faculdade de Ciências da Universidade de Lisboa.

Agradeço ao Professor José Oliveira da Faculdade de Engenharia da Universidade do Porto e a Frank Ortmann da Stellenbosch University pela simpatia e apoio na conceção de material de trabalho.

A todas as professoras e professores do Instituto Superior de Economia e Gestão que contribuíram para a minha formação.

À minha irmã, amigos e colegas um obrigado pelos momentos de distração e divertimento que me foram essenciais para o equilíbrio e bem estar.

Ao meu pai e mãe deixo o agradecimento mais especial e dedico-lhes este trabalho. Sem eles não chegaria onde estou e muito menos teria a oportunidade de o realizar.

Joel Gomes

Resumo

O problema de corte guilhotina e empacotamento bidimensional rectangular consiste em alocar múltiplas peças pequenas – itens – numa ou mais placas de tamanho maior – objectos – num padrão que minimize o desperdício de matéria-prima.

A motivação para a realização deste projecto é resolver um problema real de uma empresa portuguesa tentando, ao mesmo tempo, propor algo novo.

Para isso, desenvolvem-se e apresentam-se duas novas heurísticas, a *Guillotinable Bottom-Left First Fit Decreasing Height* (BLFFDH_G) e a *Bottom-Left First Fit Decreasing Height* (BLFFDH), baseadas na *First Fit Decreasing Height* (FFDH) e *Bottom-up left-justified* (BL), em que, após um nível ter sido preenchido com a abordagem da FFDH, e antes de se abrir um novo nível para o próximo rectângulo, o nível actual é exhaustivamente examinado, usando a heurística BL, de modo a tentar alocar itens no espaço que sobra entre dois níveis consecutivos. A diferença entre as novas heurísticas reside no facto de uma impor o corte guilhotina. Em ambas nenhuma das peças pode ser rodada ou sobreposta. Só depois de explorado o nível actual é aberto um novo.

Os resultados são comparados com heurísticas da literatura, num conjunto de instâncias reais, em corte de roupeiros, e da literatura. As heurísticas propostas são comparadas entre si em termos de tempos de execução e é determinada a complexidade empírica da programação.

Os resultados obtidos indicam que os algoritmos BLFFDH_G e BLFFDH proporcionam quase sempre melhores soluções que os algoritmos que lhe deram origem e são bastante competitivos em relação às outras heurísticas usadas nos testes. Em termos de tempo de execução, a BLFFDH_G revelou-se mais rápida que a BLFFDH, e a complexidade empírica da programação é, para ambas, $O(n^3)$.

Palavras-Chave: Corte guilhotina, métodos heurísticos, First Fit Decreasing Height, Bottom-up left-justified, Guillotinable Bottom-Left First Fit Decreasing Height, Bottom-Left First Fit Decreasing Height, bin packing, strip packing

Abstract

The guillotine cutting problem with two-dimensional rectangular packaging consists of allocating small items in one or more bins – objects – with a pattern that minimize the waste of raw materials.

The motivation for this project is to solve a real problem of a Portuguese company and, at the same time, try to propose something new.

To this aim, two new heuristics are it developed and presented, the Guillotunable Bottom-Left First Fit Decreasing Height (BLFFDH_G) and Bottom-Left First Fit Decreasing Height (BLFFDH), based on First Fit Decreasing Height (FFDH) and Bottom-up left-justified (BL), in which, after a level has been filled with the approach of FFDH, and before opening a new level to the next item, the current level is thoroughly examined, using the BL heuristic, so trying to allocate items in the space left between two consecutive levels. The difference between the new heuristics is that one ensures a pattern that is guillotine cuttable, but in none of them the items can be rotated or overlapped. Only after exploring the current level a new one is open.

The results are compared, in terms of solution, with heuristics presented in the literature, using a set of real based instances from a wardrobe cutting and literature instances. The proposed heuristics are compared in terms of execution times and its empirically complexity of programming is estimated. The results indicate that the algorithms BLFFDH_G and BLFFDH usually provide better solutions than the algorithms FFDH and BL and are quite competitive when compared with other heuristics used in the tests. In terms of execution time, the BLFFDH_G proved to be faster than BLFFDH and empirically they both have a complexity of $O(n^3)$.

Keywords: Guillotine cut, heuristics, First Fit Decreasing Height, Bottom-up left-justified, Guillotunable Bottom-Left First Fit Decreasing Height, Bottom-Left First Fit Decreasing Height, bin packing, strip packing

CONTEÚDO

Lista de Figuras.....	vi
Lista de Tabelas	vii
Lista de Acrónimos	viii
1. Introdução.....	1
2. Problema de corte guilhotina bidimensional	3
2.1. Enquadramento.....	3
2.2. Tipologia de Wäscher	5
2.3. Terminologia	6
2.4. Conceitos.....	6
2.4.1. Corte guilhotina	6
2.4.2. Nível.....	7
2.4.3. Ponto de referência	7
2.4.4. <i>Gap</i>	8
2.4.5. Sub-objecto.....	9
3. Heurísticas orientadas por níveis.....	10
3.1. Introdução.....	10
3.2. <i>Next-Fit Decreasing-Height</i> (NFDH).....	11
3.3. <i>First-Fit Decreasing-Height</i> (FFDH)	11
3.4. <i>Modified First Fit Decreasing Height</i> (MFFDH).....	12
3.5. <i>Adaptive First Fit Decreasing Height</i> (AFFDH)	13
4. Heurísticas não orientadas por níveis	14
4.1. Introdução.....	14
4.2. <i>Improved Lowest Gap Fill</i> (LGF _{IOF})	15
4.3. <i>Bottom-up left-justified</i> (BL).....	16

5.	Heurísticas Propostas	17
5.1.	Introdução.....	17
5.2.	<i>Bottom-Left First Fit Decreasing Height</i> (BLFFDH).....	18
5.3.	<i>Guillotinable Bottom-Left First Fit Decreasing Height</i> (BLFFDH _G)	20
6.	Resultados Computacionais	22
6.1.	Introdução.....	22
6.2.	Instâncias.....	23
6.2.1.	Instâncias reais.....	23
6.2.2.	Instâncias da literatura	24
6.3.	Resultados	26
6.3.1.	Instâncias reais – roupeiros	26
6.3.2.	Instâncias da literatura	28
6.4.	Estudo Empírico	31
6.4.1.	Tempos de execução	31
6.4.2.	Complexidade computacional empírica.....	33
7.	Conclusões e Trabalhos Futuros.....	34
7.1.	Conclusões	34
7.2.	Propostas para Trabalhos Futuros.....	35
8.	Bibliografia.....	36
9.	ANEXOS	39
	Anexo 1 - Fluxogramas da Heurística BLFFDH	40
	Anexo 2 - Fluxogramas da Heurística BLFFDH _G	44
	Anexo 3 - Comparação das instâncias da literatura com metodologia das instâncias práticas.....	48

LISTA DE FIGURAS

Figura 1- Padrão: (a) de não guilhotina; (b) de guilhotina.....	7
Figura 2- Definição de dois níveis e da fronteira que os separa.....	7
Figura 3- Pontos de referência.....	8
Figura 4- <i>Gap</i> e PR em: (a) algoritmos orientados por níveis; (b) algoritmos não orientados por níveis.....	8
Figura 5- Espaço destinado a sub-objecto(s).....	9
Figura 6- Sub-objectos: (a) com forma rectangular; (b) com forma irregular.	9
Figura 7- Criação de um sub-objecto quando as medidas finais são conhecidas <i>a posteriori</i>	10
Figura 8- Itens da instância usada para a explicação das heurísticas e conceitos.	10
Figura 9- Empacotamento com a heurística NFDH.....	11
Figura 10- Empacotamento com a heurística FFDH.	12
Figura 11- Empacotamento com a heurística MFFDH.....	13
Figura 12- Empacotamento com a heurística AFFDH.	14
Figura 13- Ordenação dos itens por ordem decrescente de área.	15
Figura 14- Empacotamento com a heurística LGFi _{OF}	16
Figura 15- Itens ordenados por ordem decrescente de largura.....	17
Figura 16- Empacotamento com a heurística BL usada neste trabalho.	17
Figura 17- Espaço disponível entre dois níveis sucessivos para alocação de itens.	19
Figura 18- Empacotamento com a heurística BLFFDH.	20
Figura 19- Espaço disponível entre dois níveis sucessivos para criação de sub-objectos.	21
Figura 20- Diferença do PR entre a heurística BLFFDH e BLFFDH _G	21
Figura 21- Empacotamento com a heurística BLFFDH _G	22
Figura 22- Altura máxima numa heurística orientada por níveis (a) e numa heurística não orientada por níveis (b).	23
Figura 23 - Ciclo de simulação.....	31
Figura 24- Instância Nice (a) e Path (b).....	33

LISTA DE TABELAS

Tabela 1- Média e desvio padrão para as instâncias reais.....	24
Tabela 2- Grupo de instâncias da literatura usado nos testes computacionais.....	24
Tabela 3- Média e desvio padrão para as instâncias da literatura.....	26
Tabela 4- Taxa de variação em relação à altura máxima da heurística BLFFDH _G	27
Tabela 5- Taxa de variação em relação à média da altura óptima de cada instância.....	29
Tabela 6- Percentagem de vezes que uma heurística apresenta a melhor solução face ao total do tipo de instâncias.....	30
Tabela 7- Tempo (em milissegundos) por ciclo de simulação em instâncias reais.....	32
Tabela 8- Tempo (em milissegundos) por ciclo de simulação para instâncias da literatura.....	32
Tabela 9- Coeficientes de correlação.....	33
Tabela 10- Resultados de R ² para as regressões quadrática e cúbicas.....	34

LISTA DE ACRÓNIMOS

- 1D:** Uma Dimensão
- 2D:** Duas Dimensões
- 2DRSBSBPP:** Two-Dimensional Rectangular Single Bin Size Bin Packing Problem
- 3D:** Três Dimensões
- AFFDH:** Adaptive First Fit Decreasing Height
- BFDH:** Best Fit Decreasing Height
- BL:** Bottom-up Left-justified
- BLF:** Bottom-left Fill
- C&E:** Corte e Empacotamento
- FBL:** Finite Bottom Left
- FBS:** Finite Best Strip
- FFDH:** First Fit Decreasing Height
- FFF:** Finite First Fit
- FNF:** Finite Next Fit
- LGF_i:** Improved Lowest Gap Fill
- LGF_{iOF}:** Improved Lowest Gap Fill non-oriented
- LP:** Lado Pequeno
- MFFDH:** Modified First Fit Decreasing Height
- N/A:** Não Aplicável
- NFDH:** Next Fit Decreasing Height
- PR:** Ponto de Referência
- S1R:** Série 1 de Roupeiros
- SF:** Split Fit
- SFFDH:** Split First Fit Decreasing Height
- VBA:** Visual Basic for Applications

1. INTRODUÇÃO

Os problemas de corte e empacotamento (C&E) são problemas de otimização combinatória similares. Ambos consistem em alocar num conjunto de objectos - *bins* (por exemplo, placas, folhas, *input*, oferta) um conjunto de itens (por exemplo, peças, rectângulos, *output*, procura) de igual ou menor dimensão. Todos os itens devem estar contidos nos objectos e não se podem sobrepor. O padrão produzido deve ser tal que minimize o desperdício de *input*¹ ou matéria-prima. Os objectos e itens definem-se com uma (1D), duas (2D), três (3D) ou mais dimensões, dependendo do tipo de problema. Uma solução do mesmo pode resultar no uso de alguns ou todos os objectos e itens (Wäscher, Haussner, & Schumann, 2007).

A ênfase deste trabalho vai para os problemas bidimensionais de *strip packing* e *bin packing*. Neste último, conhecido na tipologia de Wäscher (Wäscher *et al*, 2007) como *Two-Dimensional Rectangular Single Bin Size Bin Packing Problem* (2DRSBSBPP), é dado um número ilimitado de objectos, de dimensões finitas, com uma largura W e altura H , e um número finito de itens rectangulares a alocar, de largura $w_j \leq W$ e altura $h_j \leq H$.

Nos problemas bidimensionais de *strip packing* tem-se uma faixa, ou objecto, com largura W finita e uma altura infinita. Os itens rectangulares a alocar têm também uma dimensão finita, com uma largura $w_j \leq W$ e altura h_j .

Os problemas de C&E têm diversas aplicações práticas na indústria e comércio, desde a disposição de artigos num jornal ao carregamento de materiais em contentores, paletes ou camiões, passando por aplicações em domínios tão diversos como o da informática, na distribuição da memória de um computador. Até mesmo em aplicações mais abstractas, envolvendo dimensões não espaciais, como o peso, o tempo ou mesmo financeiras, se encontram problemas deste tipo (Silva, 2003).

¹ No caso de empacotamento de paletes o *input* seria o espaço.

Estes problemas pertencem à classe dos NP-difíceis devido à sua complexidade, sendo necessário elevadíssimos recursos computacionais por forma a tentar encontrar uma solução óptima, especialmente quando o número de itens a alocar é alto. Daí que se utilize, na maioria das vezes, algoritmos que recorrem a métodos heurísticos ou meta-heurísticos, em detrimento de métodos exactos, a fim de alcançar resultados o mais próximo possíveis do óptimo (Silva, 2003).

A motivação para a realização deste projecto é resolver um problema real de uma empresa portuguesa tentando, ao mesmo tempo, inovar. Este consiste no corte de peças ou itens rectangulares de roupeiro. As peças são cortadas de uma ou várias placas rectangulares, ou objectos, de iguais dimensões finitas (largura W e altura H), e não podem ser rodadas nem sobrepostas. O padrão de corte produzido deve minimizar o desperdício de matéria-prima e ser guilhotinável (corte de ponta-a-ponta), permitindo assim que as máquinas industriais consigam cortar os itens.

Este estudo tem o objectivo de:

- Propor duas novas heurísticas de empacotamento de itens rectangulares com orientação para o caso de duas dimensões (2DRSBSBPP), em que uma tem uma restrição adicional de corte guilhotina;
- Testar o desempenho das heurísticas, comparando-as com outras presentes na literatura de *strip packing* e *bin packing*;
- Fazer uma análise dos resultados para ambas as heurísticas.

No capítulo 2 é feita uma revisão da literatura, sendo referida a tipologia de Wäscher, os conceitos e terminologia usados ao longo do trabalho.

Detalham-se algumas heurísticas orientadas e não orientadas por níveis, nos capítulos 3 e 4, que são usadas nos testes computacionais.

No capítulo 5 propõem-se duas novas heurísticas cuja diferença reside nas restrições impostas. Uma delas tem mais uma restrição, a de originar um padrão guilhotinável.

No capítulo 6 efectuaem-se testes computacionais. Os programas subjacentes foram implementados em *Visual Basic for Applications* (VBA) (ver, por exemplo, (Halvorson, 2010)). Nestes testes, comparam-se as novas heurísticas, quer entre si quer com outras da literatura, pormenorizadas nos capítulos 3 e 4. São utilizadas dois grupos de instâncias: reais e da literatura. No primeiro grupo, consideram-se instâncias de roupeiros que foram extraídas de obras efectuadas por uma empresa. Da literatura são usadas as instâncias de Wang & Valenzuela (2001), Hopper (2000), Burke & Kendall (1999) e Jakobs (1996). Neste capítulo é também determinada a complexidade empírica das heurísticas propostas.

No capítulo 7 é feita uma síntese do trabalho e da comparação das novas heurísticas com as da literatura e são dadas sugestões para trabalhos futuros.

2. PROBLEMA DE CORTE GUILHOTINA BIDIMENSIONAL

2.1. Enquadramento

Nos problemas de *strip packing* e *bin packing*, entre outros, podem dividir-se as heurísticas em três grupos, as orientadas por níveis, as não orientadas por níveis e as híbridas. Este último grupo não será alvo de estudo neste trabalho.

As heurísticas orientadas por níveis são aquelas cujos itens são colocados em diferentes níveis (ou fileiras) do objecto, justificados à esquerda. O empacotamento é construído por uma sequência de níveis, em que cada item é alocado com a aresta inferior a repousar num desses níveis. A altura do nível é determinada pelo item mais alto nele alocado (Hopper, 2000).

Nas heurísticas orientadas por níveis destacam-se as clássicas de *strip packing*: *Next Fit Decreasing Height* (NFDH); *First Fit Decreasing Height* (FFDH); e *Split Fit* (SF), de Coffman, Garey & Tarjan (1980). Coffman & Shor (1990) introduziram e estudaram em detalhe a *Best Fit Decreasing Height* (BFDH). Rode & Rosemberg (1987) propuseram a *Modified First Fit Decreasing Height* (MFFDH), que, embora similar às clássicas, cria sub-objectos rectangulares para conseguir aproveitar o espaço que sobra entre os diferentes níveis.

Com este mesmo objectivo, e baseadas na MFFDH, também Silva (2003) apresentou a *Adaptive First Fit Decreasing Height* (AFFDH) e a *Split First Fit Decreasing Height* (SFFDH). No entanto, estas são mais flexíveis, com o sub-objecto da AFFDH a poder assumir uma forma irregular para se adaptar aos itens alocados num nível e a SFFDH a poder criar mais do que um sub-objecto no mesmo nível para o mesmo efeito.

Berkey & Wang (1987) propuseram as heurísticas *Finite Next Fit* (FNF) e *Finite First Fit* (FFF), que são adaptações das clássicas para a resolução de problemas de *bin packing*, cujo objecto tem dimensões finitas. Analogamente, apresentaram também a *Finite Best Strip* (FBS), em que os itens são alocados como num problema de *strip packing* e posteriormente os níveis são alocados a um objecto de dimensões finitas, e a *Finite Bottom Left* (FBL), não orientada por níveis.

Nas heurísticas não orientadas por níveis salientam-se, por exemplo, a *Bottom-up Left-justified* (BL), proposta por Baker *et al* (1980) e aperfeiçoada em 1999 por Liu & Teng, segundo Bortfeldt (2006); a *Bottom-left Fill* (BLF), apresentada por Chazelle (1983); e a *Improved Lowest Gap Fill* para o caso orientado (LGF_i) e para o caso não orientado (LGF_{iOF}), propostas por Wong & Lee (2009).

Todas as heurísticas referidas nesta secção, e apresentadas no capítulo 5, não usam a informação de qual a altura óptima, isto é, não necessitam de saber que o desperdício é, por exemplo, zero. As heurísticas que usam esta informação reformulam o empacotamento assim que o desperdício é superior ao do óptimo, caso haja.

Distinguem-se também as heurísticas que funcionam em uma fase das de duas fases. As heurísticas de uma fase alocam os itens directamente no objecto, enquanto as de duas fases alocam os itens em níveis a um objecto num problema de *strip packing*, e depois organizam esses níveis em objectos de dimensões finitas, *bin packing*.

Os problemas podem ser *offline*, quando o conjunto de itens a ser empacotado é conhecido antes de ser iniciado o empacotamento, ou *online* quando o próximo item só é conhecido depois de alocar o item actual, e uma vez alocado nunca poderá mudar a sua posição. Existem heurísticas que funcionam em ambas as situações, como o NFDH. O trabalho actual cinge-se a problemas do tipo *offline*.

Neste capítulo aborda-se a tipologia de Wäscher, apresenta-se a terminologia e serão explicados os conceitos. Para explicar os conceitos e heurísticas é utilizado um conjunto de itens uniforme ao longo do trabalho, possibilitando a comparação do padrão produzido por cada uma. Na explicação de conceitos são seleccionados os itens que melhor os evidenciem e expliquem, enquanto nas heurísticas todos os itens são utilizados. O objecto define-se com dimensões finitas ou variáveis, consoante a heurística e o conceito subjacente.

2.2. Tipologia de Wäscher

Wäscher, Haussner & Schumann (2007) publicaram uma tipologia de modo a criar uma base comum aos problema de C&E. Segundo (Ortmann, 2010), esta surge para colmatar algumas deficiências que foram ficando evidentes na tipologia proposta por Dyckhoff em 1990.

Como tal, é usada a tipologia de Wäscher para enquadrar e facilitar a comparação do trabalho nos demais da literatura. Esta tipologia usa cinco critérios para a definição da combinação de problemas de C&E, sendo eles: dimensões, tipo de alocação, variedade de objectos, variedade de itens e forma dos itens. O problema abordado neste trabalho enquadra-se nos de tipo *Bin Packing Problem* devido às seguintes caracterizações:

- Dimensão
 - Os itens e objectos têm duas dimensões (2D).
- Tipo de alocação
 - Minimização de *input*, no qual um dado conjunto de itens tem de ser alocado a um dado conjunto de objectos, sendo os objectos suficientes para acomodar todos os itens. Todos os itens têm de ser alocados a uma selecção (sub-conjunto) de objectos de valor mínimo. O valor pode representar a área, o custo, a quantidade de material ou a receita. Neste trabalho quer-se minimizar a área desperdiçada.
- Variedade de objectos
 - Vários objectos de iguais dimensões e finitas, com forma rectangular e formados por material homogéneo.

- Variedade de itens
 - Variedade altamente heterogénea, pois são poucos os itens de forma e tamanho semelhantes. Como tal, os itens são tratados como elementos individuais com procura igual a um.
- Forma dos itens
 - Os itens têm uma forma rectangular.

Posto isto, e como referido na Introdução, o problema tem o nome de *Two-Dimensional Rectangular Single Bin Size Bin Packing Problem* (2DRSBSBPP).

2.3. Terminologia

O problema em estudo consiste, formalmente, em alocar, sem sobreposição nem rotação, um conjunto, $N = \{1, \dots, n\}$, de itens rectangulares no mínimo número de objectos rectangulares, $P = \{1, \dots, p\}$, e da melhor forma possível, minimizando assim o consumo de matéria-prima. Um item $i \in N$ é caracterizado pela sua largura w_i e altura h_i , enquanto os objectos $j \in P$ são iguais e caracterizados pela largura W e altura H (Polyakovsky & Hallah, 2009).

Os itens rectangulares a serem alocados não podem exceder as dimensões do objecto, assim $w_i \leq W$ e $h_i \leq H, \forall i \in N$.

2.4. Conceitos

Nesta secção são introduzidos os conceitos usados ao longo do trabalho, usando itens de uma instância criada para o efeito de modo a ajudar a sua interpretação, Figura 8.

2.4.1. Corte guilhotina

Quando se cortam certos tipos de materiais, como vidro ou madeira, pode ser necessário que os itens tenham de ser cortados da placa em cortes guilhotina, ou, cortes de ponta-a-ponta, como se pode ver na Figura 1(b). Esta mostra os três primeiros cortes guilhotina que iniciam separação de todos os itens. Os primeiros dois cortes separam o objecto em três partes e o terceiro separa uma das partes em duas.

A Figura 1(a) não permite que a máquina usada na empresa, que esteve na base do estudo, faça a separação dos itens sem que alguns sejam cortados em dois. Daí advém a necessidade de impor o corte guilhotina.

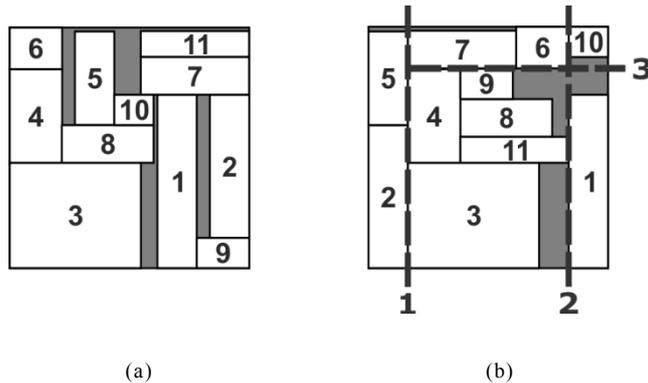


Figura 1- Padrão: (a) de não guilhotina; (b) de guilhotina.

2.4.2. Nível

Um nível tem a largura do objecto ou sub-objecto em questão, e a sua altura é igual à do primeiro item nele alocado. O primeiro nível é criado com a aresta inferior dos itens nele alocados assente na aresta inferior do objecto. A parte de baixo de cada um dos níveis subsequentes é definida por uma linha horizontal, fronteira, traçada pela altura do primeiro item (logo, o mais alto) colocado no nível anterior, como se ilustra na Figura 2.

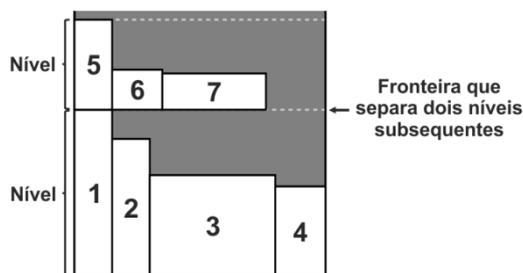


Figura 2- Definição de dois níveis e da fronteira que os separa.

2.4.3. Ponto de referência

O ponto de referência (PR) é representado pelas coordenadas (x, y) e indica o ponto mais em baixo e à esquerda no objecto ou nível em análise onde é permitido alocar um item, sem o sobrepor a outros já alocados.

Quando um item é alocado a um objecto faz-se coincidir o vértice inferior esquerdo desse item com o PR. Este move-se imediatamente para o vértice inferior direito desse item. Se não for possível alocar um novo item a esse novo PR, então este é novamente actualizado, incrementando-se, sempre que possível, a abcissa antes da ordenada. A Figura 3 exhibe vários PR possíveis ao longo de um empacotamento.

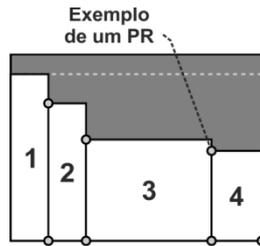


Figura 3- Pontos de referência.

2.4.4. Gap

O *gap* é uma área rectangular definida sempre que o ponto de referência é actualizado. A sua altura e largura chamam-se *gap* vertical e *gap* horizontal, respectivamente. O *gap* vertical é então o comprimento do segmento de recta vertical que une o PR ao topo do nível, Figura 4(a), ou objecto, Figura 4(b), conforme a heurística. A sua largura é o comprimento do segmento de recta horizontal do PR ao lado direito do objecto, Figura 4(a), ou até ao lado esquerdo do primeiro item intersectado, Figura 4(b), caso o segmento intersecte um item. A Figura 4(a) ilustra o *gap* num algoritmo orientado por níveis e a Figura 4(b) num algoritmo não orientado por níveis.

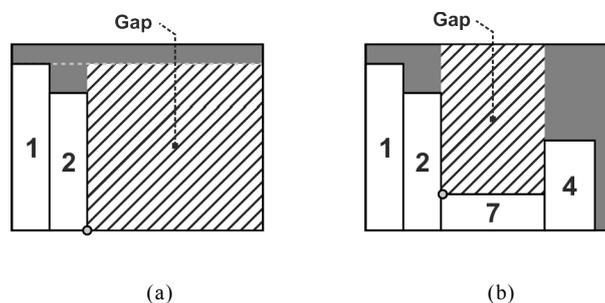


Figura 4- Gap e PR em: (a) algoritmos orientados por níveis; (b) algoritmos não orientados por níveis.

Define-se o *gap* vertical e horizontal para se saber, em cada instante, qual é o espaço disponível, em relação ao PR, para alocar itens. Só depois de se conhecer esse espaço é procurado um item para alocar no PR. Se não for possível alocar nenhum item ao PR, tendo em conta o *gap* calculado, então o PR é actualizado e um novo *gap* determinado.

O *gap* vertical e *gap* horizontal pode ser ignorado quando se aloca o primeiro item no objecto (o maior item tem de caber no objecto) ou num determinado nível em problemas de *strip packing* e algoritmos orientados por níveis.

2.4.5. Sub-objecto

Um sub-objecto possui dimensões finitas e estas podem ser conhecidas *a priori* ou *a posteriori*, conforme a heurística utilizada. Pode ser criado um, vários ou nenhum no espaço que sobra entre a alocação dos itens no nível actual e o topo desse nível ou início do próximo, caso venha a ser formado, como se mostra na Figura 5.



Figura 5- Espaço destinado a sub-objecto(s).

Nas heurísticas MFFDH e AFFDH aqui apresentadas, as dimensões do sub-objecto são conhecidas *a priori* e nesta última a geometria pode assumir uma forma irregular, Figura 6(b). Por sua vez, um sub-objecto pode comportar vários níveis em ambas as heurísticas e a sua criação não depende do *gap*, uma vez que o sub-objecto é definido independentemente de lhe ser possível alocar itens.

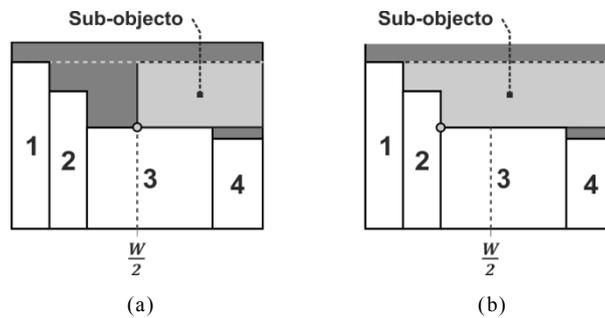


Figura 6- Sub-objectos: (a) com forma rectangular; (b) com forma irregular.

Na heurística BLFFDH_G aqui proposta, a criação do sub-objecto depende do *gap* e apenas comporta um nível. Nestes casos, um sub-objecto só é criado se for possível alocar pelo menos um item a esse *gap* e as suas dimensões finais são apenas conhecidas *a posteriori* mas sem nunca exceder as do *gap* que lhe deram origem. A sua altura é igual à do primeiro item e a sua largura vai desde o PR que o originou até ao lado direito do último item que nele for possível alocar, Figura 7(c).

A Figura 7 ilustra o processo de criação deste tipo de sub-objecto, desde o PR que o originou, (a), até não ser possível alocar mais itens, (b), determinando assim, *a posteriori*, as suas dimensões finais, (c).

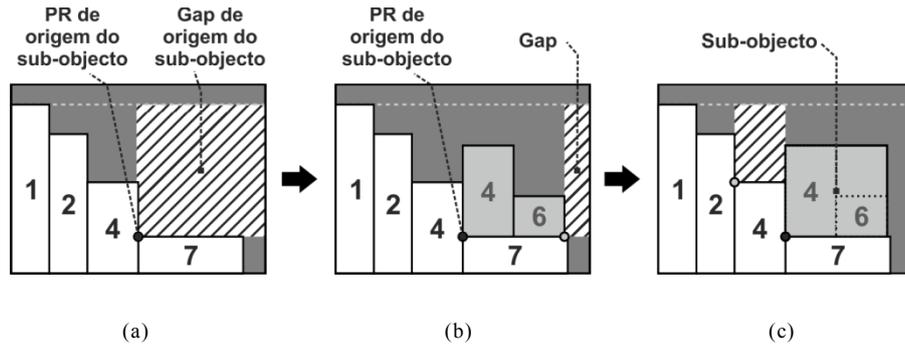


Figura 7- Criação de um sub-objecto quando as medidas finais são conhecidas *a posteriori*.

3. HEURÍSTICAS ORIENTADAS POR NÍVEIS

3.1. Introdução

As heurísticas de C&E aqui apresentadas supõem que os itens em N estão numerados de acordo com a sua correspondente ordenação decrescente por altura e não é permitida a sua rotação. Não sendo mencionado como são desempatados os itens da mesma altura, neste trabalho o desempate é feito por ordem decrescente de largura. Como se verá, o padrão de empacotamento produzido por este tipo de heurísticas garante o corte guilhotina.

A Figura 8 ilustra a ordenação dos itens da instância usada para a explicação das heurísticas e conceitos. A numeração destes itens também respeita a ordenação anteriormente referida.

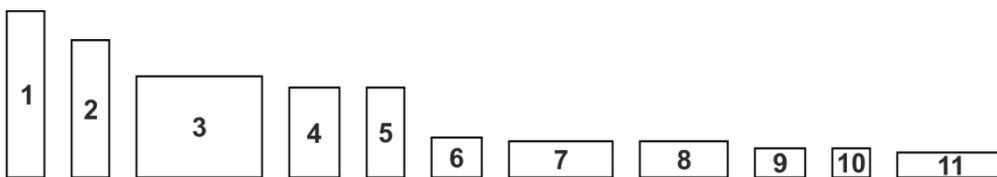


Figura 8- Itens da instância usada para a explicação das heurísticas e conceitos.

3.2. Next-Fit Decreasing-Height (NFDH)

No algoritmo NFDH, proposto por (Coffman, Garey, & Tarjan, 1980), os itens são justificados à esquerda num nível até não haver espaço à direita para acomodar o próximo item. Nesse ponto, o próximo nível é definido, o empacotamento do nível actual descontinuado, continuando o empacotamento no novo nível. Os itens são empacotados sequencialmente, ou seja, o item $i + 1$ só pode ser empacotado depois do item i .

A Figura 9 mostra um empacotamento utilizando esta heurística. Quando o primeiro nível não conseguir acomodar mais itens, é aberto um novo nível, Figura 9(a). A altura deste é definida aquando do alocamento do item 5, e o mesmo é fechado quando não comportar mais itens, Figura 9(b), abrindo um novo nível, Figura 9(c), até todos os itens serem alocados Figura 9(d).

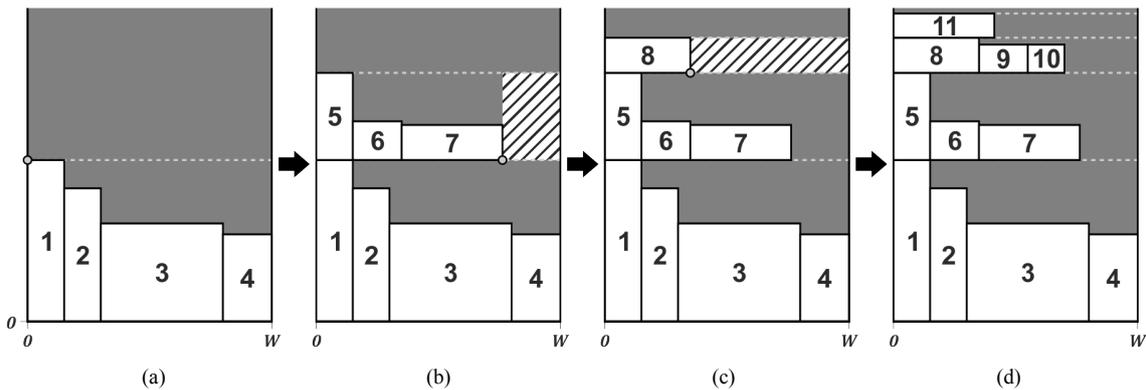


Figura 9- Empacotamento com a heurística NFDH.

3.3. First-Fit Decreasing-Height (FFDH)

Em qualquer ponto da sequência de empacotamento da heurística FFDH (Coffman, Garey, & Tarjan, 1980), o próximo item a ser empacotado é colocado no primeiro nível, a contar da parte de baixo do objecto, em que encaixe, sendo justificado à esquerda. Se nenhum dos níveis actuais conseguir acomodar este item, então é iniciado um novo nível como no algoritmo NFDH.

Uma diferença essencial entre os dois algoritmos é que em FFDH se pode sempre retroceder a um nível anterior para empacotar um novo item, enquanto em NFDH só se pode empacotar os itens subsequentes no nível actual ou posterior.

A Figura 10 exibe um empacotamento utilizando esta heurística. Quando o primeiro nível não conseguir acomodar mais itens, é aberto um novo nível com altura definida aquando do acomodamento do item 5, Figura 10(a). Como o item 8 não cabe neste nível, é aberto um terceiro para o acomodar, Figura 10(b). O PR volta ao nível anterior ainda não completo na tentativa de alocar o próximo item, Figura 10(c), contrariamente ao que sucedera com NFDH, Figura 9(c). O processo continua até todos os itens terem sido alocados, Figura 10(d).

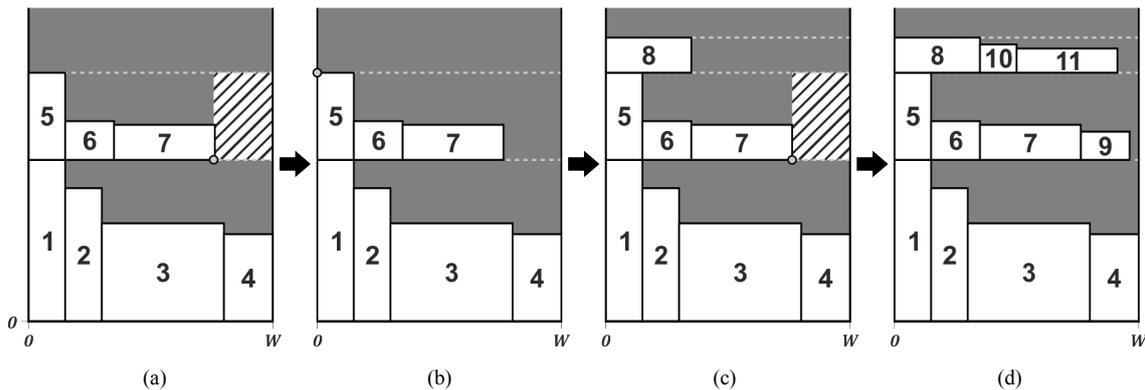


Figura 10- Empacotamento com a heurística FFDH.

3.4. Modified First Fit Decreasing Height (MFFDH)

A heurística MFFDH (Rode & Rosenberg, 1987) é similar à FFDH, no entanto quando um dos itens ultrapassa metade da largura do objecto ($W/2$) é criado, se houver espaço, um sub-objecto rectangular, com largura ($W/2$) e altura definida pela distância entre o cima desse item e o topo do nível, como se ilustra na Figura 11(a). A partir desse momento, as peças podem ser posicionadas neste sub-objecto utilizando o algoritmo FFDH.

A alocação de itens prossegue, empacotando os restantes, de acordo com o algoritmo FFDH, até um item noutra nível ultrapassar metade da largura do objecto ($W/2$), originando um novo sub-objecto. O empacotamento pára quando não houver mais itens.

A Figura 11 mostra um empacotamento utilizando esta heurística. Quando o item 3 é empacotado, é ultrapassada metade da largura do objecto, sendo criado um sub-objecto por cima desse item, Figura 11(a). Prosseguindo a ordem dos itens, o item 4 é alocado no sítio mais em baixo onde encaixa, Figura 11(b). O sub-objecto não tem espaço para acomodar o item 5, sendo aberto um novo nível, mas consegue acondicionar o item 6, Figura 11(b). Os itens 7 e 8 só encaixam no mesmo nível do 5, sendo um novo sub-objecto aberto por cima do nível 7, Figura 11(c). Continuando a aplicar a heurística FFDH até não haver mais itens, é produzido o padrão da Figura 11(d).

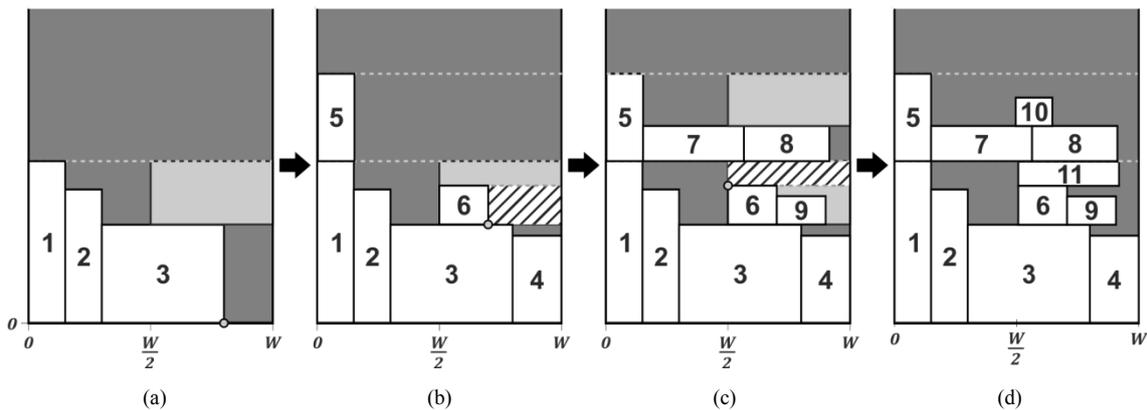


Figura 11- Empacotamento com a heurística MFFDH.

3.5. Adaptive First Fit Decreasing Height (AFFDH)

A heurística AFFDH (Silva, 2003) é similar à MFFDH, na criação do sub-objecto. Quando não cabem mais itens num determinado nível, é criado um sub-objecto por cima do item que ultrapassa a metade da largura do objecto ($W/2$), se houver espaço. A altura máxima do novo sub-objecto é definida pela distância entre o topo desse item e o topo do nível onde ele é criado, Figura 12(a).

A diferença face à heurística MFFDH em relação ao sub-objecto reside na sua largura, uma vez que este pode ser expandido para a esquerda, até aos itens posicionados no mesmo nível que lhe fazem fronteira, podendo a sua largura ser superior a $W/2$. Assim, um sub-objecto pode assumir uma forma geométrica irregular, pois adapta-se à geometria de itens alocados nesse nível, Figura 12(a). O limite direito do sub-objecto continua a ser definido pelo da largura W do objecto, Figura 12(a).

Utiliza-se novamente o algoritmo FFDH para a alocação de itens no interior desse sub-objecto, sendo fechado quando nele não couberem mais itens. A alocação de itens continua noutro nível, repetindo-se o processo descrito, e terminando quando não existirem mais itens de N por alocar.

A Figura 12exibe um empacotamento utilizando a heurística AFFDH. O sub-objecto é criado por cima do item que ultrapassa metade da largura do objecto, item 3, quando num nível não é possível encaixar mais itens, Figura 12(a). Nesse ponto, o processo de alocação passa para o sub-objecto, Figura 12(b), até não ser possível acomodar mais itens, Figura 12(c). A lógica mantém-se nos outros níveis até não haver mais itens por alocar, Figura 12(d).

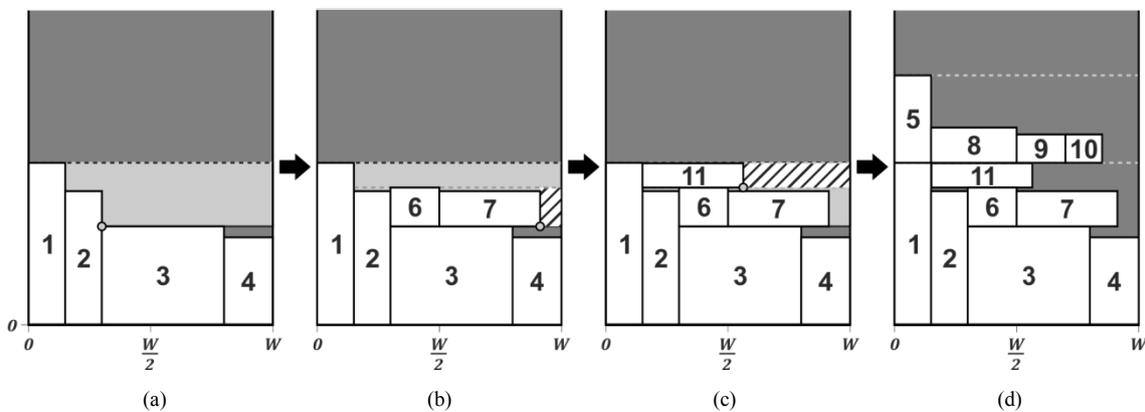


Figura 12- Empacotamento com a heurística AFFDH.

4. HEURÍSTICAS NÃO ORIENTADAS POR NÍVEIS

4.1. Introdução

As heurísticas apresentadas nesta secção têm em comum o facto de não garantirem um padrão guilhotinável, não admitirem a rotação dos itens e usarem a estabilidade *bottom-left*. A estabilidade *bottom-left* considera a alocação dos itens no ponto mais em baixo e à esquerda possível, dependendo a definição deste ponto da heurística considerada. A ordenação dos itens, diferindo com os métodos, é expressa na explicação de cada heurística.

Os itens da instância utilizada, para ilustrar conceitos e heurísticas, mantêm a numeração da secção anterior de modo a facilitar a comparação entre os vários algoritmos.

4.2. Improved Lowest Gap Fill (LGF_{IOF})

A heurística LGF_{IOF} (Wong & Lee, 2009) é uma versão modificada da LGFi (Wong & Lee, 2009) para resolver instâncias em que a rotação não é permitida, ou seja, quando os itens têm orientação. Esta heurística tem duas etapas: a fase de pré-processamento e a fase de empacotamento.

No pré-processamento os rectângulos são ordenados por ordem decrescente de área, Figura 13. O desempate é feito por ordem crescente das diferenças entre a largura e altura de cada item, dando preferência a um item com maior altura em detrimento de um com maior largura.

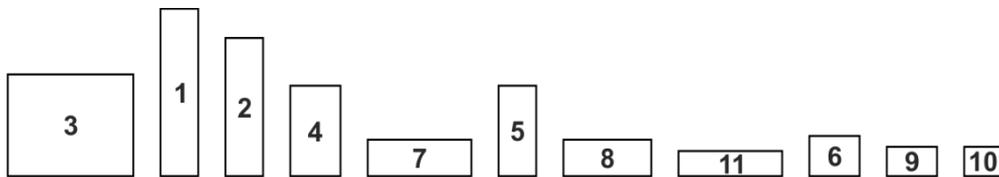


Figura 13- Ordenação dos itens por ordem decrescente de área.

Na fase de empacotamento começa-se por procurar e guardar LP, medida do lado mais pequeno dos itens por empacotar, ou seja, o mínimo entre a altura e largura dos itens por alocar. Esse valor é actualizado quando o correspondente item for empacotado.

O primeiro item da lista é colocado no canto inferior esquerdo do objecto, o PR é actualizado, movendo-se para a direita. O *gap* horizontal e vertical também é actualizado, de acordo com a nova posição do PR. Nesta heurística define-se ainda o *gap* actual como sendo o mínimo entre o *gap* vertical e o *gap* horizontal. Se o valor do *gap* actual for menor que LP, então o PR é mudado para outra posição, alterando, primeiramente a sua abcissa, se possível, e depois a ordenada.

De seguida, a lista de itens é novamente examinada e se o *gap* actual coincidir com o *gap* horizontal é dada prioridade ao item cuja largura o consiga preencher na totalidade. De forma análoga, é aplicado o mesmo princípio se o *gap* actual for vertical, mas tendo em conta a altura dos itens.

Caso não exista nenhum rectângulo que consiga preencher na totalidade o *gap* actual, é seleccionado o primeiro rectângulo da lista que caiba no *gap* horizontal e vertical, Figura 14(a) e (b).

Quando ainda existem itens por alocar e um objecto está cheio, ele é fechado, um novo objecto é inicializado e o processo repete-se. Apenas se pode criar um objecto de cada vez.

A Figura 14 ilustra um empacotamento com esta heurística. O item 8 é escolhido para a sua posição por ser o primeiro que preenche na totalidade o *gap* actual, que era, neste caso, o *gap* horizontal. Após alocar este item, não foi possível continuar a aumentar a abcissa, então o PR subiu e deslocou-se para a esquerda, Figura 14(a). O item 2 foi o primeiro que, embora não preencha o *gap* actual, cabia no *gap* vertical e horizontal. Depois de alocar o item 4, o PR volta a ser actualizado (sobe e a justifica-se à esquerda), Figura 14(b). O processo continua com a mesma lógica até não haver itens por alocar, Figura 14(d).

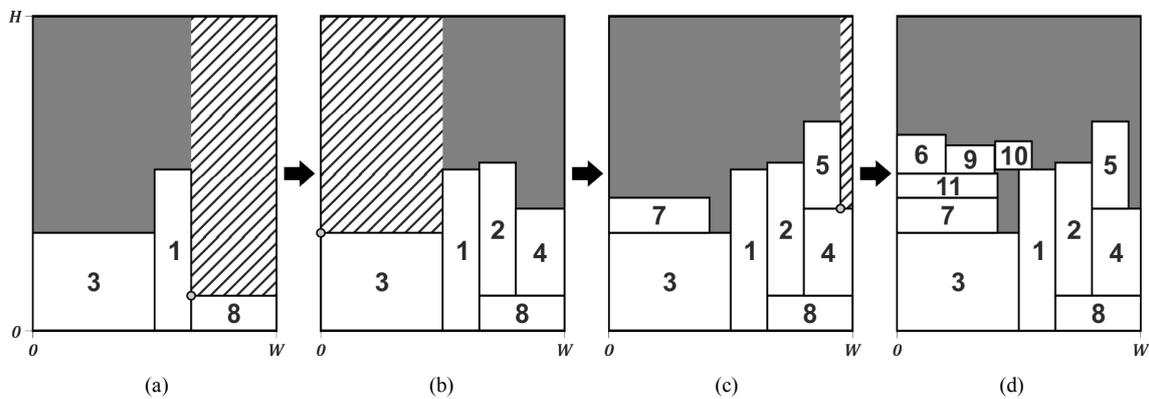


Figura 14- Empacotamento com a heurística LGFiOF.

4.3. Bottom-up left-justified (BL)

Bottom-up left-justified (BL) é uma classe de heurísticas, apresentada por (Baker, Coffman, & Rivest, 1980), em que os itens são empacotados na localização mais em baixo possível e justificados à esquerda, como na LGFiOF anteriormente apresentada. Existem inúmeras heurísticas que preservam a estabilidade *bottom-left*, em que se estabelecem regras na escolha do ponto mais em baixo e à esquerda.

Na heurística BL desenvolvida por (Baker, Coffman, & Rivest, 1980), não existe a obrigação de ordenação dos itens, o que, segundo estes, leva a um empacotamento pobre. Os autores mencionam uma melhoria na qualidade de empacotamento se os itens forem ordenados por ordem decrescente de largura, face à não ordenação ou mesmo à ordenação por ordem decrescente de altura.

Infelizmente os autores não providenciaram uma explicação detalhada de um método que possa ser usado para realizar computacionalmente o empacotamento (Ortmann, 2010).

Na heurística BL, aqui programada e usada, os itens são então ordenados por ordem decrescente de largura, Figura 15. A alocação dos itens não é feita sequencialmente, isto é, em cada iteração examina-se a lista de itens, sendo seleccionado o primeiro item compatível com os *gap* vertical e horizontal.

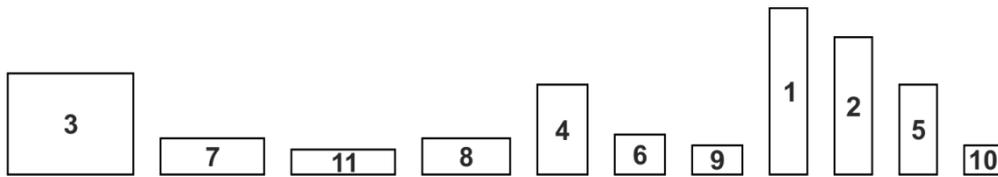


Figura 15- Itens ordenados por ordem decrescente de largura.

A Figura 16 ilustra um empacotamento utilizando esta heurística. Quando o *gap* vertical ou horizontal da Figura 16(a) não consegue acomodar um item, o PR é actualizado para o próximo ponto mais em baixo e justificado à esquerda, Figura 16(b) e (c). O padrão produzido é visível na Figura 16(d).

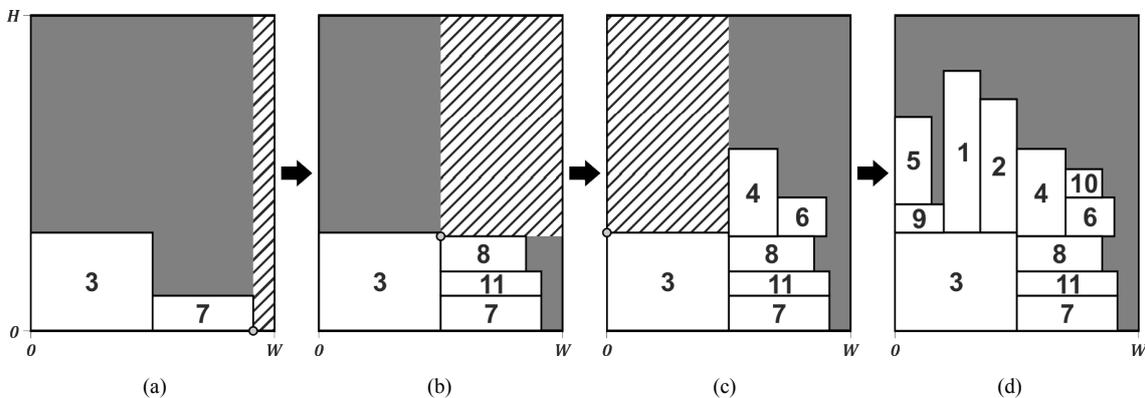


Figura 16-Empacotamento com a heurística BL usada neste trabalho.

5. HEURÍSTICAS PROPOSTAS

5.1. Introdução

Nas heurísticas de C&E, o desafio que se coloca é como melhorar o padrão de empacotamento. Uma das maneiras de o fazer é usando técnicas de modo a conseguir alocar itens nos espaços que sobram entre níveis.

Várias heurísticas foram desenvolvidas tentando aproveitar estes espaços, designadamente criando níveis mais pequenos e preenchendo-os com técnicas similares dos níveis maiores, como por exemplo as heurísticas MFFDH e AFFDH.

Nesse sentido, são propostas duas novas heurísticas orientadas por níveis, a *Bottom Left First Fit Decreasing Height* (BLFFDH), para problemas que não exijam um padrão guilhotinável, e a *Guillotinable Bottom-Left First Fit Decreasing Height* (BLFFDH_G), para problemas em que seja necessário assegurar um padrão guilhotinável. Em ambas os itens não podem sofrer rotação, ou seja, os itens têm orientação.

Estes dois algoritmos têm na sua génese a combinação de duas das heurísticas anteriormente descritas, a BL e a FFDH. Nestas introduzem-se duas alterações fundamentais: os itens não são alocados sequencialmente, isto é, o item $i + 1$ pode ser empacotado antes do item i , e não é possível voltar aos níveis anteriores, como acontece em FFDH. Os itens são escolhidos do conjunto N , a começar pelo mais alto, consoante o espaço disponível. Na heurística BLFFDH_G, e em relação à heurística BL, o posicionamento dos sub-objects mantém a estabilidade *bottom-left* embora com determinadas regras.

Estas heurísticas supõem que os itens estão ordenados e numerados em N por ordem decrescente de altura, com desempate por ordem decrescente de largura (como na Figura 8).

5.2. Bottom-Left First Fit Decreasing Height (BLFFDH)

No algoritmo BLFFDH começa-se pela inicialização de um objecto, designado por objecto actual. O PR é definido como sendo o vértice inferior esquerdo desse objecto ($x = 0; y = 0$) e, neste caso, as dimensões do *gap* vertical e horizontal são iguais à altura e largura do objecto, respectivamente.

O processo de empacotamento inicia-se utilizando a heurística FFDH na alocação de itens, sendo o primeiro nível definido aquando da alocação do primeiro item. O próximo item a ser alocado é procurado no conjunto N , a começar pelo item de maior altura por alocar, tendo em conta o espaço disponível, segundo os *gap* vertical e horizontal.

Quando com FFDH não for possível alocar mais itens no nível em análise tenta-se alocar itens no espaço que sobra entre dois níveis consecutivos, isto é, no espaço entre a alocação dos itens no nível actual e o topo desse nível, Figura 17, recorrendo à heurística BL para posicionar o PR.

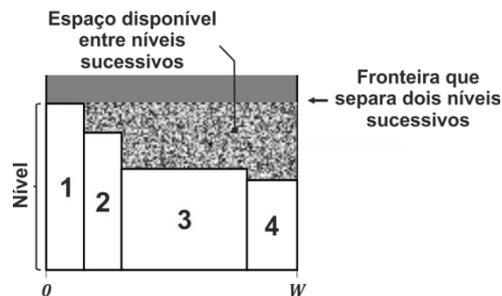


Figura 17- Espaço disponível entre dois níveis sucessivos para alocação de itens.

A área de alocação disponível é pesquisada de modo a posicionar o PR o mais em baixo e à esquerda possível, Figura 18(a). De seguida determina-se o *gap* vertical e horizontal e escolhe-se o primeiro item, do conjunto N e ainda não alocado, que nele encaixe. O PR é actualizado de novo, passando a coincidir com o vértice inferior direito desse item.

Caso não seja possível alocar um item a esse novo PR, então ele é sucessivamente actualizado até ser possível alocar um item ou até não ser possível definir um PR diferente. Quando tal acontece, esse nível é fechado e um novo nível aberto, por cima do que foi fechado, continuando o processo de alocação enquanto existirem itens por alocar. Um nível, depois de inicializado, só pode ser fechado quando já não for possível alocar-lhe mais itens ou não houver mais itens por alocar.

O objecto actual é fechado quando não comportar mais níveis. Neste caso, um novo objecto é inicializado e o processo continua até serem empacotados todos os itens.

Na Figura 18 ilustra-se um empacotamento utilizando esta heurística. Quando o primeiro nível não consegue comportar mais itens, é iniciada a pesquisa do primeiro PR no espaço que sobra entre dois níveis consecutivos, e determinado o respectivo *gap* vertical e horizontal, Figura 18(a). O processo de pesquisa continua, alterando, sempre que possível, a abcissa antes da ordenada, e mais itens são alocados, Figura 18(b). Quando não for possível definir um PR diferente, o nível actual é descontinuado e um novo nível aberto, Figura 18(c). O processo continua até não existirem mais itens por alocar, Figura 18(d).

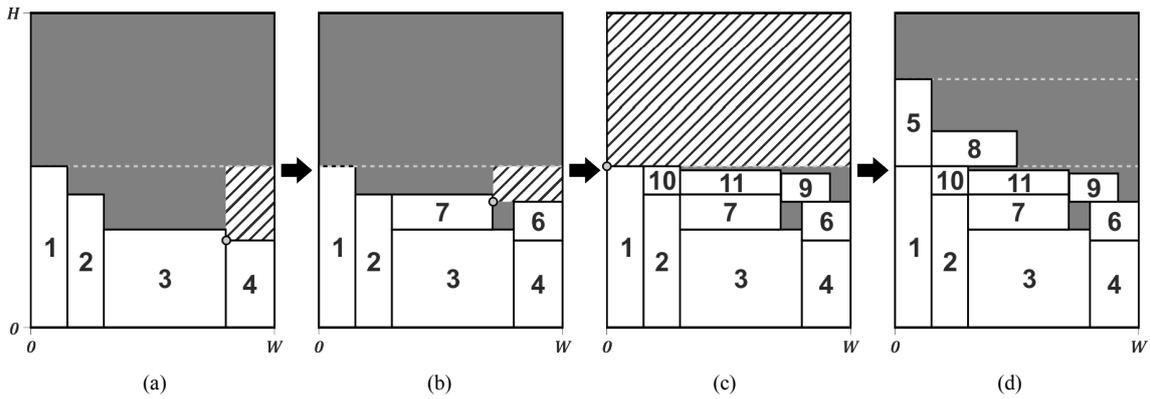


Figura 18- Empacotamento com a heurística BLFFDH.

5.3. Guillotnable Bottom-Left First Fit Decreasing Height (BLFFDH_G)

Como no método anterior, em BLFFDH_G começa-se por inicializar um objecto, sendo este o objecto actual. O PR é então definido como sendo o vértice inferior esquerdo desse objecto ($x = 0; y = 0$) e face a esse PR as dimensões dos *gap* vertical e horizontal são iguais à altura e largura do objecto, respectivamente.

O processo de empacotamento inicia-se utilizando a heurística FFDH na alocação de itens, sendo o primeiro nível definido aquando da alocação do primeiro item. O próximo item a ser alocado é procurado no conjunto N , a começar pelo de maior altura não alocado, tendo em conta o espaço disponível, segundo os *gap* vertical e horizontal.

Os itens são empacotados nesse nível enquanto existir espaço suficiente à direita para acomodar o próximo item. A partir daí é iniciada a pesquisa de sub-objectos no espaço que sobra entre dois níveis consecutivos, isto é, o espaço entre a alocação dos itens no nível actual e o topo desse nível, Figura 19. Para tal é usada uma heurística que preserva a estabilidade *bottom-left* ao posicionar o PR. A área de alocação disponível é pesquisada de modo a encontrar o PR mais em baixo e à esquerda onde é permitido definir os sub-objectos. Note-se que pode ser possível criar vários ou nenhum, consoante o espaço existente.

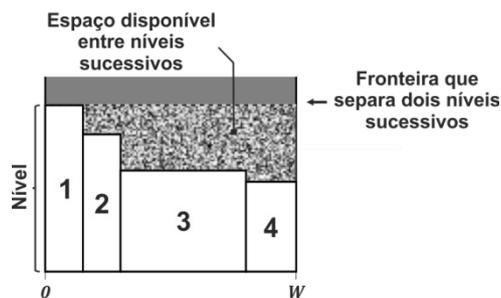


Figura 19- Espaço disponível entre dois níveis sucessivos para criação de sub-objects.

Os sub-objects podem ser alocados por cima dos itens colocados no nível actual com o FFDH, ou por cima de outros sub-objects. No entanto, o PR tem de ter a mesma abcissa desse item ou sub-objecto, ao contrário do que acontece com a heurística BLFFDH. A Figura 20 ilustra a diferença nos PR com estes dois algoritmos. Um sub-objecto apenas comporta um nível e é preenchido com a lógica de FFDH, sendo a sua altura igual à do primeiro item nele alocado e a sua largura igual à distância do PR que lhe deu origem ao vértice inferior direito do último item nele alocado, como ilustra a Figura 7.

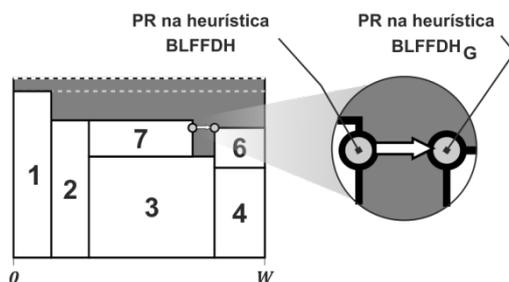


Figura 20- Diferença do PR entre a heurística BLFFDH e BLFFDH_G.

Um sub-objecto só é gerado se nele for possível alocar pelo menos um item. Para tal o PR é sucessivamente actualizado até ser possível alocar um novo item, ou até não ser possível definir um PR diferente. Neste último caso, o empacotamento desse nível é descontinuado, continuando o empacotamento num novo nível.

Assim que o objecto actual não comportar mais níveis, é fechado, um novo objecto é inicializado, continuando o processo até não existirem mais itens por alocar.

Na Figura 21 ilustra-se um empacotamento utilizando esta heurística. Quando o primeiro nível não consegue comportar mais itens, é iniciada a pesquisa do primeiro PR no espaço que sobra entre dois níveis consecutivos, e determinado o respectivo *gap* vertical e horizontal, Figura 21(a). Quando os itens 6 e 7 são alocados, não existe espaço suficiente à direita para acomodar outros itens. Nesse caso, as dimensões dos sub-objects onde esses itens se inserem são iguais aos próprios itens. Após alocar o item 7 o PR é actualizado para o vértice superior esquerdo do item 6. Note-se que só é permitido criar sub-objects no vértice superior esquerdo de outros sub-objects ou dos itens alocados (1,2,3 e 4) antes de se tentar preencher o espaço que sobra entre dois níveis consecutivos, Figura 21(b). Quando não for possível definir um PR diferente, o nível actual é descontinuado e um novo nível aberto, Figura 21(c). O processo continua até não existirem mais itens, Figura 21(d).

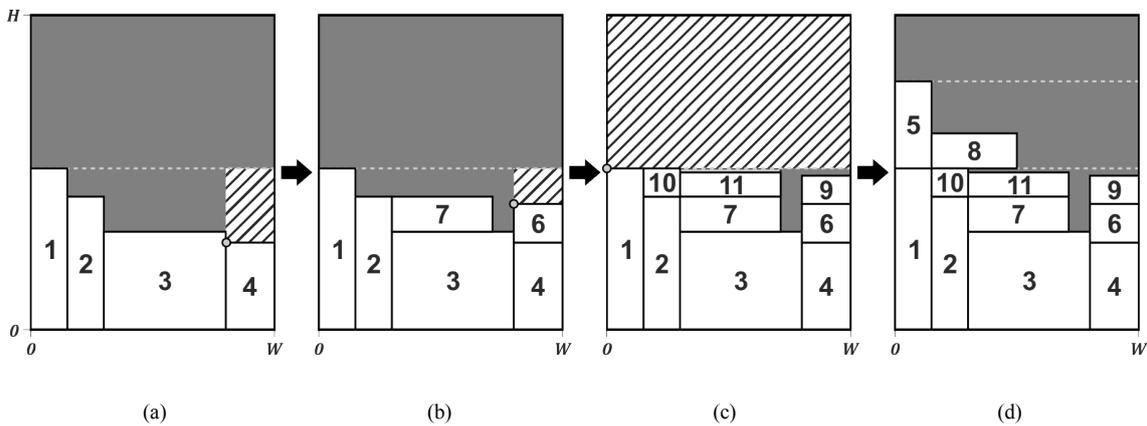


Figura 21- Empacotamento com a heurística BLFFDH_G.

6. RESULTADOS COMPUTACIONAIS

6.1. Introdução

Depois de apresentadas as novas heurísticas, interessa avaliar a qualidade das soluções que proporcionam. Para isso será feita uma comparação com heurísticas presentes na literatura. No caso das heurísticas orientadas por níveis, é utilizada a altura da última camada como termo de comparação, Figura 22(a). A comparação entre heurísticas não orientadas por níveis será feita tendo em conta a distância máxima que vai da aresta inferior do objecto até à aresta superior de um item, isto é, da aresta inferior do objecto à aresta com maior ordenada de um determinado item, Figura 22(b) (item 3).

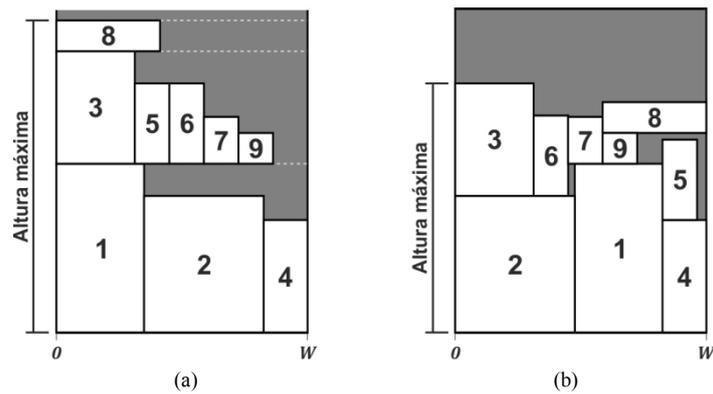


Figura 22- Altura máxima numa heurística orientada por níveis (a) e numa heurística não orientada por níveis (b).

Os métodos testados foram implementados sob o *Microsoft Windows® 7 de 32 bits*, utilizando o *Microsoft® Visual Basic® for Applications (VBA)* versão *retail 6.5.1053* com implementação em *Microsoft Excel® 2010*. O computador utilizado no conjunto de testes apresenta as seguintes características principais: processador *Intel® Core™ i3 M370* a 2.40 GHz, 4GB de memória RAM DDR3 e disco rígido de 500GB.

6.2. Instâncias

Para testar as heurísticas foram utilizados dois grupos de instâncias: instâncias reais de corte de componentes de roupeiros, obtidas numa empresa nacional, e instâncias conhecidas da literatura. Em qualquer dos testes efectuados os itens têm orientação, isto é, não é permitida rotação ortogonal. Nas instâncias da literatura, o padrão óptimo de empacotamento é guilhotinável.

6.2.1. Instâncias reais²

No grupo das instâncias reais, a base de dados criada contém o histórico de 6 obras, chamada *Série 1 de Roupeiros (S1R)*, com um conjunto de $n = 76, 245, 310, 615, 642, 1293$ itens para roupeiros. Na criação desta base de dados apenas foram consideradas as peças com espessura e características equivalentes.

² As instâncias reais podem ser extraídas de <http://sites.google.com/site/orheuristic/>

Nos casos práticos em estudo, os objectos aos quais os itens serão alocados têm uma dimensão de 2740mm×1830mm. No entanto, e para efeitos de comparação, será apenas considerada uma placa cuja altura será suficientemente grande para alocar todos os itens, podendo assim, facilmente, serem comparadas as alturas máximas, Figura 22. Embora na prática seja preciso contar com a espessura do disco de corte, cerca de 3mm, efectuado por máquinas industriais (p.ex.: esquadrejadoras, seccionadoras), nos testes tal é ignorado para facilitar o uso destas instâncias por outros autores.

A Tabela 1 exhibe a média e desvio padrão das dimensões neste grupo de instâncias.

	Instâncias	n	μ_{larg}	μ_{alt}	σ_{larg}	σ_{alt}
SIR	S1R.76	76	817,1	378,6	669,0	212,5
	S1R.245	245	747,8	329,0	620,8	216,0
	S1R.310	310	692,4	345,6	587,0	213,0
	S1R.618	618	686,1	345,5	590,5	198,8
	S1R.642	642	697,7	342,6	604,4	203,0
	S1R.1293	1293	660,5	364,1	597,5	206,0

Tabela 1- Média e desvio padrão para as instâncias reais.

6.2.2. Instâncias da literatura

A Tabela 2 resume o grupo de instâncias conhecidas da literatura que são usadas nos testes computacionais. Para todas é conhecida a solução óptima.

Classe de instâncias	Referência	N.º instâncias
Nice	(Wang & Valenzuela, 2001)	21
Path	(Wang & Valenzuela, 2001)	21
T series	(Hopper , 2000)	35
T2-Kendall	(Burke & Kendall, 1999)	1
T2-J	(Jakobs, 1996)	2

Tabela 2- Grupo de instâncias da literatura usado nos testes computacionais.

O conjunto *Path* e *Nice* é composto por 42 instâncias, com dimensões entre 25 e 2000 itens ($n = 25, 50, 100, 200, 500, 1000, 2000$). Os valores apresentados na secção 6.3.2 dizem respeito à média de 3 instâncias da mesma classe e para o mesmo valor de n . Na classe *Nice* os itens têm tamanhos e formas semelhantes enquanto na *Path* existem significativas diferenças nos tamanhos e formas dos mesmos. O objecto onde serão alocados estes itens tem 100 unidades de largura e uma altura óptima, determinada pelo criador da instância, de também 100 unidades.

De salientar que (Wang & Valenzuela, 2001) apenas fornecem um conjunto de instruções para a criação das instâncias. Nestas, existe uma componente de aleatoriedade, daí que não seja possível comparar os resultados da secção 6.3.2 directamente a outro trabalho que não utilize a mesma fonte.

As instâncias *Path* e *Nice*, com n de 25 a 500, foram extraídas de [http://dip.sun.ac.za/~vuuren/repositories/levelpaper/MumfordData\[1\].htm](http://dip.sun.ac.za/~vuuren/repositories/levelpaper/MumfordData[1].htm) (Vuuren, 2006). Já as instâncias com $n = 1000$ e 2000 foram pedidas por correio electrónico directamente a Frank G. Ortmann, tendo este generosamente cedido o conjunto aqui usado.

O conjunto *T series* é composto por 35 instâncias, com 17 a 199 itens ($n = 17, 25, 29, 49, 73, 97, 199$). As dimensões dos itens foram geradas aleatoriamente pelo autor com uma proporção máxima entre a altura e a largura, usualmente designada por *aspect ratio*, de 7 unidades. Os valores apresentados na secção 6.3.2 dizem respeito à média de 5 instâncias da mesma classe e para o mesmo n . O objecto onde serão alocados estes itens tem 200 unidades de largura e uma altura óptima de 200 unidades.

O conjunto de instâncias *T2-Kendall* e *T2-J* foram projectadas em papel. A primeira é constituída por uma instância com 13 itens a alocar num objecto com 80 unidades de largura e uma altura óptima de 140 unidades. A segunda, *T2-J*, é constituída por duas instâncias com $n = 25, 50$ a alocar num objecto com 40 unidades de largura e uma altura óptima de 15 unidades.

Os conjuntos de instâncias *T series* *T2-Kendall* e *T2-J* foram criados pelos próprios autores e extraídos do sítio da internet do *EURO Special Interest Group on Cutting and Packing* (ESICUP).

A Tabela 3 apresenta a média e desvio padrão das dimensões para as instâncias da literatura.

Instâncias	n	μ_{larg}	μ_{alt}	σ_{larg}	σ_{alt}	
Nice	25	25	22,5	18,8	58,7	15,7
	50	50	15,4	13,8	22,8	10,9
	100	100	10,7	10,1	11,3	9,1
	200	200	7,6	7,2	5,2	4,1
	500	500	4,8	4,6	1,9	1,9
	1000	1000	3,4	3,3	0,8	0,8
	2000	2000	2,4	2,3	0,4	0,4
Path	25	25	32,9	13,9	305,7	115,4
	50	50	24,6	10,7	215,3	67,0
	100	100	16,7	9,3	133,8	58,3
	200	200	12,2	7,3	87,1	36,1
	500	500	8,1	5,5	38,1	23,4
	1000	1000	6,0	4,2	21,1	15,0
	2000	2000	4,3	3,2	11,6	8,2
T series	T1	17	51,7	45,9	866,0	686,8
	T2	25	40,7	39,7	418,9	424,2
	T3	29	38,5	37,8	517,3	377,6
	T4	49	28,8	27,6	237,7	273,0
	T5	73	23,8	22,9	195,2	193,0
	T6	97	19,7	19,9	175,3	193,2
	T7	199	14,1	14,1	71,0	72,1
T2 Kendall	13	32,7	32,0	138,7	153,8	
Jakobs	T2-J1	25	5,2	4,7	3,5	1,4
	T2-J2	50	3,4	3,5	0,8	0,7

Tabela 3- Média e desvio padrão para as instâncias da literatura.

6.3. Resultados

6.3.1. Instâncias reais – roupeiros

Não é conhecido o óptimo das instâncias reais, por isso foi utilizado, como termo de comparação, a altura máxima obtida pela heurística BLFFDH_G. Os resultados da taxa de variação traduzem a variação percentual das heurísticas testadas ($H_{heurística}$) face à heurística BLFFDH_G, sendo calculados utilizando a seguinte expressão:

$$Taxa\ de\ varia\c{c}\tilde{a}\tilde{o}\ (\%) = \left(\frac{H_{heurística}}{H_{BLFFDH_G}} - 1 \right) \times 100,$$

em que:

- $H_{heurística}$ é a altura máxima da heurística testada
- H_{BLFFDH_G} é a altura máxima atingida pela heurística BLFFDH_G.

Estes valores apresentam-se na Tabela 4, com a média da melhor solução a negrito.

		Heurísticas							
Instâncias	n	BLFFDH	FFDH	NFDH	MFFDH	AFFDH	LGF _{iOF}	BL	
SIR	SIR.76	76	0,00	1,38	9,83	1,28	1,38	0,82	0,82
	SIR.245	245	0,00	0,80	18,02	0,31	0,00	0,77	0,87
	SIR.310	310	0,00	0,87	18,82	0,48	0,48	-1,03	0,46
	SIR.618	618	0,00	0,87	14,77	0,76	0,76	0,12	0,39
	SIR.642	642	0,00	0,98	17,05	0,88	0,98	-0,32	0,29
	SIR.1293	1293	0,00	2,09	16,63	1,98	1,99	-1,36	0,26

Tabela 4- Taxa de variação em relação à altura máxima da heurística BLFFDH_G.

Da Tabela 4 verifica-se que não existe diferença nas alturas máximas entre as heurísticas BLFFDH e BLFFDH_G, apesar de uma impor a restrição extra de guilhotina. Por outro lado, e nestas instâncias, estas também são sempre melhores do que as duas heurísticas que as inspiraram, a FFDH e BL.

Se nos cingirmos ao grupo das heurísticas orientadas por níveis (ou seja, a BLFFDH, BLFFDH_G, FFDH, NFDH, MFFDH e AFFDH), constata-se que as duas heurísticas propostas apresentam as melhores soluções, face ao total das instâncias, em 100% dos casos, isto é, seis em seis. Apenas a heurística AFFDH apresenta uma altura máxima igual na instância SIR.245, correspondendo a cerca de 16,67% dos casos.

As heurísticas propostas só originam piores resultados que LGF_{iOF} em três das seis instâncias da Tabela 4, ou seja, em 50% dos casos.

6.3.2. Instâncias da literatura

Para todas as instâncias deste grupo é conhecido o ótimo, que é utilizado como termo de comparação. Os resultados da taxa de variação traduzem, assim, a variação percentual da altura média das instâncias para a heurística testada ($\bar{H}_{heurística}$) face ao valor ótimo da altura dessa instância (H_{opt}), sendo calculados utilizando a seguinte expressão:

$$Taxa\ de\ variação\ (\%) = \left(\frac{\bar{H}_{heurística}}{H_{opt}} - 1 \right) \times 100,$$

com,

$$\bar{H}_{heurística} = \frac{1}{m} \sum_{k=1}^m (H_{heurística})_k$$

E em que,

- $\bar{H}_{heurística}$ - média aritmética da altura máxima das m instâncias para a heurística testada;
- H_{opt} - altura ótima para a instância;
- $(H_{heurística})_k$ - altura máxima da instância k ;
- m - número de instâncias da mesma classe e com o mesmo número de itens.

Na Tabela 5 apresentam-se as taxas de variação, com a média da melhor solução a negrito.

Instâncias	n	Heurísticas								
		BLFFDH	BLFFDH _G	FFDH	NFDH	MFFDH	AFFDH	LGFi _{OF}	BL	
Nice	25	25	21,99	21,99	21,99	29,25	21,99	21,99	16,71	21,46
	50	50	12,63	12,63	15,52	20,54	12,83	12,83	13,40	18,03
	100	100	10,93	10,93	13,99	15,32	12,51	12,51	9,54	16,48
	200	200	7,73	7,73	11,37	13,48	9,79	9,66	8,37	13,54
	500	500	5,63	5,63	8,78	10,05	7,43	7,43	6,69	10,41
	1000	1000	4,01	4,01	6,62	7,13	4,45	4,45	4,50	8,95
	2000	2000	3,53	3,53	5,07	6,08	4,02	4	5,08	7,53
Path	25	25	16,49	19,29	39,15	45,75	20,94	15,96	11,07	23,54
	50	50	16,44	16,44	30,93	43,58	18,16	19,32	11,19	17,00
	100	100	10,27	11,06	30,96	47,76	16,99	15,75	6,85	18,27
	200	200	6,46	7,15	30,39	40,36	18,42	14,03	7,47	24,04
	500	500	5,23	5,73	29,66	35,34	15,09	11,15	5,06	19,67
	1000	1000	4,60	5,14	30,40	34,61	15,25	10,61	4,46	17,54
	2000	2000	3,89	4,30	29,08	31,38	15,64	10,04	3,78	19,59
T series	T1	17	32,00	32,10	46,50	58,70	35,90	30,70	24,50	38,70
	T2	25	19,60	22,90	40,80	49,20	24,20	21,00	15,00	31,90
	T3	29	20,00	23,10	40,80	48,00	26,60	24,60	12,60	31,50
	T4	49	12,00	12,00	27,60	33,30	18,10	16,60	14,00	27,20
	T5	73	10,40	11,80	27,20	33,10	18,00	16,10	9,60	21,00
	T6	97	9,50	8,70	31,70	38,20	17,10	14,60	9,30	24,30
	T7	199	5,60	5,70	27,30	31,20	15,60	13,00	5,10	10,00
T2 Kendall	13	20,00	20,00	50,00	50,00	38,57	38,57	22,86	11,43	
Jakobs	T2-J1	25	26,67	26,67	40,00	40,00	26,67	26,67	20,00	13,33
	T2-J2	50	20,00	20,00	20,00	20,00	20,00	20,00	13,33	13,33

Tabela 5- Taxa de variação em relação à média da altura óptima de cada instância.

Da Tabela 5 verifica-se que nas instâncias do tipo *Nice*, *T2-Kendall*, *T2-J1* e *T2-J2*, não existe diferença nas alturas máximas médias entre a heurística BLFFDH e a BLFFDH_G. Essa diferença é observável nas restantes instâncias, onde a heurística BLFFDH apresenta mais vezes, melhores soluções do que BLFFDH_G. Em termos percentuais, a heurística BLFFDH obtém melhores soluções em cerca de 46% do total de instâncias (11 em 24 instâncias), empatando em 50% (12 em 24) e tendo pior solução em 4% (1 em 24). Contudo, relembra-se que esta heurística não garante um padrão guilhotinável.

A Tabela 6 foi construída de modo a condensar os resultados em três grupos, ajudando na interpretação da Tabela 5. Os valores apresentados indicam a percentagem de vezes que uma determinada heurística apresenta a melhor solução face ao total do tipo de instâncias, com a maior percentagem a negrito.

Melhor solução	Heurísticas							
	BLFFDH	BLFFDH _G	FFDH	NFDH	MFFDH	AFFDH	LGF _{iOF}	BL
C/ padrão guilhotinável	N/A	87,5	8,3	4,2	12,5	25,0	N/A	N/A
Orientada por níveis	91,7	54,2	4,2	4,2	12,5	16,7	N/A	N/A
Todas as heurísticas	29,2	29,2	0,0	0,0	0,0	0,0	58,3	12,5

Tabela 6- Percentagem de vezes que uma heurística apresenta a melhor solução face ao total do tipo de instâncias.

No grupo das heurísticas que garantem um padrão guilhotinável (ou seja, BLFFDH_G, FFDH, NFDH, MFFDH e AFFDH), verifica-se que a heurística BLFFDH_G apresenta a melhor solução em 87,5% dos casos, o que corresponde a 21 em 24 tipos de instâncias. A segunda melhor heurística é a AFFDH com apenas 25% dos casos, representando uma grande diferença face à anterior.

No grupo das heurísticas orientadas por níveis, a BLFFDH apresenta o maior número de melhores soluções, com 22, sendo a BLFFDH_G a segunda melhor com 13.

Só no conjunto de todas as heurísticas é que BLFFDH e BLFFDH_G deixam de proporcionar as melhores soluções, sendo o melhor desempenho obtido com LGF_{iOF}. Esta consegue obter 14 das melhores soluções, o que corresponde a cerca de 58,3% dos casos.

Na Tabela 5 é visível que a qualidade média das soluções melhora, num mesmo grupo de instâncias, quando o número de itens aumenta. Essa melhoria é transversal a todas as heurísticas e, muito provavelmente e em grande parte, deve-se ao facto de as dimensões do objecto, fixa e óptima, se manterem inalteradas, enquanto a dimensão dos itens diminui com o aumento do número de objectos, como se observa na Tabela 3.

Comparando a Tabela 3 com a Tabela 5, verifica-se que, para as heurísticas orientadas por níveis, a BLFFDH e a BLFFDH_G conseguem uma melhor e mais significativa qualidade de empacotamento nas instâncias que apresentam maior desvio padrão, tendo em conta as dimensões médias dos itens, nomeadamente nas instâncias de tipo *Path*.

6.4. Estudo Empírico

6.4.1. Tempos de execução

Para se compararem as heurísticas em termos de tempo de execução, optou-se por executar cada heurística nas instâncias reais e do tipo *Path* e *Nice* durante um tempo fixo (30 minutos). As instâncias da literatura utilizadas foram escolhidas por terem um grande número de itens, como as reais, sendo considerada uma instância da mesma classe (em número de m) para cada número de itens ($n = 25, 50, 100, 200, 500, 1000, 2000$).

Fixou-se um tempo de execução para aumentar a precisão, uma vez que estas heurísticas têm tempos de execução muito baixos, muitas vezes inferiores a 1 milissegundo (ms). Tendo em conta que o contador utilizado na programação tem uma precisão de 1ms, torna-se necessário aumentar a resolução do contador para determinar o tempo que a heurística demora a alocar os itens.

Assim, durante os 30 minutos, foram ininterruptamente feitos vários ciclos de simulação, em maior ou menor número consoante o tipo de heurística, instância e número de itens por alocar. O ciclo de simulação engloba apenas a heurística a ser testada, ou seja, corresponde à determinação de um empacotamento, deixando de fora a leitura, ordenação e escrita de resultados, como mostra a Figura 23.



Figura 23 - Ciclo de simulação

O tempo por cada ciclo de simulação vem expresso em milissegundos e é calculado dividindo o tempo de execução total (30 minutos) pelo número total de ciclos efectuados nesse tempo, como mostra a seguinte expressão:

$$\text{Tempo por ciclo de simulação (ms)} = \frac{\text{Tempo de execução total}_{ms}}{\text{Número total de ciclos}}$$

Os resultados desses cálculos constam na Tabela 7 e Tabela 8:

	n	BLFFDH	BLFFDH_G
SIR	76	1,03	1,31
	245	7,63	7,04
	310	10,27	9,36
	618	44,22	39,71
	642	51,86	43,91
	1293	213,36	172,73

Tabela 7-. Tempo (em milissegundos) por ciclo de simulação em instâncias reais.

	n	BLFFDH		BLFFDH_G	
		Nice	Path	Nice	Path
Wang & Valenzuela (2001)	25	0,22	0,35	0,20	0,31
	50	0,56	1,06	0,48	0,83
	100	1,34	3,39	1,21	2,19
	200	3,78	10,28	3,24	5,54
	500	19,28	40,19	12,30	23,27
	1000	53,65	126,52	41,71	70,72
	2000	147,85	338,91	122,89	210,47

Tabela 8- Tempo (em milissegundos) por ciclo de simulação para instâncias da literatura.

Entre as heurísticas aqui descritas, a BLFFDH_G apresenta, regra geral, os tempos de execução mais rápidos. Tal acontece pela forma como foram implementadas em VBA pelo autor. Enquanto a BLFFDH trata todos os itens como únicos, pesquisando a posição de cada um a fim de criar o PR e determinar o *gap*, a BLFFDH_G pesquisa sub-objectos ao invés de itens. Como os sub-objectos podem conter vários itens, a pesquisa torna-se mais rápida.

Nas instâncias da literatura observa-se notoriamente uma outra diferença, e esta significativa. Trata-se da variação do tempo de execução consoante o tipo de instância, sendo as instâncias *Path* as que exigem mais tempo. A Tabela 3 pode indicar que tal acontece devido a uma maior variação nas dimensões dos itens face às instâncias *Nice*. A isto conjuga-se o tempo de pesquisa para cada PR no espaço que sobra entre dois níveis, mais demorado por serem pesquisados todos os itens ou sub-objectos já alocados no nível, para cada nova posição do PR, e dos itens não alocados para tentar indexar um item a esse PR.

A Figura 24 mostra um padrão de empacotamento utilizando a heurística BLFFDH para uma instância *Nice*, Figura 24(a), e *Path*, Figura 24(b), com 50 itens. A observação destas ilustrações indica que nas instâncias do tipo *Path* existem mais PR no espaço que sobra entre dois níveis, principalmente nos níveis mais baixos.

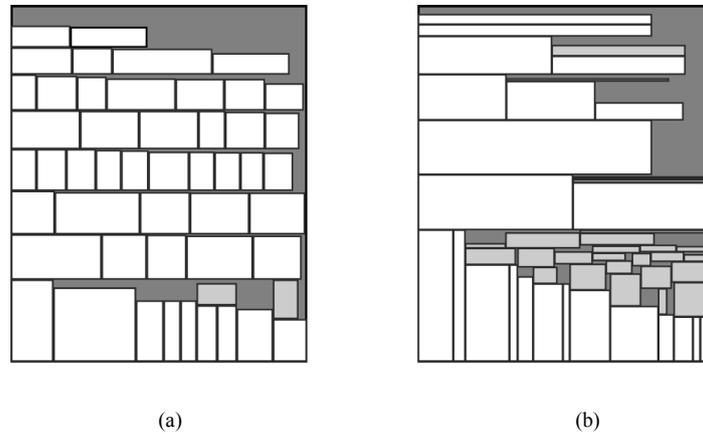


Figura 24- Instância Nice (a) e Path (b)

Esta conjugação parece indicar que, com estas heurísticas, em instâncias que originem muitos PR no espaço que sobra entre dois níveis, e com muita variação de dimensões, a pesquisa de PR é mais intensa, tornando as heurísticas mais lentas.

6.4.2. Complexidade computacional empírica

Os graus de complexidade das heurísticas foram determinados empiricamente, em termos de tempo, através da relação existente entre os tempos do ciclo de simulação de uma determinada heurística e as dimensões das instâncias *S1R*, *Path* e *Nice*, referidas na secção 6.4.1.

Utilizando o número de itens como variável independente, foram feitas regressões dos tempos do ciclo de simulação em função do número de itens, tentando identificar as curvas polinomiais que melhor representam os dados. A correlação entre ambos, para cada heurística e instância, é bastante elevada, como mostra a Tabela 9.

Heurística	Instâncias		
	S1R	Nice	Path
BLFFDH	0,961	0,989	0,989
BLFFDH _G	0,965	0,984	0,984

Tabela 9- Coeficientes de correlação.

A Tabela 10 apresenta os resultados dos coeficientes de determinação (R^2) dos dois tipos de regressão cujo valor era o mais elevado, acima de 0,999.

Heurística	Instância	Tipo de Regressão	
		Quadrática	Cúbica
BLFFDH	S1R	0,99974	0,99974
	Nice	0,99955	0,99991
	Path	0,99903	0,99999
BLFFDH _G	S1R	0,9999	0,99993
	Nice	0,99937	0,99997
	Path	0,99977	1,00000

Tabela 10- Resultados de R^2 para as regressões quadrática e cúbicas.

As formas de regressão que melhor encaixaram foram a quadrática e a cúbica. As exponenciais, logarítmica ou de potências, apresentaram sempre piores valores de R^2 . Com os coeficientes de correlação e determinação próximos da unidade, é possível inferir acerca do grau de complexidade empírica das heurísticas programadas.

Dos tipos de regressão, é a cúbica que apresenta os coeficientes de determinação mais elevados, encaixando-se melhor nos dados. Isto indica que a complexidade empírica das duas heurísticas programadas é $O(n^3)$.

7. CONCLUSÕES E TRABALHOS FUTUROS

7.1. Conclusões

O objectivo deste trabalho era determinar soluções admissíveis para o problema de corte bidimensional. Foram propostas duas novas heurísticas, a *Guillotinable Bottom-Left First Fit Decreasing Height* (BLFFDH_G) e a *Bottom-Left First Fit Decreasing Height* (BLFFDH), para alocar itens rectangulares a objectos também rectangulares, para o caso bidimensional, 2DRSBSBPP. Nenhum dos itens pode ser rodado ou sobreposto. Ambas as heurísticas são do tipo *bin packing*, orientadas por níveis e destinam-se a problemas do tipo *offline*. A diferença entre as novas heurísticas reside no facto de uma impor o corte guilhotina, a BLFFDH_G.

Os novos métodos foram comparados entre si e com outras heurísticas da literatura, nomeadamente a FFDH, a NFDH, a MFFDH, a AFFDH, a LGF_{IOF} e a BL, utilizando dois tipos de instâncias: da literatura e reais de corte de roupeiros.

Foram obtidos resultados promissores para as heurísticas propostas, mostrando também que as heurísticas orientadas por níveis conseguem ser relativamente competitivas nos testes realizados neste trabalho, face a outras como a LGF_{iOF} e a BL.

As duas heurísticas propostas conseguem nas instâncias reais e face a todos os algoritmos testados, obter o melhor padrão em 50% dos casos. Nas instâncias da literatura, e relativamente à altura média máxima, o melhor padrão de empacotamento é obtido em 29,2% dos casos em ambas as heurísticas propostas. Das heurísticas orientadas por níveis, nas instâncias reais, a BLFFDH_G e a BLFFDH apresentam as melhores soluções em 100% dos casos. Já nas instâncias da literatura, e relativamente à altura média máxima, a BLFFDH proporcionou melhores soluções em 91,7% das instâncias e a BLFFDH_G em 54,2%.

A BLFFDH_G e a BLFFDH combinam as heurísticas FFDH e BL e, em praticamente todos os testes, proporcionam melhores soluções que as que lhes deram origem. Em termos de tempo de execução, a BLFFDH_G revelou-se mais rápida que a BLFFDH, embora a complexidade empírica da programação seja, para ambas, igual a $O(n^3)$.

7.2. Propostas para Trabalhos Futuros

Durante a elaboração deste trabalho final de mestrado foi ficando patente a possibilidade de realização de futuros trabalhos relacionados com as heurísticas propostas, que se sumarizam de seguida:

- Descobrir as limitações das heurísticas propostas - geralmente os testes indicam o quão boas são as heurísticas mas não dizem onde começam a falhar. Note-se que a qualidade das soluções finais depende do tipo de problema, ou seja, uma heurística pode ser boa para um tipo de problema e em outro obter soluções de fraca qualidade;
- Adaptar as heurísticas propostas para rotação ortogonal - ajustar as heurísticas ao caso em que é permitida a rotação ortogonal de todos ou só de alguns itens;
- Flexibilizar a heurística BLFFDH_G - permitir que em determinadas situações, e sem pôr em causa o padrão guilhotinável, os sub-objects possam ser justificados o mais à esquerda possível. Isto permitiria, em certas instâncias, um melhor aproveitamento do espaço que sobra entre dois níveis consecutivos.

8. BIBLIOGRAFIA

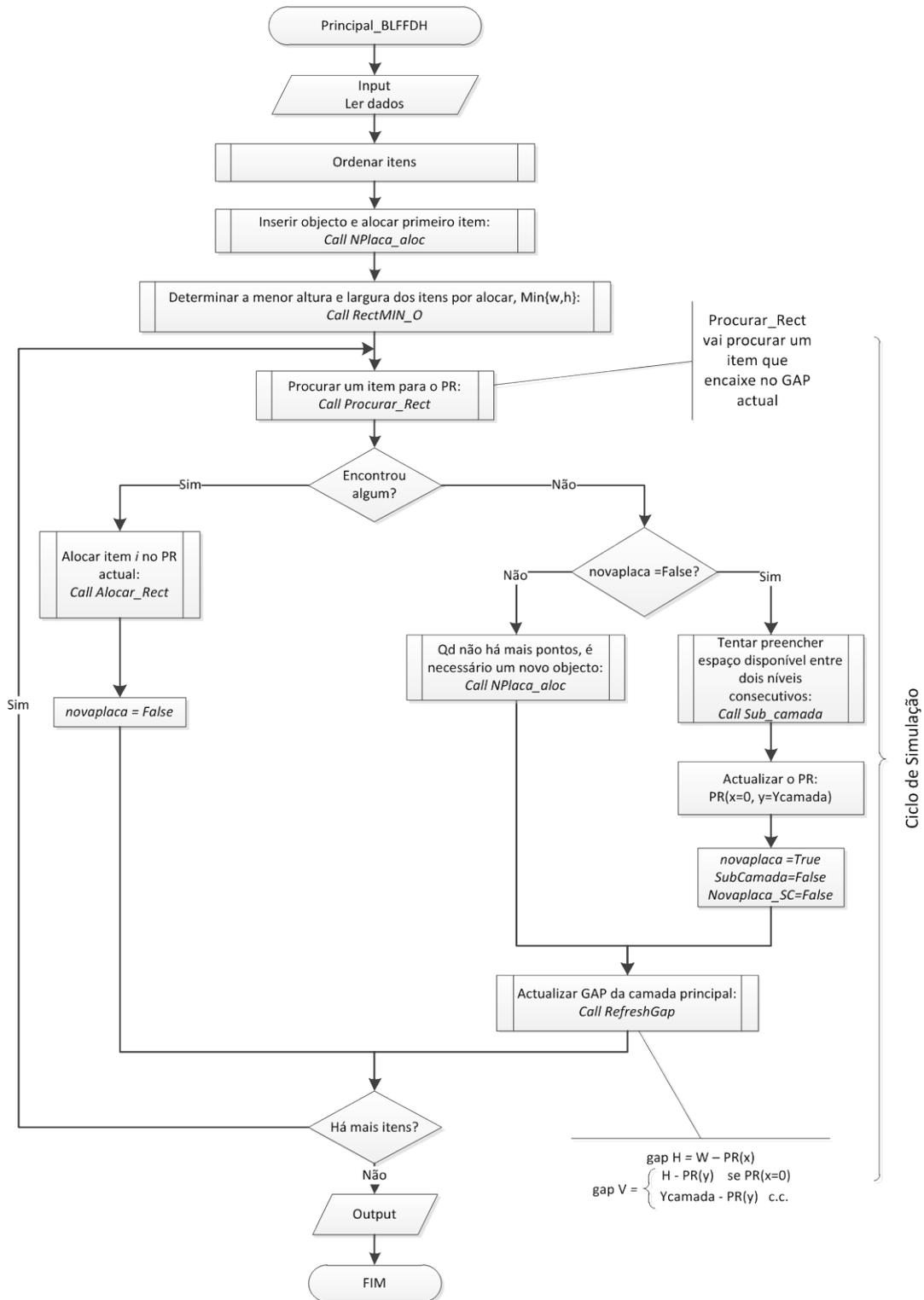
- Amossen, R., & Pisinger, D. (Fevereiro de 2010). *Multi-dimensional bin packing problems with guillotine constraints*. Obtido em 1 de Abril de 2011, de Rasmus Resen Amossen: http://rasmus.resen.org/pub/ap10_opp_csp.pdf
- Araya, I., Neveu, B., & Riff, M. C. (11 de Setembro de 2007). *An efficient hyperheuristic for strip-packing problems*. Obtido em 1 de Março de 2011, de <ftp://ftp-sop.inria.fr/coprin/neveu/articles/iaraya-HSPpaper.pdf>
- Baker, B., Coffman, J. E., & Rivest, R. (1980). Orthogonal packings in two dimensions. *SIAM J. Comput.*, 9 (4), 846-855.
- Berkey, J., & Wang, P. (Maio de 1987). Two-dimensional finite bin-packing algorithms. *The Journal of the Operational Research Society*, 38 (5), 423-429.
- Bortfeldt, A. (2006). A genetic algorithm for the two-dimensional strip packing problem with rectangular pieces. *European Journal of Operational Research*, 172, 814-837.
- Burke, E., & Kendall, G. (1999). *Listing Gallery: Data Sets 2D - Rectangular*. Obtido em 3 de Agosto de 2011, de EURO Special Interest Group on Cutting and Packing: http://paginas.fe.up.pt/~esicup/tiki-download_file.php?fileId=137
- Chazelle, B. (1983). The bottom-left bin-packing heuristic: an efficient implementation. *IEEE Transactions on Computers*, 32 (8), 697-707.
- Coffman, J. E., & Shor, P. W. (1990). Average-case analysis of cutting and packing in two dimensions. *European Journal of Operational Research*, 44, 134-144.
- Coffman, J. E., Garey, D. S., & Tarjan, R. E. (1980). Performance bounds for level oriented two-dimensional packing algorithms. *SIAM Journal on Computing*, 9 (4), 808-826.
- Dolatabadi, M., Lodi, A., & Monaci, M. (2010). Exact algorithms for the two-dimensional guillotine Knapsack. *Computers & Operations Research*, Accepted manuscript.

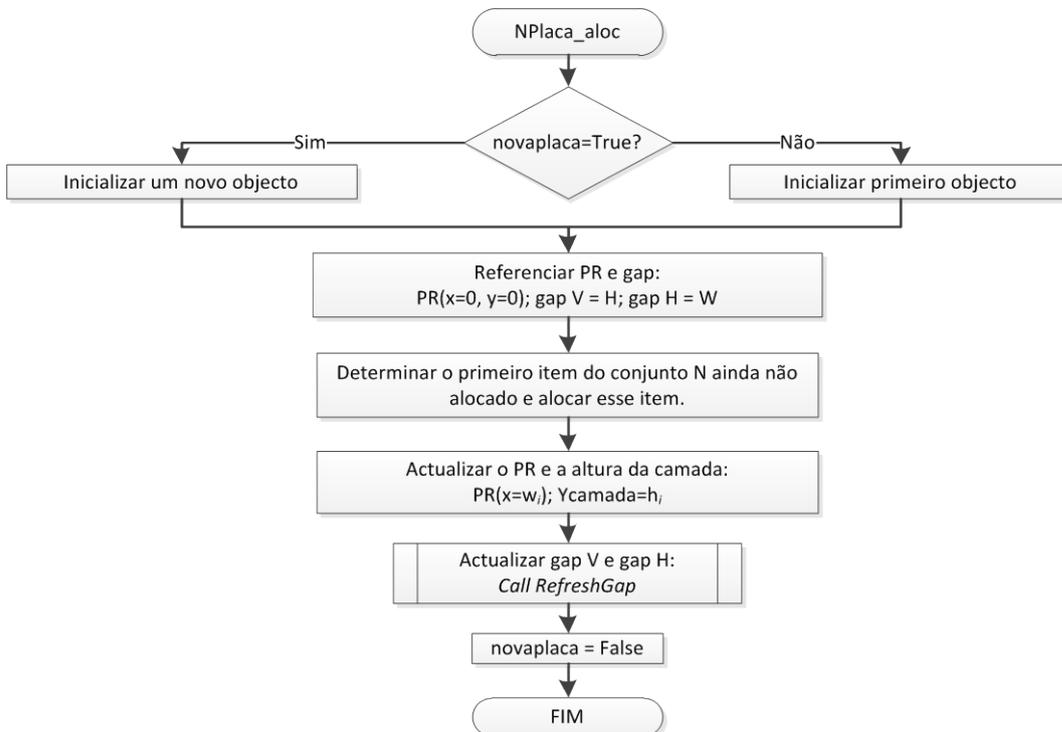
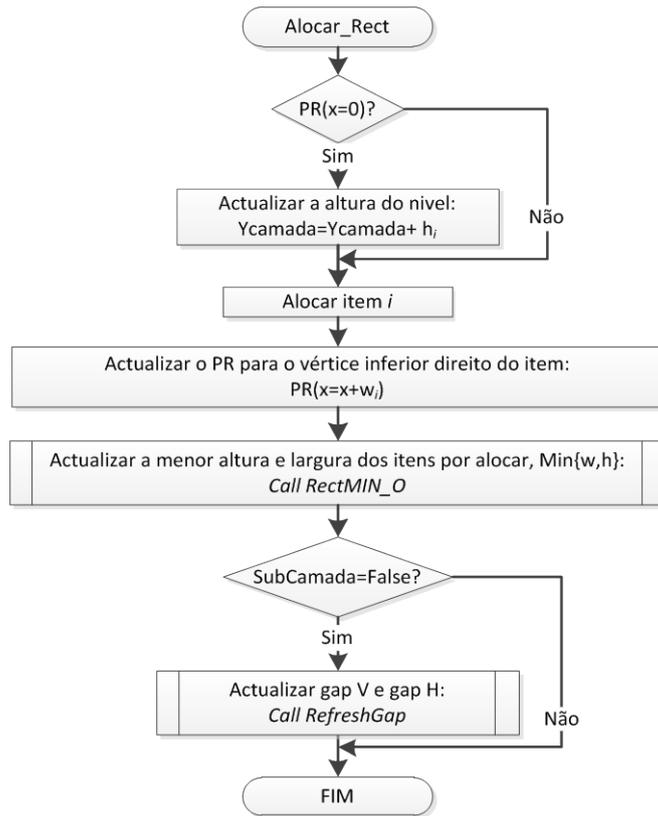
- Halvorson, M. (2010). *Microsoft Visual Basic 2010 Step by Step*. Redmond: Microsoft Press.
- Hopper, E. (2000). *Listing Gallery: Data Sets 2D - Rectangular*. Obtido em 3 de Agosto de 2011, de EURO Special Interest Group on Cutting and Packing: http://paginas.fe.up.pt/~esicup/tiki-download_file.php?fileId=131
- Hopper, E. (2000). *Two-dimensional packing utilising evolutionary algorithms and other meta-heuristic methods*. A Thesis submitted to the University of Wales for the Degree of Doctor of Philosophy, University of Wales, Cardiff, School of Engineering.
- Jakobs, S. (1996). *Listing Gallery: Data Sets 2D - Rectangular*. Obtido em 3 de Agosto de 2011, de EURO Special Interest Group on Cutting and Packing: http://paginas.fe.up.pt/~esicup/tiki-download_file.php?fileId=138
- Lodi, A., Martello, S., & Vigo, D. (Abril de 1999). Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. *INFORMS Journal on Computing, Volume 11*, Páginas 345 - 357.
- Lodi, A., Martello, S., & Vigo, D. (2002). Recent advances on two-dimensional bin packing problems. *Discrete Applied Mathematics, 123*, 379-396.
- Ntene, N., & Vuuren, J. H. (27 de Janeiro de 2007). *A survey and comparison of heuristics for the 2D oriented on-line strip packing problem*. Obtido em 1 de Março de 2011, de Jan H van Vuuren Research Publications: <http://dip.sun.ac.za/~vuuren/papers/onlinestrip.pdf>
- Ortmann, F. G. (2010). *Heuristics for offline rectangular packing problems*. Dissertation presented for the degree PhD (Operations Research), Stellenbosch University, Department of Logistics.
- Ortmann, F. G., Ntene, N., & Vuuren, J. H. (2010). New and improved level heuristics for the rectangular strip packing and variable-sized bin packing problems. *European Journal of Operational Research, 203*, 306-315.

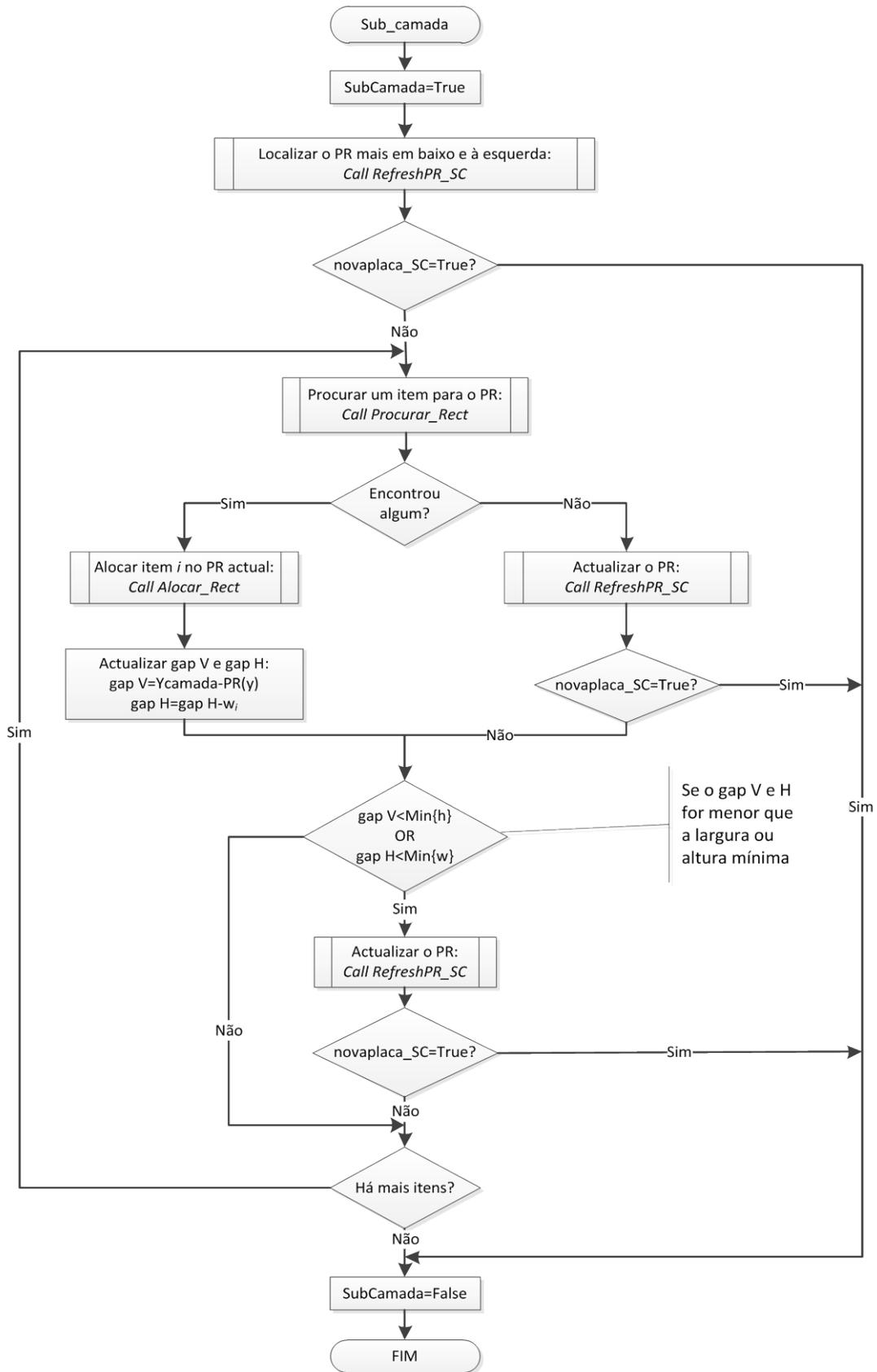
- Polyakovskiy, S., & Hallah, R. (2009). An agent-based approach to the two-dimensional guillotine bin packing problem. *European Journal of Operational Research*, 192, 767-781.
- Rode, M., & Rosenberg, O. (1987). An analysis of heuristic trim-loss algorithms. *Engineering Costs and Production Economics*, 12, 71-78.
- Silva, H. J. (2003). *Heurísticas orientadas à camada para problemas de empacotamento rectangular*. Tese de Mestrado, Escola de Gestão do Porto - Universidade do Porto.
- Valenzuela, C., & Wang, P. (2011). Heuristics for large strip packing problems with guillotine patterns: an empirical study. *4th Metaheuristics International Conference*, (pp. 417-421). Porto, Portugal.
- Vuuren, J. H. (13 de Fevereiro de 2006). *Mumford-Valenzuela benchmark data (2001)*. Obtido em 5 de Abril de 2011, de Web site de Applied Mathematics: [http://dip.sun.ac.za/~vuuren/repositories/levelpaper/MumfordData\[1\].htm](http://dip.sun.ac.za/~vuuren/repositories/levelpaper/MumfordData[1].htm)
- Wang, P., & Valenzuela, C. (2001). Data set generation for rectangular placement problems. *European Journal of Operational Research*, 134(2), 378-391.
- Wäscher, G., Haussner, H., & Schumann, H. (18 de Março de 2004). Obtido em 31 de Agosto de 2011, de EURO Special Interest Group on Cutting and Packing: http://paginas.fe.up.pt/~esicup/tiki-download_file.php?fileId=57
- Wäscher, G., Haussner, H., & Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183, 1109-1130.
- Wong, L., & Lee, L. (2009). Heuristic placement routines for two-dimensional bin packing problem. *Journal of Mathematics and Statistics* 5 (4), Páginas 334-341.

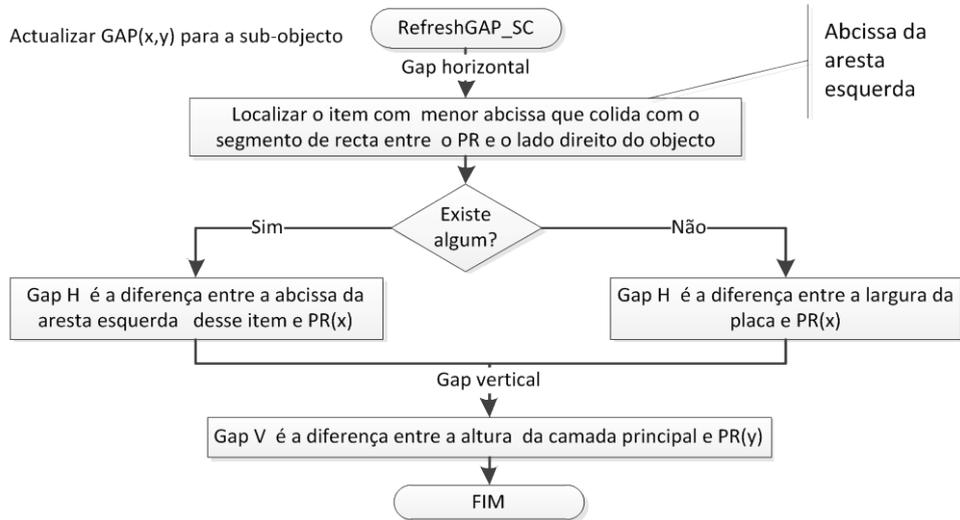
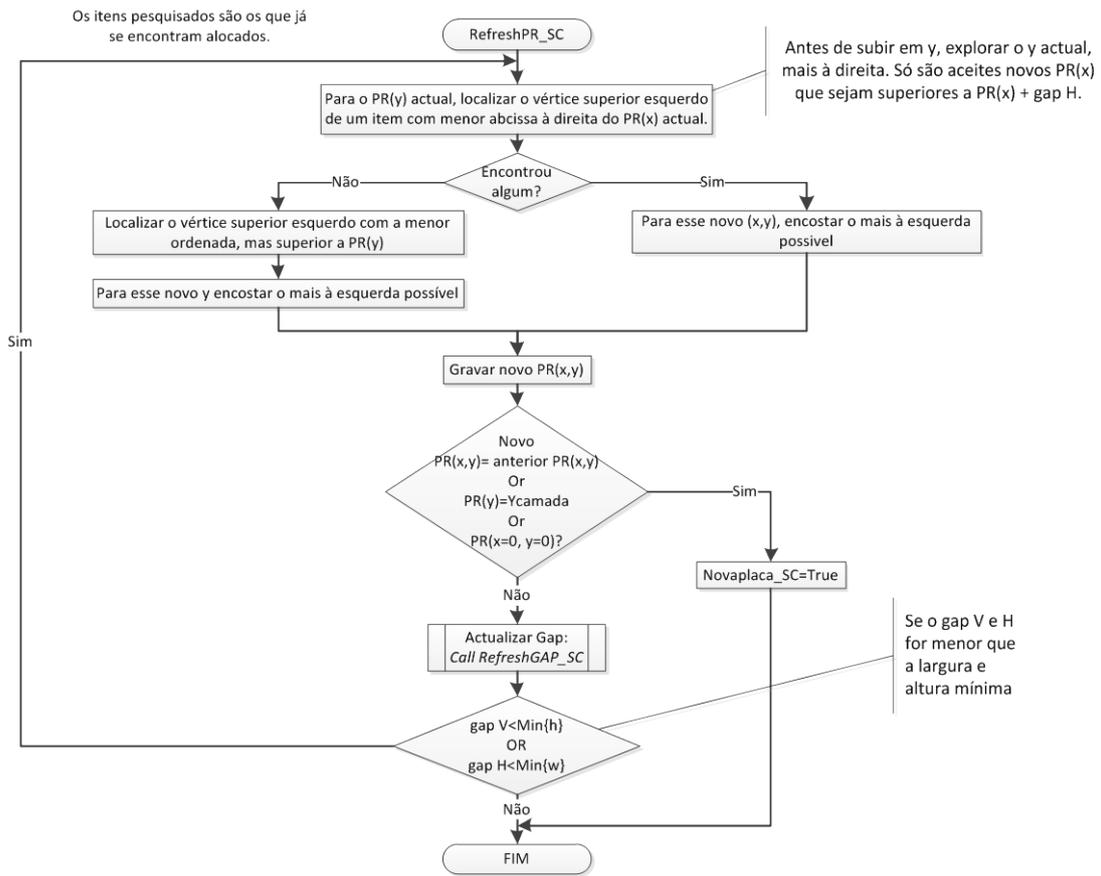
9. ANEXOS

ANEXO 1 - FLUXOGRAMAS DA HEURÍSTICA BLFFDH

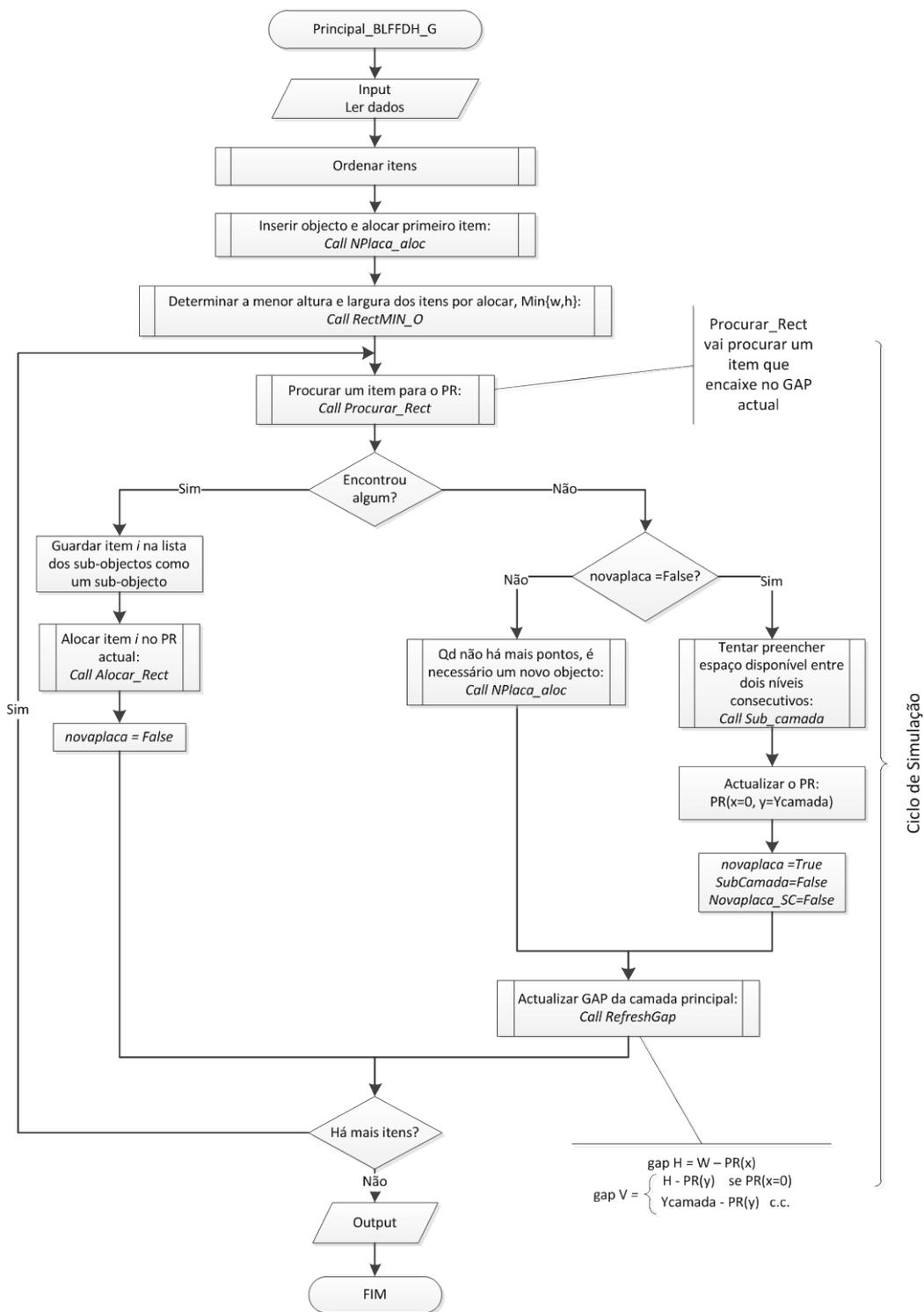


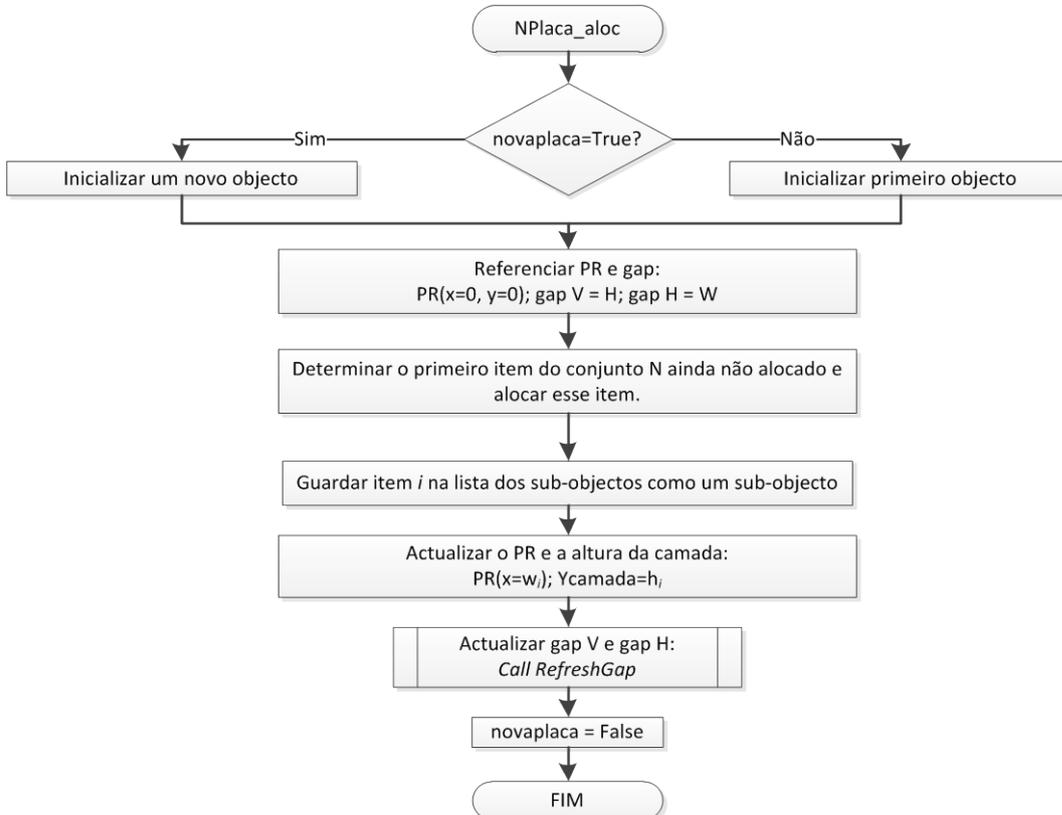
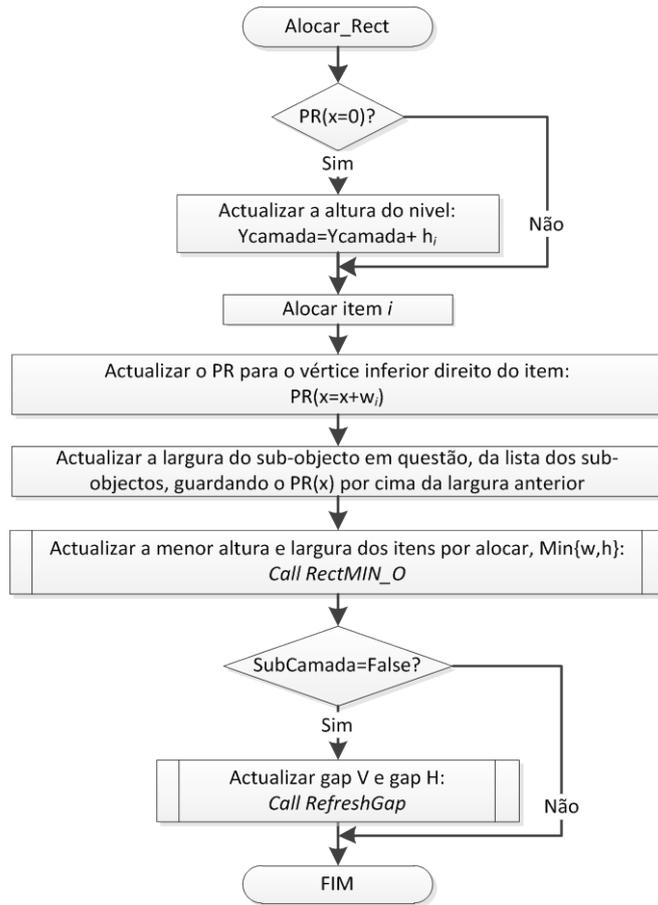


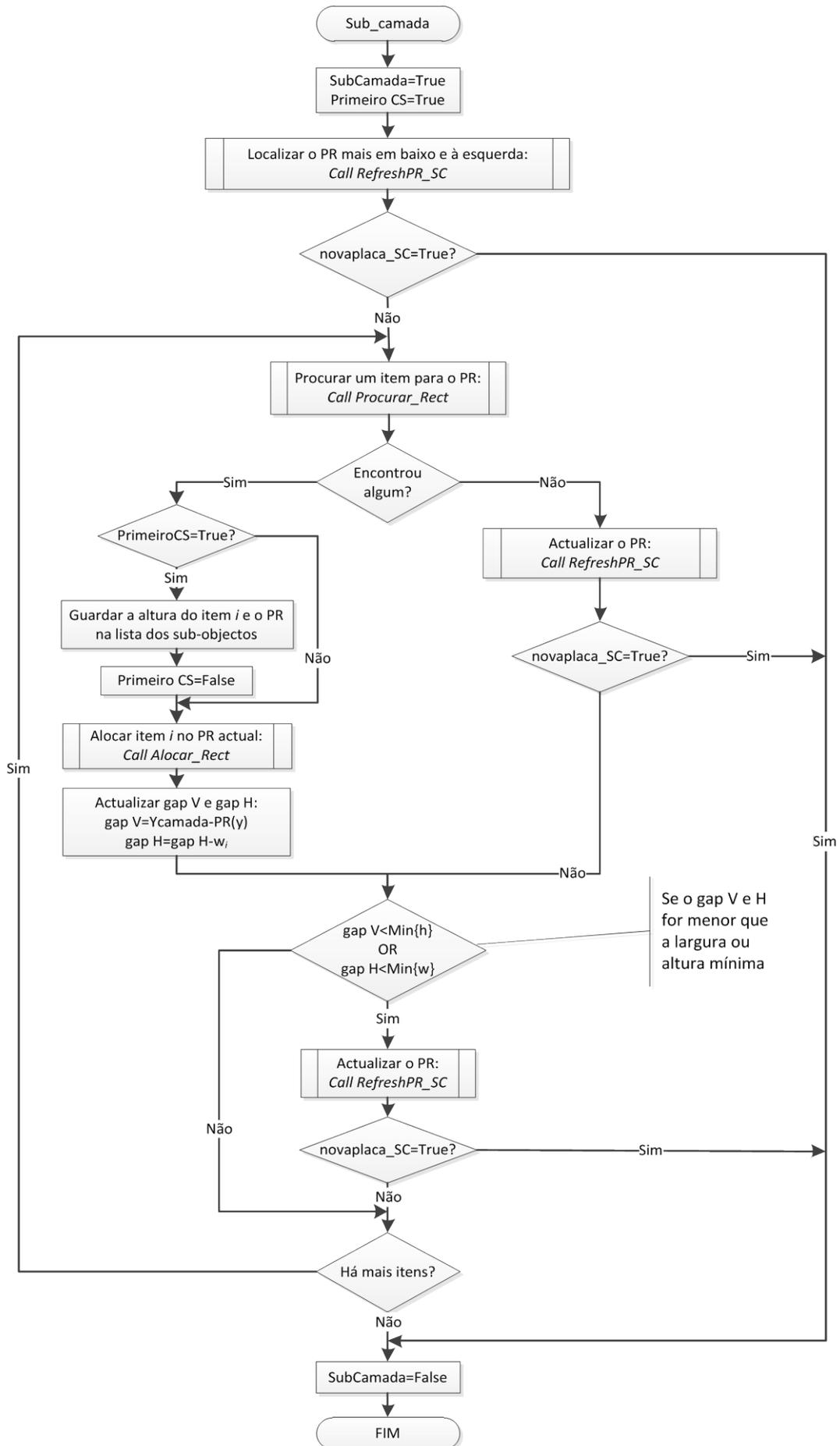




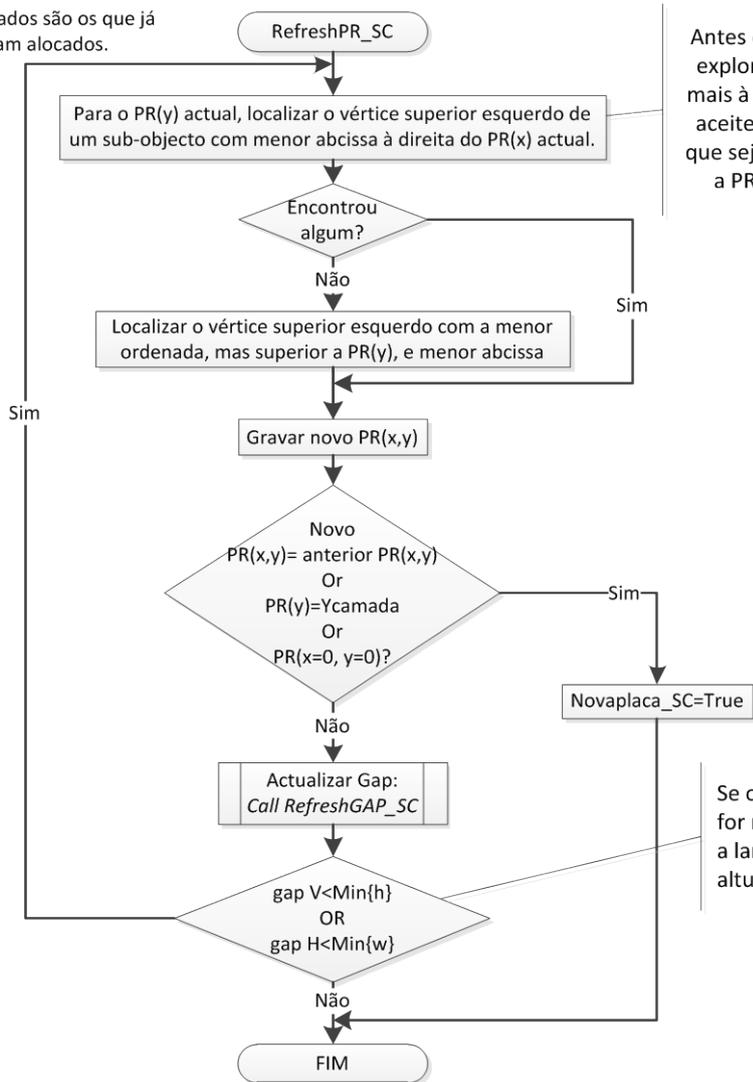
ANEXO 2 - FLUXOGRAMAS DA HEURÍSTICA BLFFDH_G







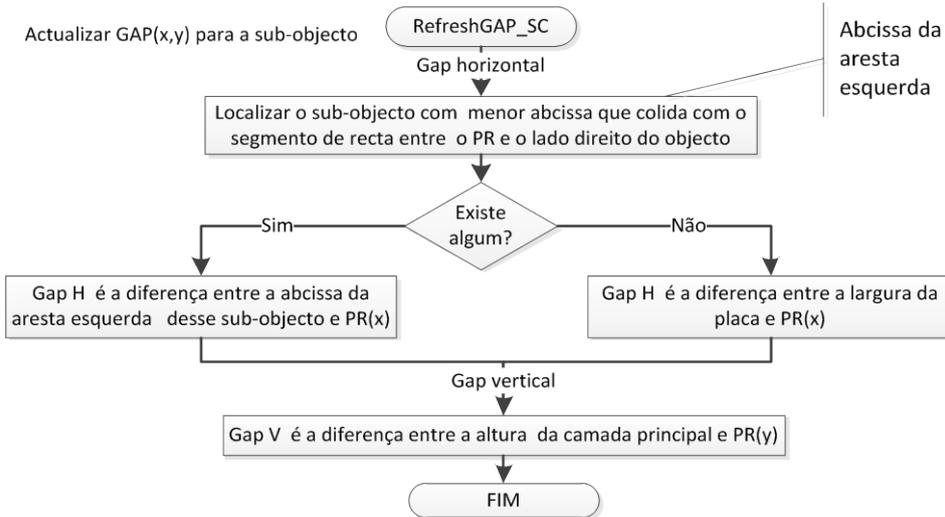
Os itens pesquisados são os que já se encontram alocados.



Antes de subir em y, explorar o y actual, mais à direita. Só são aceites novos PR(x) que sejam superiores a PR(x) + gap H.

Se o gap V e H for menor que a largura e altura mínima

Actualizar GAP(x,y) para a sub-objecto



Abcissa da aresta esquerda

ANEXO 3 - COMPARAÇÃO DAS INSTÂNCIAS DA LITERATURA COM METODOLOGIA DAS INSTÂNCIAS PRÁTICAS

Será utilizada a metodologia de comparação das instâncias reais para as instâncias da literatura. Na fórmula do ponto 6.3.2 ao invés de se utilizar H_{opt} utiliza-se \bar{H}_{BLFFDH_G} . A tabela seguinte apresenta os resultados, em percentagem, com base nessa nova fórmula.

		Heurísticas							
Instâncias	n	BLFFDH	FFDH	NFDH	MFFDH	AFFDH	LGF _{OF}	BL	
Nice	25	25	0,00	0,00	5,95	0,00	0,00	-4,33	-0,44
	50	50	0,00	2,57	7,02	0,18	0,18	0,69	4,80
	100	100	0,00	2,75	3,95	1,42	1,42	-1,26	5,00
	200	200	0,00	3,37	5,33	1,91	1,79	0,59	5,39
	500	500	0,00	2,98	4,19	1,70	1,70	1,01	4,53
	1000	1000	0,00	2,50	3,00	0,42	0,42	0,47	4,75
	2000	2000	0,00	1,49	2,46	0,47	0,46	1,50	3,86
Path	25	25	-2,35	16,65	22,18	1,38	-2,79	-6,89	3,57
	50	50	0,00	12,44	23,30	1,48	2,48	-4,51	0,48
	100	100	-0,71	17,91	33,04	5,33	4,22	-3,80	6,49
	200	200	-0,65	21,69	30,99	10,52	6,42	0,30	15,76
	500	500	-0,48	22,63	28,00	8,84	5,13	-0,64	13,18
	1000	1000	-0,51	24,03	28,03	9,63	5,20	-0,64	11,80
	2000	2000	-0,39	23,75	25,96	10,87	5,50	-0,50	14,66
T series	T1	17	-0,08	10,90	20,14	2,88	-1,06	-5,75	5,00
	T2	25	-2,69	14,56	21,40	1,06	-1,55	-6,43	7,32
	T3	29	-2,52	14,38	20,23	2,84	1,22	-8,53	6,82
	T4	49	0,00	13,93	19,02	5,45	4,11	1,79	13,57
	T5	73	-1,25	13,77	19,05	5,55	3,85	-1,97	8,23
	T6	97	0,74	21,16	27,14	7,73	5,43	0,55	14,35
	T7	199	-0,09	20,44	24,12	9,37	6,91	-0,57	4,07
T2 Kendall	13	0,00	25,00	25,00	15,48	15,48	2,38	-7,14	
Jakobs	T2-J1	25	0,00	10,53	10,53	0,00	0,00	-5,26	-10,53
	T2-J2	50	0,00	0,00	0,00	0,00	0,00	-5,56	-5,56

Taxa de variação (%) em relação à altura máxima da heurística BLFFDH_G.