



Análise de formulações matemáticas para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios

Vanessa Munhoz Reina Bezerra

Instituto de Ciências Matemáticas e de Computação - USP
Av. Trabalhador São-carlense, 400, Centro, 13566-590, São Carlos - SP, Brasil
vanessaicmc@gmail.com

Aline Aparecida de Souza Leão

Universidade Federal Tecnológica do Paraná - UTFPR
Av. Alberto Carazzai, 1640, Centro, 86300-000, Cornélio Procópio - PR, Brasil
aasleao@gmail.com

José Fernando Oliveira

INESCTEC, Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-590, Porto, Portugal
jfo@fe.up.pt

Maristela Oliveira dos Santos

Instituto de Ciências Matemáticas e de Computação - USP
Av. Trabalhador São-carlense, 400, Centro, 13566-590, São Carlos - SP, Brasil
mari@icmc.usp.br

Resumo

O problema de empacotamento em faixas bidimensional guilhotinado 2-estágios tem recebido pouca atenção por parte da comunidade científica. Atualmente, o modelo mais competitivo para este problema, até onde sabemos, é o proposto por Lodi et al., onde é acrescentado ao problema a restrição de que os itens devem ser alocados formando níveis. Recentemente Furini et al. introduziram um novo modelo matemático que, embora não seja especificamente para este problema, é um modelo eficiente para problemas de empacotamento bidimensional gerais. Além disso, o modelo de Furini et al. pode ser adaptado para problemas de empacotamento 2-estágios. Neste artigo, propomos uma adaptação deste modelo para abordar especificamente o empacotamento em faixas bidimensional guilhotinado 2-estágios. Mais ainda, comparamos o desempenho computacional de ambos os modelos usando instâncias clássicas da literatura.

PALAVRAS CHAVE. Corte e Empacotamento, Corte Guilhotinado, Programação Inteira.
Tópicos. **PM - Programação Matemática.**

Abstract

The two dimensional two-staged guillotine strip packing problem has received little attention in the literature. The most competitive model, as far as the authors know, is the one proposed by Lodi et al., where the items are packed in levels. Recently Furini et al. introduced a new mathematical model, despite not addressing specifically this problem, it is efficient for general two-dimensional packing problems. In addition, the model of Furini et al. can be adapted for two-staged packing problems. In this paper, we propose an adaptation of this model to specifically address the two dimensional two-staged guillotine strip packing. We also compare the computational performance of both models using classical instances of the literature.

KEYWORDS. Cutting and Packing, Guillotine Cutting, Integer Programming.
Paper topics. **PM - Mathematical Programming.**



1. Introdução

Modelos matemáticos para o estudo de problemas de cortes e empacotamentos tiveram início em 1939 com o matemático e economista soviético Leonid Vitaliyevich Kantorovich (ver Kantorovich [1960]) e ganharam destaque durante a década de 60 com os trabalhos Gilmore e Gomory [1961], Gilmore e Gomory [1963] e Gilmore e Gomory [1965]. Tais problemas aparecem com muita frequência na indústria e no comércio.

Em várias aplicações industriais é requerido alocar um conjunto de itens retangulares em unidades retangulares padronizadas maiores (objetos) com o objetivo de minimizar o desperdício de matéria-prima. Por exemplo, nas indústrias de papel ou tecido, os componentes retangulares (itens) precisam ser cortados a partir de um rolo de material e o objetivo é obter todos os itens utilizando o comprimento mínimo do rolo. Estes problemas de otimização são conhecidos na literatura como: problemas de empacotamento bidimensional em faixas. A maioria das contribuições da literatura são dedicadas ao caso em que os itens a serem empacotados possuem orientação fixa com respeito à unidade em estoque (faixa ou *strip*). Além disso, ao problema de empacotamento em faixas são acrescentadas restrições relativas ao tipo de corte (guilhotinado ou não), limitação do número de cortes (estágios), entre outras. Tais restrições são impostas pelas características dos equipamentos utilizados no processo de corte. Mais ainda, em Lodi [1999], Lodi et al. [1999], Lodi et al. [2002] e Lodi et al. [2004] é acrescentada ao problema a restrição de que os itens devem ser alocados formando níveis (ou prateleiras). Neste trabalho, consideraremos que os itens são alocados de forma ortogonal no objeto e possuem orientação fixa, o corte é do tipo guilhotinado 2-estágios e os itens são alocados formando níveis.

Devido a complexidade do problema, as principais pesquisas são baseadas em métodos heurísticos e existem poucos trabalhos na literatura que apresentam modelos matemáticos. Portanto, neste trabalho temos como objetivo apresentar a formulação matemática MF para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios em níveis. A formulação foi obtida por meio de uma alteração no modelo apresentado em Furini et al. [2016] para problemas de corte bidimensional, no qual os autores se concentraram no problema da mochila bidimensional e fizeram uma breve discussão sobre a extensão da modelagem para o problema de corte de estoque bidimensional e para o problema de empacotamento em faixas. Neste artigo não foi apresentado testes computacionais exaustivos para o problema de empacotamento em faixas, os autores analisaram instâncias com no máximo 5 itens e não comparam o desempenho da modelagem com modelos da literatura. No presente trabalho, realizamos experimentos computacionais utilizando-se uma classe com 50 instâncias da literatura com a finalidade de analisar e comparar o desempenho dos modelos MF e Lodi et al. [2004].

Na Seção 2, descrevemos o modelo matemático introduzido em Lodi et al. [2004] (esta modelagem possui um número polinomial de variáveis e restrições.) e apresentamos a formulação matemática MF para o problema. A Seção 3 apresenta a relaxação linear dos modelos descritos na Seção 2, bem como os resultados computacionais para 50 instâncias da literatura. Para finalizar, introduzimos algumas observações na Seção 4.

2. O problema de empacotamento em faixas

Definição 1. Dados um conjunto de n itens retangulares com dimensões (w_i, h_i) , onde w_i representa a largura e h_i a altura de cada item, para $i = 1, \dots, n$, e um objeto retangular (faixa) com largura fixa W e altura H grande o suficiente ("altura infinita") para alocar todos os itens. O objetivo do problema de empacotamento em faixas é alocar, sem sobreposição, todos os itens na faixa, minimizando a altura utilizada.

Na tipologia dada por Wascher et al. [2007], o problema apresentado acima pode ser classificado como: 2D - (*Rectangular*) *Open Dimensional Problem* - ODP: problema de corte bidimensional (retangular) com uma dimensão aberta.



O problema de empacotamento em faixas bidimensional com os itens apresentando orientação fixa e considerando que o corte é do tipo guilhotinado feito em 2-estágios é uma versão do problema de empacotamento em faixas que tem recebido pouca atenção por parte da comunidade científica. Uma formulação matemática com um número polinomial de variáveis e restrições foi proposta por Lodi et al. [2004] a qual ainda é bastante competitiva.

A seguir, na Seção 2.1 descreveremos o modelo matemático apresentado em Lodi et al. [2004] e, em seguida, na Seção 2.2 introduzimos a formulação matemática MF.

2.1 Formulação Matemática proposta por Lodi et al. (2004)

Nesta seção, descreveremos o modelo matemático para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios proposto por Lodi et al. [2004]. Neste modelo, os itens são alocados formando níveis. O primeiro nível é o fundo da faixa (ou *strip*), e os itens são alocados com sua base nela. Os itens começam a ser inseridos neste primeiro nível e a altura dele é determinada pela altura do item mais alto. Deste modo, o próximo nível se inicia na linha horizontal desenhada no topo do item mais alto do nível anterior, e assim por diante (ver Figura (b)).

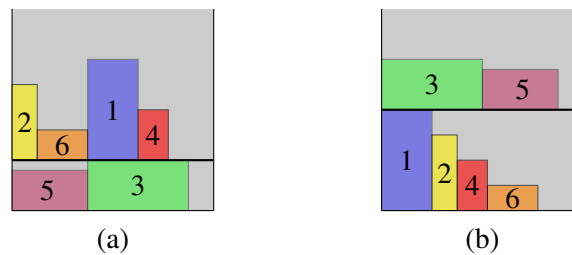


Figura 1: (a) Empacotamento em níveis; (b) empacotamento em nível normalizado (representa o empacotamento utilizado em Lodi et al. [2004], no qual os itens de cada nível são ordenados de forma decrescente pela altura) - adaptada de Lodi et al. [2002].

Para a modelagem matemática do problema, as considerações são: em cada nível, o primeiro item (mais à esquerda) possui a maior altura; o primeiro nível (mais baixo) é o nível mais alto do objeto; e, os itens são ordenados e renumerados de forma decrescente em relação à altura, isto é, $h_1 \geq h_2 \geq \dots h_n$.

As variáveis de decisão são:

$$\bullet y_i = \begin{cases} 1, & \text{se o item } i \text{ inicializa o nível } i; \\ 0, & \text{caso contrário.} \end{cases} \quad (i = 1, \dots, n)$$

$$\bullet x_{ij} = \begin{cases} 1, & \text{se o item } j \text{ está alocado no nível } i; \\ 0, & \text{caso contrário.} \end{cases} \quad (i = 1, \dots, n-1, j > i)$$

A formulação matemática proposta por Lodi et al. [2004] para o problema é:



$$\text{Min} \sum_{i=1}^n h_i y_i \quad (1)$$

$$\text{sujeito a :} \sum_{i=1}^{j-1} x_{ij} + y_j = 1, \quad j = 1, \dots, n \quad (2)$$

$$\sum_{j=i+1}^n w_j x_{ij} \leq (W - w_i) y_i, \quad i = 1, \dots, n - 1 \quad (3)$$

$$y_i \in \{0, 1\}, \quad i = 1, \dots, n \quad (4)$$

$$x_{ij} \in \{0, 1\}, \quad i = 1, \dots, n - 1, \quad j > i \quad (5)$$

A função objetivo (1) deste problema tem, como critério de otimização, a minimização da altura utilizada da faixa (ou *strip*). As equações (2) garantem que cada item será alocado exatamente uma vez (inicializando um nível ou em um nível inicializado por um item precedente (mais alto)). As restrições (3) impõem que, em cada nível, a largura do objeto (*strip*) não será ultrapassada. E, as expressões (4) e (5) definem o tipo das variáveis do problema.

Exemplo 2. Considere uma instância com objeto inicial (faixa) de largura fixa $W = 10$ e três tipos de itens, indexados por j , com as alturas e larguras apresentadas na Tabela 1.

Tabela 1: Dados relativo aos itens.

Tipo de item (j)	Altura (h_j)	Largura (w_j)
1	9	5
2	7	5
3	6	10

A solução ótima deste problema é 15 e o layout desta solução pode ser visto na Figura 2, na qual os traços em negrito representam os níveis e a parte hachurada denota o desperdício.

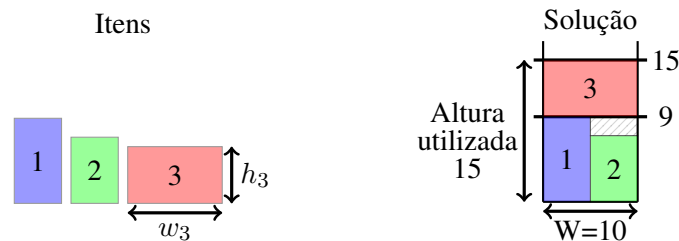


Figura 2: Solução do exemplo utilizando o modelo matemático Lodi et al. (2004).

2.2 Formulação Matemática MF

Nesta seção, o modelo apresentado em Furini et al. [2016] para problemas de corte bi-dimensional guilhotinado é adaptado para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios. Neste artigo, os autores se concentraram no problema da mochila bidimensional e fizeram uma breve discussão sobre a extensão da modelagem para o problema de corte de estoque bidimensional e para o problema de empacotamento em faixas.

A ideia principal da formulação matemática MF é semelhante ao que foi apresentado em Furini et al. [2016]. A partir de uma placa retangular inicial, obtemos placas menores por meio de



um corte guilhotinado horizontal (1º estágio do corte); para cada placa obtida, precisamos decidir se realizamos novos cortes ou mantemos a placa; mantemos a placa quando suas dimensões são iguais às dimensões de um dos itens. O processo é iterativo até que a altura das placas não sejam suficientemente grandes para caber algum item ($H_k \leq h_j$). De maneira análoga, realizamos um corte na vertical (2º estágio do corte) e obtemos duas placas menores, novamente é necessário decidir se realizaremos outro corte ou armazenaremos a placa ($W_k \leq w_j$), e assim sucessivamente até que não exista mais placas que caibam itens. Diferentemente do que é feito em Furini et al. [2016], que realiza os cortes na horizontal e vertical simultaneamente, pois o problema possui k -estágios, aqui primeiro realizamos os cortes na horizontal (gerando todos níveis) e, depois na vertical.

Assim como em Furini et al. [2016], em MF cada decisão de corte está representada por uma tripla (p, k, o) , na qual a posição p indica a distância a partir do canto inferior esquerdo de uma placa k , onde um corte com a orientação o é executado. Na Figura 3 (a) temos um exemplo de corte horizontal realizado na posição p em uma placa k , produzindo duas placas menores k_1 e k_2 e, na Figura 3 (b) um corte vertical.

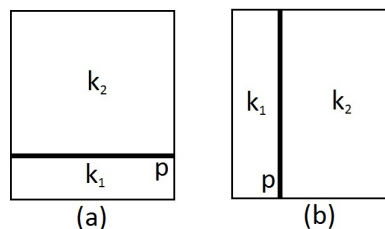


Figura 3: O corte na posição p produz duas placas k_1 e k_2 - (a) horizontal e (b) vertical (adaptada de Furini et al. [2016]).

Seja R o conjunto de placas no qual a placa retangular indexada por $k = 0$ indica a placa inicial e possui dimensões (H, W) , onde H é um limitante superior para a placa inicial (faixa); cada placa $k \in R$ tem dimensões $(H_k, W_k) \in \{1, \dots, H\} \times \{1, \dots, W\}$. Denotamos por $O = \{h, v\}$ o conjunto de possíveis orientações para um corte (1º estágio (h) - horizontal e 2º estágio (v) - vertical, respectivamente) e, $N \subseteq R$ indica o subconjunto de placas com dimensões iguais a um dos itens, deste modo também podemos denotar o conjunto de itens por N . Além disso, para toda placa k denotamos por $P^o(k)$ o conjunto de posições nas quais podemos cortar a placa k na orientação o .

Note que dependendo da cardinalidade dos conjuntos R e $P^o(k), k \in R, o \in O$, o modelo pode ter um número muito grande de variáveis. O número de placas que consideramos e o número de cortes realizados em cada placa (geralmente, produzindo novas placas) determinam, na prática, o tamanho do modelo. Assim, precisamos definir o conjunto de posições $P^o(k), k \in R, o \in O$, o qual será denominado conjunto de discretização da placa.

Seja I_k o conjunto de itens que cabem na placa k , ou seja, $I_k = \{j \in N; h_j \leq H_k \text{ e } w_j \leq W_k\}$. O conjunto completo das posições P onde um corte pode ser realizado inclui as dimensões de todos os itens $j \in N$, bem como todas as combinações das dimensões destes itens. Se considerarmos os itens ordenados de forma crescente pela altura (ou seja, $h_1 < h_2 < \dots < h_n$), temos que o conjunto de discretização da placa com dimensões (H_k, W_k) para cortes horizontais é dado por: $P^h(k) = \left\{ p; 0 < p < H_k, \exists z_j \in \{0, 1\}; p = \sum_{j \in I_k} z_j h_j \right\}$. E, considerando os itens ordenados de forma crescente pela largura ($w_1 < w_2 < \dots < w_n$), o conjunto de discretização para cortes verticais, é: $P^v(k) = \left\{ p; 0 < p < W_k, \exists z_j \in \{0, 1\}; p = \sum_{j \in I_k} z_j w_j \right\}$.

Estes conjuntos consideram o conceito de exclusão de Herz (ver Herz [1972]) e podem ser calculados utilizando uma técnica de programação dinâmica padrão (DP); ver, por exemplo, Christofides e Whitlock [1977].



Para a formulação MF, primeiramente precisamos de um limitante superior para a faixa, digamos H . A partir da placa 0, duas novas placas são obtidas por meio de um corte horizontal (1º estágio), com $P^h(0) \subseteq 1, \dots, H$, tais placas são armazenadas no conjunto R quando o seu tamanho é tal que pode caber algum item; caso contrário, é descartada. Note que não realizamos cortes verticais (2º estágio) na placa inicial (isto é, $P^v(0) = \emptyset$); além disso, no primeiro corte realizado na horizontal, consideramos que somente a parte inferior da placa inicial é um retângulo finito o qual pode ser utilizado como uma placa residual, enquanto que a parte superior da faixa é um retângulo "infinito". As variáveis e os parâmetros utilizados no modelo MF são obtidos processando o conjunto de itens N e a placa retangular inicial (placa 0) tal como descrito no Algoritmo 1.

Na primeira fase do algoritmo de enumeração de placas e variáveis geramos as faixas horizontais (níveis), que por sua vez são cortadas verticalmente a fim de obtermos os itens finais (2º estágio do corte). Note que, para obter alguns itens é necessário a "apara", que nada mais é que um corte perpendicular ao corte anterior (isto é, um corte horizontal) e a mesma não é computada como um novo estágio. Além disso, o algoritmo e o modelo precisam considerar demanda, ambos funcionam para demanda unitária, entretanto quando a instância avaliada possui itens repetidos é necessário agrupar estes itens.

Utilizamos as variáveis inteiras x_{pk}^o , as quais indicam o número de vezes que a placa do tipo k é cortada na posição p com orientação o , d_i representando a quantidade de placas do tipo i , e os parâmetros a_{pki}^o . Em Furini et al. [2016], estes parâmetros são definidos por: $a_{pki}^o = 1$, se a placa do tipo i é obtida pelo corte da placa k na posição p com orientação o e $a_{pki}^o = 0$ caso contrário; entretanto, em MF é preciso acrescentar o fato de que quando uma placa tipo k é cortada na posição p com orientação o gera duas placas idênticas, fazemos $a_{pki}^o = 2$.

Portanto, a formulação matemática MF é descrita por:

$$\text{Min } H \tag{6}$$

$$\text{sujeito a : } p x_{p0}^h \leq H, \quad p \in P^h(0) \tag{7}$$

$$\sum_{p \in P^h(0)} x_{p0}^h = 1 \tag{8}$$

$$\sum_{k \in R} \sum_{o \in O} \sum_{p \in P^o(k)} a_{pki}^o x_{pk}^o - \sum_{o \in O} \sum_{p \in P^o(i)} x_{pi}^o - d_i \geq 0, \quad i \in N; i \neq 0 \tag{9}$$

$$\sum_{k \in R} \sum_{o \in O} \sum_{p \in P^o(k)} a_{pki}^o x_{pk}^o - \sum_{o \in O} \sum_{p \in P^o(i)} x_{pi}^o \geq 0, \quad i \in R \setminus N \tag{10}$$

$$x_{pk}^o \in \mathbb{Z}_+, \quad k \in R, \quad o \in O, \quad p \in P^o(k) \tag{11}$$

A função objetivo (6) deste problema tem, como critério de otimização, a minimização da altura utilizada da placa inicial (*strip*). A altura H da placa é definida como mostrado nas restrições (7). As equações (8) impõem que o primeiro corte seja na horizontal (onde p é a posição do corte, com $p \in P^h(0)$). As restrições (9) garantem que o número de placas i que são cortadas ou mantidas como itens não exceda o número de placas i obtidas por meio do corte de algumas outras placas. As desigualdades (10) são equivalentes as restrições anteriores para placas $i \notin N$. E, a expressão (11) define o domínio das variáveis do problema.



início

```

Placa 0, conjunto de itens  $N$ ;
Inicialize  $R = 0$ ;
Marque 0 como não analisada;
enquanto  $R$  conter placas não analisadas para o 1º estágio do corte faça
    Pegue uma placa  $k \in R$  não analisada;
    Calcule o conjunto de posições  $P^h(k)$ ;
    para toda posição  $p \in P^h(k)$  fazer
        Corte  $k$  na posição  $p$  e gere as placas 1 e 2;
        se  $1 \notin R$  e cabe algum item em 1 então
             $R = R \cup \{1\}$ ;
        fim
        se  $2 \notin R$  e cabe algum item em 2 então
             $R = R \cup \{2\}$ ;
        fim
    fim
    Marque  $k$  como analisada para o 1º estágio do corte;
    Crie  $x_{pk}^h$  e  $a_{pki}^h$ ;
fim
enquanto  $R$  conter placas não analisadas para o 2º estágio do corte faça
    Pegue uma placa  $k \in R$  não analisada;
    Calcule o conjunto de posições  $P^v(k)$ ;
    para toda posição  $p \in P^v(k)$  fazer
        Corte  $k$  na posição  $p$  e gere as placas 1 e 2;
        se  $1 \notin R$  e cabe algum item em 1 então
             $R = R \cup \{1\}$ ;
        fim
        se  $2 \notin R$  e cabe algum item em 2 então
             $R = R \cup \{2\}$ ;
        fim
    fim
    Marque  $k$  como analisada para o 2º estágio do corte;
    Crie  $x_{pk}^v$  e  $a_{pki}^v$ ;
fim
fim
    
```

Algoritmo 1: Algoritmo para gerar parâmetros e variáveis.

Exemplo 3. Considere o exemplo 2, apresentado anteriormente, uma altura inicial (limitante superior para a faixa) $H = 44$ e os três tipos de itens, indexados por j , reordenados conforme apresentado na Tabela 2.

Tabela 2: Dados relativo aos itens.

Tipo de item (j)	Altura (h_j)	Largura (w_j)
0	6	10
1	7	5
2	9	5

Sem perda de generalidade podemos admitir que o índice da placa inicial (faixa) é dado



por $n + 1 = 3$, deste modo, segue-se que o conjunto de itens que cabem na placa inicial é dado por $I_3 = \{0, 1, 2\}$ e que o conjunto de posições na horizontal é $P^h(3) = \{6, 7, 9, 13, 15, 16, 22\}$.

No primeiro corte da primeira fase do algoritmo (1º estágio do corte), para toda posição $p \in P^h(3)$ realizamos um corte horizontal obtendo uma faixa para cada corte (lembre-se de que no primeiro corte realizado na horizontal, consideramos que somente a parte inferior da placa inicial é um retângulo finito o qual pode ser utilizado como uma placa residual). Por meio destes cortes obtemos 7 tipos de faixas (placas residuais) todas com largura $W_k = 10$ e altura 6, 7, 9, 13, 15, 16 e 22, respectivamente (ver Figura 4).

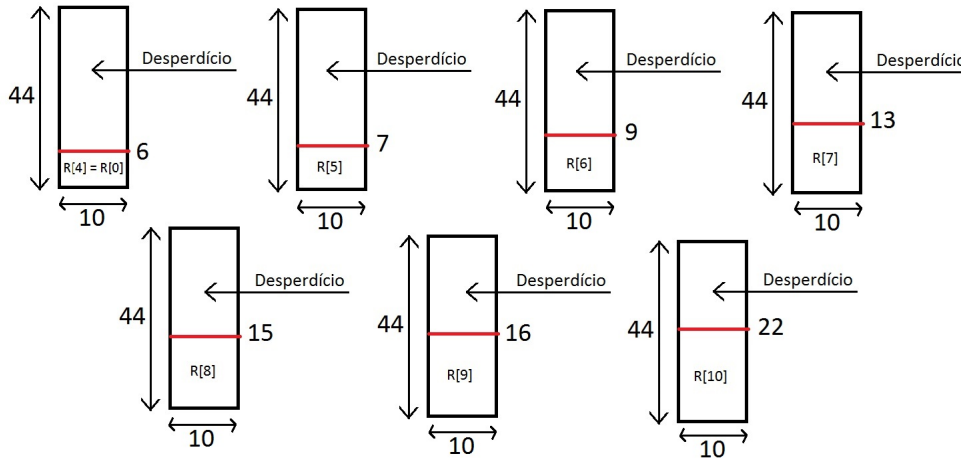


Figura 4: Primeiro corte da primeira fase do algoritmo.

Os traços em vermelho indicam a posição que foi realizado o corte e, estes cortes, geram os parâmetros $a_{630}^h, a_{735}^h, a_{936}^h, a_{13,3,7}^h, a_{15,3,8}^h, a_{16,3,9}^h$ e $a_{22,3,10}^h$, bem como as variáveis $x_{63}^h, x_{73}^h, x_{93}^h, x_{13,3}^h, x_{15,3}^h, x_{16,3}^h$ e $x_{22,3}^h$.

A partir do "segundo" corte executado na horizontal, para toda posição p é realizado um corte que gera uma ou duas placas residuais (superior e inferior). Um exemplo de um corte é a obtenção de uma placa do tipo 6 e uma placa do tipo 0 a partir de uma placa do tipo 8, conforme mostra a Figura 5. Por meio deste corte obtemos a variável x_{68}^h e os parâmetros a_{680}^h e a_{686}^h .

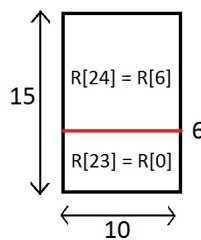


Figura 5: Obtenção de uma placa do tipo 6 e uma placa do tipo 0 a partir de uma placa do tipo 8.

Na segunda fase do algoritmo (2º estágio do corte), realizamos o corte na vertical e também geramos uma ou duas placas residuais (à direita e à esquerda). Por exemplo, cortando uma placa do tipo 6 na posição 5, obtemos duas placas do tipo 2 e geramos a variável x_{56}^v e o parâmetro a_{562}^v que, neste caso, recebe o valor 2 (ver Figura 6).

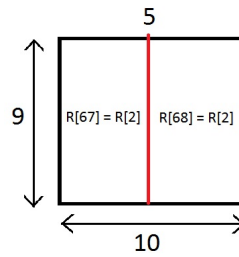


Figura 6: Obtenção de duas placas do tipo 2 a partir de uma placa do tipo 6.

Para finalizar precisamos realizar a "apara" de modo a obtermos um item final. A Figura 7 mostra uma "apara" para a obtenção de um item do tipo 1. Tal "apara" gera a variável x_{72}^h e o parâmetro a_{721}^h .

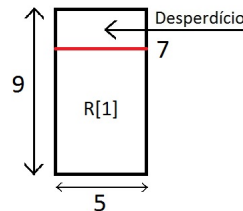


Figura 7: Apara para a obtenção de um item do tipo 1

E assim, obtem-se o modelo MF para este exemplo. A solução ótima deste problema é 15. A Figura 3 apresenta o layout desta solução, no qual os traços em negrito representam os cortes executados (no 1º e 2º estágios) e a parte hachurada denota o desperdício.

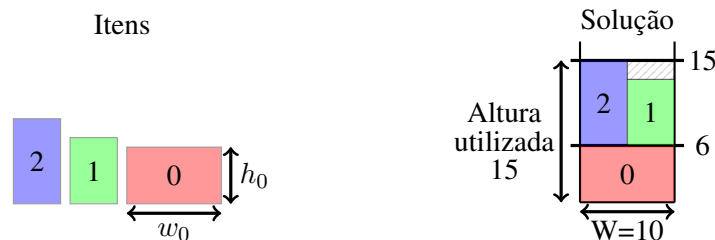


Figura 8: Solução do exemplo utilizando o modelo matemático MF.

3. Resultados Computacionais

Nesta seção, apresentamos os testes computacionais realizados para avaliar os modelos descritos anteriormente para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios em nível, bem como a relaxação linear destas formulações matemáticas. Os modelos analisados foram implementados em linguagem C++ utilizando-se a ferramenta *Concert Technology* do *solver* comercial cplex 12.6 com a parametrização padrão. Os testes computacionais apresentados a seguir foram realizados de maneira sequencial em um computador equipado com 2 Processadores Intel Xeon E5-2680v2 de 2.8 GHz com dez núcleos e com 128 GB DDR3 1866MHz de memória RAM. Foi fixado um limite de tempo de execução de 3600 segundos, de modo que, nos casos em que a solução ótima não foi encontrada no tempo limite, capturou-se a melhor solução fornecida pelo *solver* e o limitante inferior. O *Gap* foi calculado pela expressão $Gap = \frac{(z - lb)}{z} \times 100$, onde z representa a solução incumbente e lb o melhor limitante inferior encontrado.



Para analisar o desempenho dos modelos implementados neste trabalho, foram efetuados testes computacionais considerando um conjunto de instâncias da literatura. Este conjunto é composto por 10 classes representadas por $Class1, \dots, Class10$. Cada uma destas classes possui 50 problemas-teste, sendo 10 para cada valor de $n \in \{20, 40, 60, 80, 100\}$. Aqui utilizamos a $Class2$ proposta por Berkey e Wang [1987]. Vale ressaltar que em Furini et al. [2016] não foi apresentado testes computacionais exaustivos para o problema de empacotamento em faixas, neste artigo os autores analisaram instâncias com no máximo 5 itens; além disso, não comparam o desempenho da modelagem com modelos da literatura.

A Tabela 3 compara os resultados obtidos pelo modelo Lodi et al. (2004) com os obtidos pela formulação MF. Suas colunas estão dispostas da seguinte maneira: a coluna Instâncias indica a classe de problemas utilizada; a segunda coluna, indicada por n , representa o número de itens das instâncias e, as seções Lodi et al. (2004) e MF contém as colunas N_R (número de instâncias resolvidas na otimalidade), N_{CR} (número de instâncias que o cplex não conseguiu nem carregar o modelo/obter uma solução inicial no tempo limite), N_V (número médio de variáveis do modelo), G_{Me} (Gap médio em porcentagem) e t_{Me} (tempo médio de execução, em segundos de CPU). Note que indicamos por TL, quando a instância atinge o limite de tempo de execução fixado.

Tabela 3: Desempenho dos modelos Lodi et al. (2004) e MF para as instâncias da $Class2$.

Instâncias	n	Lodi et al. (2004)					MF				
		N_R	N_{CR}	$N_V(\times 10^2)$	$G_{Me}(\%)$	$t_{Me}(s)$	N_R	N_{CR}	$N_V(\times 10^4)$	$G_{Me}(\%)$	$t_{Me}(s)$
$Class2$	20	10	0	2.03	0.00	2.78	8	0	2.99	1.53	1128.12
	40	10	0	8.11	0.00	13.62	0	0	6.26	5.92	TL
	60	6	0	18.19	1.07	1898.14	0	0	11.60	43.13	TL
	80	1	0	32.28	2.70	3375.30	0	4	17.11	52.13	TL
	100	0	0	50.24	2.93	TL	0	10	23.34	—	TL

Como pode ser visto na Tabela 3, considerando o tempo limite de 1 hora de execução, o cplex foi capaz de encontrar a solução ótima em 27 das 50 instâncias contidas na $Class2$ (ou seja, em 54%) para o modelo de Lodi et al. [2004]. Entretanto, para a formulação matemática MF o cplex só foi capaz de provar a otimalidade em 8 instâncias (isto é, em 16% das instâncias avaliadas). Além disso, podemos observar que o Gap médio para as instâncias com 80 itens é de 2.70% no modelo da literatura, enquanto que em MF este Gap é muito superior (52.13%). Mais ainda, nas instâncias contendo 100 itens, o cplex nem consegue carregar o modelo/obter uma solução inicial no tempo limite para a formulação MF. Portanto, podemos concluir que, para este problema, a extensão do modelo Furini et al. [2016] não se mostrou vantajosa, principalmente devido ao número de variáveis necessárias para a construção do modelo matemático.

3.1 Relaxação linear

Nesta seção, são descritos os resultados obtidos por meio da relaxação linear dos modelos apresentados anteriormente para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios em nível. A Tabela 4 mostra estes resultados e, suas colunas estão dispostas como segue: a coluna Instâncias indica a classe de problemas utilizada; a segunda coluna, indicada por n , representa o número de itens da instância e, as seções Lodi et al. (2004) e MF contém as colunas z_{Me} e t_{Me} indicando, respectivamente, o valor médio da solução encontrada e o tempo médio de processamento do cplex, em segundos de CPU.



Tabela 4: Desempenho da relaxação linear dos modelos Lodi et al. (2004) e MF para as instâncias da *Class2*.

Instâncias	n	Lodi et al. (2004)		MF	
		z_{Me}	$t_{Me}(s)$	z_{Me}	$t_{Me}(s)$
<i>Class2</i>	20	21.16	0.01	0.20	1.36
	40	39.68	0.01	0.18	4.13
	60	60.14	0.02	0.18	6.91
	80	82.82	0.04	0.19	12.37
	100	99.99	0.05	0.18	21.12

Ao relaxar a condição de integralidade podemos observar que a formulação MF não apresenta bons resultados (ver tabela 4), o que já era esperado visto que a altura da placa inicial (H) depende apenas do valor da "maior" variável multiplicado pela sua posição ($p x_{p0}^h$). Assim, como os itens podem se sobrepor e o valor de x_{p0}^h é pequeno, temos que o valor de H é pequeno.

4. Conclusões

Neste trabalho, apresentamos a formulação matemática MF para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios. Além disso, utilizando instâncias da literatura, realizamos uma comparação entre esta a formulação MF e o modelo apresentado em Lodi et al. [2004]. Vale ressaltar que o modelo MF não apresenta um bom desempenho para o problema, pois em várias situações, o cplex nem consegue carregar o modelo/obter uma solução inicial no tempo limite. O modelo MF consiste em um caso particular da formulação matemática introduzida em Furini et al. [2016] para tratar o problema de empacotamento bidimensional. Note que, por meio dos testes computacionais realizados, podemos concluir que abordagem não é viável. O próximo passo será adaptar outros modelos de corte e estoque bidimensional da literatura para o problema de empacotamento em faixas bidimensional guilhotinado 2-estágios.

Agradecimentos

Os autores agradecem ao Centro de Ciências Matemáticas Aplicadas à Indústria - Ce-MEAI (CEPID FAPESP No. 2013/07375-0), ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e à Universidade Federal da Grande Dourados (UFGD).



Referências

- Berkey, J. O. e Wang, P. Y. (1987). Two-dimensional finite bin-packing algorithms. *Journal of the operational research society*, p. 423–429.
- Christofides, N. e Whitlock, C. (1977). An algorithm for two-dimensional cutting problems. *Operations Research*, 25(1):30–44.
- Furini, F., Malaguti, E., e Thomopulos, D. (2016). Modeling two-dimensional guillotine cutting problems via integer programming. *INFORMS Journal on Computing*, 28(4):736–751.
- Gilmore, P. C. e Gomory, R. E. (1961). A linear programming approach to the cutting-stock problem. *Operations research*, 9(6):849–859.
- Gilmore, P. C. e Gomory, R. E. (1963). A linear programming approach to the cutting stock problem-part ii. *Operations research*, 11(6):863–888.
- Gilmore, P. C. e Gomory, R. E. (1965). Multistage cutting stock problems of two and more dimensions. *Operations research*, 13(1):94–120.
- Herz, J. (1972). Recursive computational procedure for two-dimensional stock cutting. *IBM Journal of Research and Development*, 16(5):462–469.
- Kantorovich, L. V. (1960). Mathematical methods of organizing and planning production. *Management Science*, 6(4):366–422.
- Lodi, A. (1999). *Algorithms for two-dimensional bin packing and assignment problems*. PhD thesis, Universita Degli Studi di Bologna.
- Lodi, A., Martello, S., e Monaci, M. (2002). Two-dimensional packing problems: A survey. *European journal of operational research*, 141(2):241–252.
- Lodi, A., Martello, S., e Vigo, D. (1999). Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. *INFORMS Journal on Computing*, 11(4):345–357.
- Lodi, A., Martello, S., e Vigo, D. (2004). Models and bounds for two-dimensional level packing problems. *Journal of Combinatorial Optimization*, 8(3):363–379.
- Wascher, G., HauBner, H., e Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3):1109 – 1130.