



## Um modelo de programação não-linear inteira mista para o problema de empacotamento de cilindros em níveis

**Rainne Florisbello Gonçalves**

Instituto de Ciências Matemáticas e de Computação  
Avenida Trabalhador São-carlense, 13566-590, São Carlos - SP, Brasil  
rainne.fg@usp.br

**Luiz Henrique Cherri**

Instituto de Ciências Matemáticas e de Computação  
Avenida Trabalhador São-carlense, 13566-590, São Carlos - SP, Brasil  
lhcherri@icmc.usp.br

**Marina Andretta**

Instituto de Ciências Matemáticas e de Computação  
Avenida Trabalhador São-carlense, 13566-590, São Carlos - SP, Brasil  
andretta@icmc.usp.br

### RESUMO

No problema de empacotamento de cilindros em níveis desejamos posicionar itens cilíndricos em recipientes paralelepípedicos com o intuito de minimizar a quantidade de recipientes utilizados. Para este caso, considera-se a possibilidade de adicionar níveis ao recipiente. Neste trabalho, apresentamos um modelo de programação não-linear inteira mista para o problema. Este modelo foi desenvolvido com base no problema encontrado em indústrias de cerâmica e resolvido por um *solver* de otimização global. Os testes computacionais foram realizados utilizando instâncias criadas pelos próprios autores. Com os resultados obtidos foi possível verificar que o modelo é capaz de encontrar soluções viáveis em uma média de tempo razoável, principalmente para instâncias pequenas.

**PALAVRAS CHAVE.** Corte e empacotamento, Modelo de otimização não-linear, Indústria de cerâmica.

**Área principal:** IND - PO na Indústria, PM - Programação Matemática

### ABSTRACT

In the problem of cylinder packing in levels the aim is to allocate cylindrical items in units, given as parallelepipeds, by minimizing the amount of units. In this case, it is possible to add levels in the unit. In this paper, we present a quadratic integer mixed mathematical model for the problem. This model was developed based on a problem found in ceramic industries and it was solved by a global optimization solver. The computational experiments were run over instances created by the authors. The results have shown that it is possible to find optimal solution mainly for small instances.

**KEYWORDS.** Cutting and packing, Non-linear optimization modelling, Ceramic industry.

**Main area:** OR in Industry, Mathematical Programming



## 1. Introdução

Problemas de corte e empacotamento buscam determinar a posição de objetos menores (chamados de itens) dentro de objetos maiores (chamados de recipientes) de maneira ótima. Isso implica em obedecer a certas restrições, e sempre com o objetivo de minimizar custos, seja através da maximização do espaço ocupado ou da minimização dos desperdícios. Teoricamente, problemas de corte e empacotamento podem ser tratados sem distinção, visto que cortar ou empacotar, nesse caso, refere-se à repartição de um espaço em porções menores, seja para o corte ou empacotamento. São considerados problemas da classe de problemas NP-difícil [Fowler et al., 1981], e têm em comum a existência de um subproblema geométrico, motivado pelas restrições de não sobreposição entre os itens.

A importância de se estudar esses problemas dá-se pela necessidade em otimizar os processos industriais. Entre as empresas beneficiadas nestes estudos encontram-se as indústrias de transporte e armazenamento, madeira, papel, vidro, metal e cerâmica, dentre outras.

Neste trabalho, estudamos um caso particular dos problemas de empacotamento, comumente encontrado em indústrias de cerâmica, que diz respeito à disposição de itens cerâmicos dentro de fornos para que seja feita a queima, reduzindo a quantidade de fornos necessários. Neste problema, considera-se o uso de prateleiras dentro do forno para que os produtos possam ser dispostos em diversos andares, melhorando a ocupação do forno. Foi proposto um modelo matemático não-linear inteiro misto para o problema, cuja resolução foi testada quanto à sua eficácia.

Consideramos este trabalho interessante tanto para a indústria quanto para a comunidade acadêmica. As principais contribuições deste trabalho estão na diminuição do tempo de preparo do forno, melhor ocupação do forno e diminuição da quantidade de fornadas necessárias. Com relação à comunidade acadêmica, consideramos o trabalho relevante devido às poucas contribuições existentes na literatura envolvendo o problema de empacotamento em níveis [Zhao et al., 2014]. Alguns destes trabalhos serão discutidos na Seção 3.

O trabalho está organizado em seis seções. Na Seção 2, é apresentado o problema estudado neste trabalho. Na Seção 3, é feita uma revisão da literatura dos trabalhos relevantes para o desenvolvimento deste estudo. Na Seção 4, é apresentada a formulação matemática desenvolvida para o problema estudado. A Seção 5 contém os resultados numéricos obtidos ao buscar solução do modelo proposto para algumas instâncias, usando um *solver* de otimização global. Na Seção 6, são apresentadas as conclusões.

## 2. Definição do problema

Na indústria de cerâmica, um dos processos produtivos é a queima do produto em um forno para que ele possa adquirir suas propriedades finais. Dispor os objetos dentro do forno é um desafio comumente encontrado, pois a quantidade de objetos pode variar de algumas dezenas a milhares e o tempo de queima é de cerca de seis horas, considerando uma indústria de cerâmica de alta tecnologia. Assim, diminuir a quantidade de fornadas é de suma importância. O forno tem uma base retangular e seu teto é uma abóboda, porém, por simplicidade, o interior do forno será considerado como um paralelepípedo neste trabalho.

A distribuição dos objetos dentro do forno é feita pelo operador de forno, de acordo com o que ele considera a melhor opção de aproveitamento. Os objetos devem ser posicionados sem que haja sobreposição entre eles e devem estar inteiramente contidos no forno. Este processo é muito lento e, de fato, os funcionários não têm uma medida clara para saber se a disposição adotada é realmente satisfatória. Com isso, fornecer uma possível solução para a indústria já seria de grande importância. Como a maioria dos objetos produzidos apresentam formato cilíndrico, por simplificação, serão considerados apenas objetos cilíndricos neste trabalho. Vale ressaltar, também, que as indústrias geralmente posicionam itens menores dentro de itens maiores, porém este caso não será avaliado neste trabalho.



Quando na disposição dos objetos dentro do forno, podem ser usadas prateleiras (níveis) de modo que os objetos possam ser dispostos em diversos andares, aprimorando a ocupação. Ao ser considerada a possibilidade de usar prateleiras dentro dos fornos, cada prateleira adicionada representa uma divisão no forno. Deve-se, portanto, decidir quantas prateleiras serão inseridas e como será a disposição dos objetos nessas prateleiras. Nesta pesquisa, por simplificação, considera-se que a espessura da prateleira é desprezível, não interferindo significativamente na ocupação do forno. A Figura 1 apresenta uma representação dos itens à esquerda e do forno à direita. Cada prateleira usada para o posicionamento dos objetos dentro do forno corresponde a um novo nível, sendo a base do recipiente o primeiro nível. Dessa forma, dispor os objetos em prateleiras corresponde à resolução do problema de empacotamento em níveis.

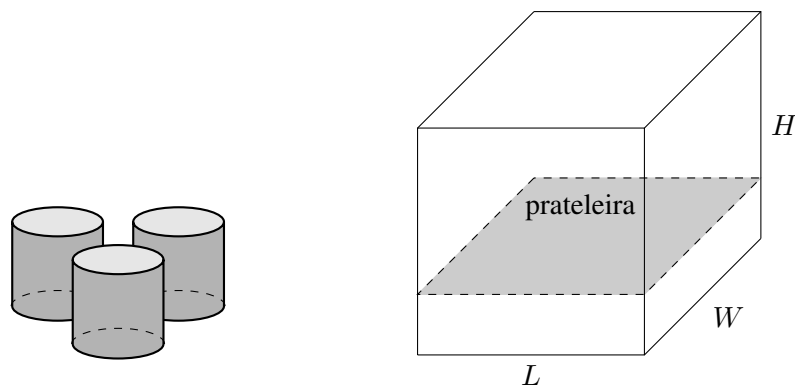


Figura 1: Representação utilizada para as peças e para o forno.

O problema de empacotamento de produtos cerâmicos em fornos é um problema tridimensional que diz respeito ao posicionamento de itens em um recipiente com formato de um paralelepípedo, podendo ser definido da seguinte maneira: dado um conjunto de  $n$  itens cilíndricos de diversos tamanhos e uma quantidade infinita de recipientes, deseja-se determinar como empacotar um subconjunto de itens dentro de cada recipiente, de modo que todos os itens sejam empacotados utilizando a menor quantidade de recipientes. É necessário que os itens não se sobreponham e a disposição final respeite as dimensões do recipiente. Neste caso, os itens são empacotados em níveis, de modo que o primeiro nível corresponde à base do recipiente e os níveis subsequentes são determinados de acordo com os critérios de resolução do problema.

### 3. Trabalhos relacionados

Os trabalhos a seguir tratam dos problemas de empacotamento de cilindros, empacotamento de círculos e empacotamento de retângulos em níveis, que são a base para a modelagem e resolução do problema tratado nesta pesquisa, visto que não foram encontrados na literatura trabalhos referentes ao problema de empacotamento de cilindros em níveis.

#### 3.1. Empacotamento de cilindros

Stoyan et al. [2016] utilizaram as funções  $\phi$  e quasi- $\phi$  para modelar o problema de empacotamento de itens convexos tridimensionais. Essa estratégia permite utilizar métodos modernos de otimização não-linear em todos os estágios, incluindo a definição do ponto inicial, a obtenção de um ponto estacionário e aproximações do mínimo global. A estratégia desenvolvida pelos autores pode ser utilizada para resolver outros problemas de empacotamento, como de itens não convexos, mostrando a eficiência da abordagem proposta.

Em geral, o problema de empacotamento de cilindro é tratado na literatura reduzindo-o à sua versão bidimensional. Isso ocorre quando os cilindros apresentam mesma altura podendo, então, desconsiderá-la e trabalhar apenas com a base circular do item. Dessa forma, tem-se um



problema de empacotamento de itens circulares em recipientes de bases retangulares, como paletes e contêineres. Assim, sua formulação e métodos de resolução são basicamente os mesmos de problemas de empacotamento de círculos. Esses problemas serão tratados na próxima seção.

### 3.2. Empacotamento de círculos

George et al. [1995] criaram uma formulação matemática não-linear e discreta para o problema de empacotamento de círculos diferentes que consiste em maximizar o valor dos itens empacotados, de modo a garantir que nenhuma parte do círculo fique fora do recipiente e que os itens não se sobreponham. Os autores consideraram sua formulação intratável quando analisados problemas de larga escala e difícil de resolver mesmo para casos com poucos itens, por isso heurísticas foram avaliadas como método de solução mais adequado. Suas heurísticas, baseadas em regras intuitivas de empacotamento, mostraram que um algoritmo simples de randomização pode ser superior ao algoritmo genético, chegando a uma solução satisfatória mais rapidamente.

Com o objetivo de posicionar círculos de raios diferentes dentro de um recipiente retangular minimizando a distância entre os itens empacotados, Stoyan e Yas'kov [1998] desenvolveram um método exato para buscar um ponto estacionário. Como o problema apresenta função objetivo linear e restrições não-lineares, os autores utilizaram um método de solução baseado na combinação entre uma modificação do algoritmo *Branch-and-Bound*, para achar os pontos estacionários, e o Método do Gradiente Reduzido. Eles destacam, ainda, que o ponto estacionário encontrado pela árvore de busca nem sempre corresponde a um mínimo local, então, utilizando esse ponto como ponto inicial para o Método do Gradiente Reduzido, um mínimo local é obtido.

Birgin e Gentil [2010] trabalharam com o problema de empacotamento de círculos de raios unitários em recipientes de diferentes formas geométricas, como círculos, quadrados, retângulos, faixas e triângulos. Os autores desenvolveram um método de convergência quadrática que tem como objetivo minimizar a dimensão do recipiente de modo que não haja sobreposição dos itens e estes permaneçam inteiramente contidos no recipiente. Para a maioria dos casos foram alcançados resultados equivalentes aos presentes na literatura. Entretanto, para o empacotamento em triângulo, retângulo e faixa houve uma significativa contribuição, pois não haviam sido ainda publicados resultados para estes problemas.

Uma formulação similar à de George et al. [1995] pode ser encontrada no trabalho de Litvinchev e Espinosa [2014]. Eles propuseram uma formulação não-linear inteira mista para o problema de empacotamento de círculos em retângulos com o objetivo de maximizar a área do recipiente coberta pelos círculos. De acordo com os autores, a ideia de utilizar variáveis inteiras surge como alternativa para explorar melhor o espaço de busca.

### 3.3. Empacotamento de retângulos em níveis

O estudo do problema de empacotamento de retângulos em níveis iniciou-se com Gilmore e Gomory [1965], com o problema de corte de estoque 2-estágios exato, e também pode ser encontrado em trabalhos como de Coffman et al. [1980], Chung et al. [1982], Frenk e Galambos [1987], Lodi et al. [1999], Lodi et al. [2002a], Lodi et al. [2002b], e trabalhos posteriores.

A primeira proposta de resolução para este problema foi dada por Chung et al. [1982]. Eles consideraram o primeiro nível como sendo a parte inferior do recipiente e que um novo nível é criado coincidindo com o topo do item mais alto empacotado no nível anterior. A solução é obtida em duas fases. A primeira consiste em resolver o problema de empacotamento bidimensional em faixa. Neste caso, é dado um recipiente com largura definida e altura infinita com o objetivo de empacotar todos os itens de modo a minimizar a altura do recipiente. A segunda fase é subdividida em recipientes finitos com altura fixa. Essa etapa consiste em resolver o problema de empacotamento unidimensional em que cada prateleira é vista como um elemento com valor equivalente à altura do item mais alto contido neste nível.



Uma formulação matemática linear inteira para esse problema foi proposta por Lodi et al. [2002a], com o objetivo de minimizar a quantidade de recipientes utilizada. Uma modelagem similar pode ser encontrada no trabalho de Puchinger e Raidl [2007].

Lodi et al. [2002b] explicam que para a resolução desse problema existem três estratégias clássicas, as quais são derivadas de algoritmos para o caso em uma dimensão. Considerando que os itens são agrupados por suas alturas, sendo empacotados nessa sequência, essas são as estratégias: *Next-Fit Decreasing Height* (NFDH), em que o item  $j$  é empacotado alinhado à esquerda no nível  $s$ , se couber. Caso contrário, um novo nível é criado e o item  $j$  é empacotado nele. Neste caso, os níveis anteriores não são revisitados; *First-Fit Decreasing Height* (FFDH), em que o item  $j$  é empacotado alinhado à esquerda no primeiro nível, onde couber. Se o primeiro nível não acomodar o item  $j$ , tenta-se posicioná-lo nos níveis subsequentes já criados, se não couber, um novo nível é inicializado como um NFDH; e *Best-Fit Decreasing Height* (BFDH), em que o item  $j$  é empacotado alinhado à esquerda no nível, entre aqueles que cabem, com o mínimo espaço horizontal inutilizado. Se nenhum nível já criado for capaz de acomodar o item  $j$ , um novo nível é inicializado como um NFDH. Autores como Chung et al. [1982], Lodi et al. [1999], Kang e Park [2003] e Ortmann et al. [2010] propuseram modificações nessas heurísticas com o objetivo de melhorar a qualidade da solução.

#### 4. Formulação matemática

No problema de empacotamento de itens cilíndricos em recipientes paralelepípedos, considerando o empacotamento em níveis, deve-se empacotar uma quantidade fixa de itens ( $n$ ) utilizando o menor número de recipientes possível, com largura ( $L$ ), comprimento ( $W$ ) e altura ( $H$ ) fixos. Esses recipientes são iguais e, portanto, apresentam as mesmas medidas. São conhecidos também a altura ( $h_i$ ) e raio ( $r_i$ ) de cada item.

De acordo com a modelagem proposta para este problema, a quantidade de recipientes e a quantidade de níveis necessários não são considerados como variáveis. Dessa forma, é feita uma estimativa de quantos recipientes ( $m$ ) e quantos níveis ( $p$ ) são necessários, no máximo, para que os itens sejam empacotados e, a partir disso, minimizar a quantidade de recipientes.

Para isso, é utilizada uma variação heurística *Bottom Left* de Gomes e Oliveira [2002] proposta para o problema de empacotamento de itens irregulares em faixa, a qual fornecerá a estimativa de recipientes e níveis utilizados. Na heurística, primeiramente é criada uma lista dos itens, ordenada em ordem decrescente de altura e área da base. Por simplicidade, considera-se que todos os níveis têm a altura do item mais alto do nível. Em seguida, o primeiro item da lista é empacotado no canto inferior esquerdo do primeiro nível do primeiro recipiente. Os demais itens são empacotados na posição mais à esquerda e abaixo possível neste andar. Quando um item não pode ser empacotado neste nível, um novo nível é iniciado para que este seja alocado (sempre no canto mais à esquerda e abaixo possível). Na última etapa, tenta-se inserir o  $n$ -ésimo item em um dos andares já existentes, e, se não for possível, este é inserido em um novo andar. Ao final do processo, verifica-se o número de níveis utilizados e calcula-se quantos fornos seriam necessários para alocar este número de níveis.

A estimativa da quantidade de recipientes e de níveis é feita da seguinte maneira: primeiro calcula-se o número máximo de itens que podem ser empilhados no forno com base no item de menor altura, dado por  $n_{max} = \frac{H}{h_{min}}$ , em que  $h_{min}$  é a altura do menor item. Depois disso, para estabelecer a quantidade de níveis é calculada uma média entre a quantidade de níveis fornecida pela heurística *Bottom-Left* ( $P_{BL}$ ) e  $n_{max}$ , ou seja,  $P_{media} = \frac{P_{BL} + n_{max}}{2}$ . A quantidade de recipientes é dada pelo mínimo entre a quantidade de recipientes fornecida pela heurística e o número de itens a ser empacotado. Vale destacar que, similar a uma situação que se encontra na prática, a quantidade de níveis em cada recipiente é limitada por cinco. Dessa forma, caso  $P_{media}$  seja maior que cinco, a quantidade de níveis recebe o valor 5 e a quantidade de recipientes é recalculada.



O modelo matemático proposto a seguir está baseado nos trabalhos de George et al. [1995] e Lodi et al. [2002a].

### Parâmetros

$m$	quantidade total de recipientes,	
$n$	quantidade total de itens,	
$p$	quantidade total de níveis para cada recipiente,	
$L \times W \times H$	largura $\times$ comprimento $\times$ altura de cada recipiente,	
$r_i$	raio do item $i$ ,	$i = 1, \dots, n$ ,
$h_i$	altura do item $i$ ,	$i = 1, \dots, n$ .

### Variáveis

$$\delta_{iks} = \begin{cases} 1, & \text{se o item } i \text{ está empacotado no nível } k \text{ do recipiente } s, \\ 0, & \text{caso contrário,} \end{cases}$$

$$\gamma_s = \begin{cases} 1, & \text{se o recipiente } s \text{ é usado,} \\ 0, & \text{caso contrário,} \end{cases}$$

$$\beta_{ks} = \begin{cases} 1, & \text{se o nível } k \text{ é usado no recipiente } s, \\ 0, & \text{caso contrário,} \end{cases}$$

$\lambda_{ks}$  altura (em relação ao recipiente) onde o nível  $k$  do recipiente  $s$  é posicionado,

$(\sigma_i^x, \sigma_i^y)$  centro do item  $i$  em relação aos eixos  $x$  e  $y$ , respectivamente,

com  $i = 1, \dots, n$ ,  $k = 1, \dots, p$  e  $s = 1, \dots, m$ .

A formulação matemática para o problema de empacotamento de cilindros em níveis é dada pelo modelo (1)-(11).

A função objetivo (1) consiste em minimizar a quantidade de recipientes utilizada para empacotar todos os itens. Para posicionar um item dentro do recipiente, é necessário que os limites do recipiente sejam obedecidos. As Restrições (2) e (3) garantem que a fronteira do recipiente não seja violada. As Restrições (5) impedem que os itens se sobreponham. As Restrições (6) garantem que cada item é empacotado apenas uma vez. Quando um item é designado para um determinado nível e recipiente, ambos são considerados usados. Isso é garantido pelas Restrições (7) e (8). Quando um nível é usado, é necessário que a altura em que ele esteja posicionado seja maior que a altura dos itens empacotados no nível anterior, o que é estabelecido pelas Restrições (9). Assim, só é preciso verificar se os itens posicionados no último nível obedecem à altura do recipiente, o que é verificado pela Restrições (4). Por fim, as Restrições (10) e (11) definem o domínio das variáveis.

Considera-se a base do recipiente como um nível e que a verificação de sobreposição é feita pela distância euclidiana dos centros dos itens empacotados, para itens posicionados no mesmo recipiente e em um mesmo nível. Além disso, os itens cilíndricos são representados pelos seus centros  $(\sigma_i^x, \sigma_i^y)$ , seus raios  $r_i$  e suas alturas  $h_i$ . Já o recipiente é representado pelo seu canto inferior esquerdo e suas dimensões.



$$\begin{aligned}
 &\text{minimizar} && \sum_{s=1}^m \gamma_s && (1) \\
 &\text{sujeito a} && r_i \leq \sigma_i^x \leq L - r_i, && i = 1, \dots, n, && (2) \\
 & && r_i \leq \sigma_i^y \leq W - r_i, && i = 1, \dots, n, && (3) \\
 & && \delta_{ips} h_i \leq H - \lambda_{ps}, && i = 1, \dots, n, \quad s = 1, \dots, m, && (4) \\
 & && (\sigma_i^x - \sigma_j^x)^2 + (\sigma_i^y - \sigma_j^y)^2 \geq \delta_{iks} \delta_{jks} (r_i + r_j)^2, && k = 1, \dots, p, \quad s = 1, \dots, m, && (5) \\
 & && && i = 2, \dots, n, \quad j = 1, \dots, i - 1, && (5) \\
 & && \sum_{s=1}^m \sum_{k=1}^p \delta_{iks} = 1, && i = 1, \dots, n && (6) \\
 & && \sum_{i=1}^n \delta_{iks} \leq n \beta_{ks}, && k = 1, \dots, p, \quad s = 1, \dots, m, && (7) \\
 & && \sum_{k=1}^p \beta_{ks} \leq p \gamma_s, && s = 1, \dots, m, && (8) \\
 & && \lambda_{ks} \geq \delta_{i,k-1,s} h_i, && i = 1, \dots, n, \quad k = 2, \dots, p, && (9) \\
 & && && s = 1, \dots, m, && (9) \\
 & && \lambda_{ks}, \sigma_i^x, \sigma_i^y \geq 0, && i = 1, \dots, n, \quad k = 1, \dots, p, && (10) \\
 & && && s = 1, \dots, m, && (10) \\
 & && \delta_{iks}, \gamma_s, \beta_{ks} \in \{0, 1\}, && i = 1, \dots, n, \quad k = 1, \dots, p, && (11) \\
 & && && s = 1, \dots, m. && (11)
 \end{aligned}$$

## 5. Experimentos computacionais

O modelo proposto para o problema de empacotamento de cilindros em níveis foi codificado em linguagem Julia 0.5<sup>1</sup> e foram utilizados os pacotes *JuMP* e *CoinOptServices* para a sua resolução. *JuMP* é uma linguagem de modelagem para otimização matemática incorporada em Julia que suporta *solvers* comerciais e *open-source* para solucionar uma variedade de problemas, incluindo os de otimização não-linear inteira mista. *CoinOptServices* dá acesso ao *Couenne*, um *solver* utilizado dentro do pacote *JuMP* para resolver problemas de otimização global.

Todos os experimentos foram realizados em um computador Intel Core i7 de 3.60GHz e 16GB de memória, usando o sistema operacional Ubuntu 16.04 LTS. Nas seções a seguir são descritas as instâncias utilizadas e apresentados os resultados numéricos.

### 5.1. Instâncias

Estamos considerando que os recipientes são de mesmo tamanho (como já mencionado na Seção 4). Para definirmos suas dimensões, foi aplicada uma escala em relação aos fornos encontrados em indústrias de cerâmica. Dessa forma, os recipientes onde são posicionados os itens possuem 80 cm de largura, 80 cm de comprimento e 120 cm de altura.

Como não foram encontradas na literatura instâncias referentes ao problema estudado neste trabalho, as instâncias foram criadas considerando três classes de itens, respectivamente, com: altura e raios variáveis, altura única e raio único. Em cada classe foram geradas quatro subclasses com: instâncias de itens pequenos, médios, grandes e variáveis. As instâncias de itens pequenos dizem respeito aos raios ou alturas dos itens que ocupam entre 1% e 25% do recipiente; as de itens médios ocupam entre 26% e 50% do recipiente; as de itens grandes ocupam entre 51% e 75% do recipiente, enquanto as de itens variáveis podem ocupar entre 1% e 75% do recipiente.

<sup>1</sup>julialang.org



Dessa forma, as instâncias são definidas da seguinte maneira: VP, VM, VG e VV são instâncias em que os raios e alturas são variáveis, considerando as subclasses de itens pequenos, médios, grandes e variáveis, respectivamente; RPAP, RPAM, RPAG e RPAV são as instâncias em que os raios são únicos e pequenos, e as alturas variam entre pequena, média, grande e variável, respectivamente; o que funciona de modo análogo para as instâncias de raio único médio – RMAP, RMAM, RMAG e RMAV – e para as instâncias de raio único grande – RGAP, RGAM, RGAG e RGAV. Algumas instâncias foram criadas para o caso em que a altura é única. Neste trabalho, estamos considerando apenas o caso em que a altura é única e pequena e os raios variam entre pequenos, médios e grandes, sendo respectivamente, APRP, APRM, APRG.

Para cada uma destas classes, foram geradas instâncias aleatoriamente. Essas instâncias foram geradas com uma quantidade de itens que varia entre 4, 6, 8 e 10. Os resultados serão apresentados de acordo com o número de itens para cada instância criada.

## 5.2. Resultados numéricos

Nas Tabelas 1, 2, 3, 4 e 5 são apresentados os resultados numéricos para o problema de empacotamento de cilindros em níveis considerando os itens de altura e raio variáveis, raio pequeno fixo, raio médio fixo, raio grande fixo e altura pequena fixa, respectivamente. Na Tabela 1, é apresentado primeiramente, o nome da instância resolvida, seguido pelo *status* da solução, apresentado na segunda coluna. A terceira coluna apresenta a quantidade de itens da instância e a coluna quatro, a quantidade de recipientes usada. A coluna cinco apresenta a quantidade de níveis usada em cada recipiente. Note que, quando a quantidade de recipientes é superior a uma unidade, o número de níveis usados em cada um desses recipientes está separado por /. Por fim, o tempo gasto, em segundos, para encontrar a solução é apresentado na sexta coluna. As Tabelas 2, 3, 4 e 5, apresentam uma coluna a mais, onde é indicado o valor do raio único em centímetros, ou, no caso da Tabela 5, o valor da altura em centímetros. Foi estabelecido um tempo máximo de 3600 segundos. A sigla TL presente em algumas tabelas indica que esse tempo limite foi alcançado.

Tabela 1: Resultados numéricos obtidos para os itens de alturas e raios variáveis.

Instância	Solução	Quantidade de itens	Quantidade de recipientes	Quantidade de níveis	Tempo (s)
VP1	Ótima	4	1	1	5,42
VP2	Ótima	6	1	1	5,50
VP3	Ótima	8	1	1	5,61
VP4	Ótima	10	1	1	5,82
VM1	Ótima	4	1	2	5,73
VM2	Ótima	6	1	2	6,18
VM3	Ótima	8	1	2	39,48
VM4	Factível	10	2	3/2	TL
VG1	Ótima	4	4	1/1/1/1	6,32
VG2	Ótima	6	6	1/1/1/1/1/1	182,15
VG3	Factível	8	8	1/1/1/1/1/1/1/1	TL
VG4	Factível	10	10	1/1/1/1/1/1/1/1/1/1	TL
VV1	Ótima	4	1	1	5,51
VV2	Factível	6	2	3/2	TL
VV3	Ótima	8	2	2/2	276,49
VV4	Factível	10	8	1/1/1/1/2/1/1/1	TL

A Tabela 1 apresenta os resultados para o caso em que a altura e o raio dos itens são variáveis. Notamos que foi encontrada a solução ótima para 11 das 16 instâncias avaliadas. Para as outras 5 instâncias foram encontradas soluções factíveis, mas não foi possível provar que se trata da solução ótima, visto que o tempo limite de 3600 segundos foi alcançado. Essas instâncias apresentam um grau de dificuldade maior, pois seus itens apresentam uma variação de tamanho





muito grande tanto em relação à altura quanto ao raio. Dessa forma, já era esperado que o tempo de execução fosse relativamente elevado, principalmente para aquelas instâncias com mais itens.

Tabela 2: Resultados numéricos obtidos para os itens de raio fixo pequeno.

Instância	Solução	Raio (cm)	Quantidade de itens	Quantidade de recipientes	Quantidade de níveis	Tempo (s)
RPAP1	Ótima	10	4	1	1	7,03
RPAP2	Ótima	10	6	1	1	5,51
RPAP3	Ótima	10	8	1	1	5,94
RPAP4	Ótima	10	10	1	1	6,06
RPAM1	Ótima	10	4	1	1	5,56
RPAM2	Ótima	10	6	1	1	5,83
RPAM3	Ótima	10	8	1	1	5,97
RPAM4	Ótima	10	10	1	1	6,17
RPAG1	Ótima	10	4	1	1	5,64
RPAG2	Ótima	10	6	1	1	5,75
RPAG3	Ótima	10	8	1	1	5,81
RPAG4	Ótima	10	10	1	1	7,10
RPAV1	Ótima	10	4	1	1	5,54
RPAV2	Ótima	10	6	1	1	5,68
RPAV3	Ótima	10	8	1	1	5,96
RPAV4	Ótima	10	10	1	1	6,46

Na Tabela 2, são apresentados os resultados para as instâncias em que o raio é fixo e pequeno. Podemos observar que o modelo consegue provar a otimalidade para todas as instâncias propostas. A média de tempo computacional para resolução das instâncias foi de 6,39 segundos, sendo, no pior caso, 7,10 segundos. Para esse caso, a otimalidade é provada mais rapidamente porque cilindros de raio pequeno podem ser facilmente alocados em um único forno.

Tabela 3: Resultados numéricos obtidos para os itens de raio fixo médio.

Instância	Solução	Raio (cm)	Quantidade de itens	Quantidade de recipientes	Quantidade de níveis	Tempo (s)
RMAP1	Ótima	20	4	1	2	5,83
RMAP2	Ótima	20	6	1	2	6,26
RMAP3	Ótima	20	8	1	3	9,84
RMAP4	Ótima	20	10	1	4	30,08
RMAM1	Ótima	20	4	1	2	5,65
RMAM2	Ótima	20	6	1	2	6,42
RMAM3	Ótima	20	8	1	2	80,86
RMAM4	Factível	20	10	2	1/2	TL
RMAG1	Ótima	20	4	1	1	5,89
RMAG2	Ótima	20	6	2	1/1	6,50
RMAG3	Ótima	20	8	3	1/1/1	17,86
RMAG4	Ótima	20	10	3	1/1/1	41,73
RMAV1	Ótima	20	4	1	1	6,60
RMAV2	Ótima	20	6	1	2	11,14
RMAV3	Ótima	20	8	1	3	452,06
RMAV4	*	20	10	*	*	TL

Analisando a Tabela 3, que apresenta os resultados para instâncias de raio médio fixo, podemos perceber que o programa encontrou solução ótima para 14 das 16 instâncias. Para essas instâncias, o tempo médio de resolução foi de 49,05 segundos. Podemos perceber que o tempo



aumentou em relação às instâncias em que o raio é único e pequeno. Isso acontece porque aumenta a dificuldade de resolução, visto que o raio, agora, ocupa uma porção maior do recipiente. No caso da instância RMAM4, foi encontrada uma solução factível, mas não foi provada sua otimalidade, pois alcançou o tempo limite de 3600 segundos. Para a instância RMAV4, marcada com \* nas colunas da solução e quantidade de recipientes, o programa não conseguiu encontrar nenhuma solução factível durante o tempo de 3600 segundos. Isso acontece porque o ponto inicial é gerado pelo próprio *solver*, então, nem sempre é garantido que será possível encontrar uma solução factível no tempo delimitado. A Figura 2 ilustra a solução obtida para a instância RMAM3. Sabemos que os itens são cilíndricos, então, a ilustração é dividida em face superior, dada por itens circulares e eixo de coordenadas  $xy$ , e face lateral, dada por itens retangulares e eixo de coordenadas  $xz$ , para melhor compreensão em relação à disposição dos itens no recipiente. São usados diferentes tons de cinza na face superior para indicar que os itens são de alturas diferentes. Dessa forma, quanto mais alto, mais escuro é o item. Os diferentes retângulos usados na face lateral também indicam a altura de cada um dos itens. A linha tracejada representa os itens que, na visão dessa face, estão “atrás” de itens mais altos.

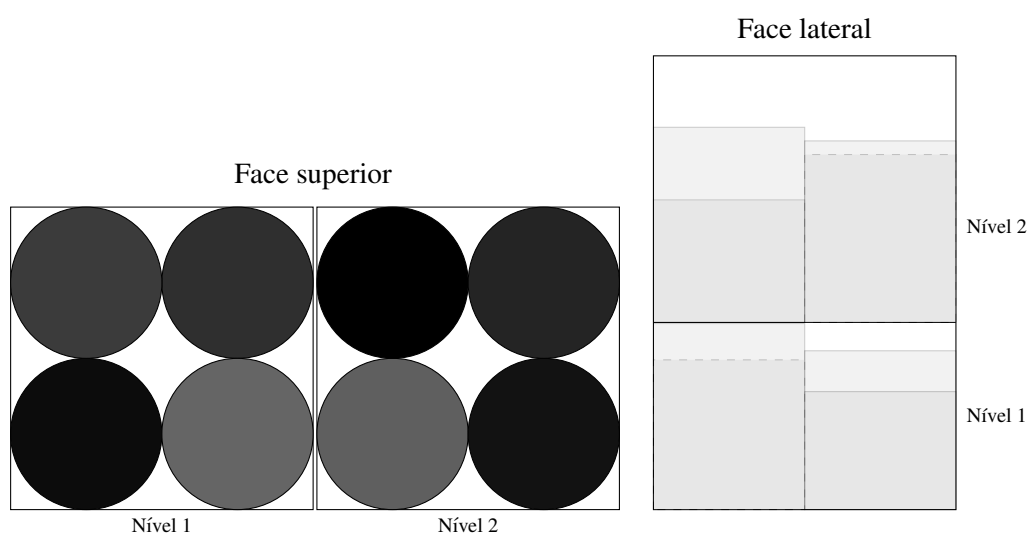


Figura 2: Exemplo de solução obtida para a instância RMAM3.

A Tabela 4 apresenta os resultados obtidos para as instâncias em que o raio é grande e fixo. Na tabela, foram encontradas as soluções ótimas para 11 das 16 instâncias analisadas. Para cinco instâncias não foi possível encontrar uma solução factível no tempo limite de 3600 segundos.

As instâncias da Tabela 5 foram geradas na tentativa de avaliar superficialmente o caso em que as alturas são únicas e os raios são variáveis e, assim, obter resultados parciais para o estudo em questão. Foram encontradas soluções ótimas para 10 das 12 instâncias, em um tempo de execução de 211,88 segundos, no pior caso. As outras duas instâncias apresentam soluções factíveis, mas não foram provadas suas otimalidades.

De modo geral, podemos notar que o tempo computacional gasto para resolução do modelo é proporcional à quantidade de itens, para cada subclasse das instâncias geradas. Além disso, a dimensão do raio também influencia bastante no aumento do tempo de execução. Nos casos em que o tempo limite foi alcançado, uma solução factível foi encontrada, porém não foi provada a otimalidade da mesma. Em alguns casos, não foi possível encontrar uma solução factível no tempo estabelecido pois não estamos fornecendo um ponto inicial para o *solver*. Dessa forma, estudar maneiras de fornecer esse ponto inicial pode melhorar os resultados obtidos.



Tabela 4: Resultados numéricos obtidos para os itens de raio fixo grande.

Instância	Solução	Raio (cm)	Quantidade de itens	Quantidade de recipientes	Quantidade de níveis	Tempo (s)
<b>RGAP1</b>	Ótima	40	4	2	2/2	10,33
<b>RGAP2</b>	Ótima	40	6	2	3/4	1387,90
<b>RGAP3</b>	*	40	*	*	*	TL
<b>RGAP4</b>	Ótima	40	10	2	5/5	219,12
<b>RGAM1</b>	Ótima	40	4	2	3/3	7,71
<b>RGAM2</b>	Ótima	40	6	3	2/2/1	1063,22
<b>RGAM3</b>	*	40	*	8	*	TL
<b>RGAM4</b>	Ótima	40	10	10	1/1/1/1/1/1/1/1/1/1	329,00
<b>RGAG1</b>	Ótima	40	4	4	1/1/1/1	6,08
<b>RGAG2</b>	Ótima	40	6	6	1/1/1/1/1/1	10,32
<b>RGAG3</b>	Ótima	40	8	8	1/1/1/1/1/1/1/1	72,93
<b>RGAG4</b>	Ótima	40	10	10	1/1/1/1/1/1/1/1/1/1	166,83
<b>RGAV1</b>	Ótima	40	4	2	1/3	37,32
<b>RGAV2</b>	*	40	6	*	*	TL
<b>RGAV3</b>	*	40	8	*	*	TL
<b>RGAV4</b>	*	40	10	*	*	TL

Tabela 5: Resultados numéricos obtidos para os itens de altura fixa pequena.

Instância	Solução	Altura (cm)	Quantidade de itens	Quantidade de recipientes	Quantidade de níveis	Tempo (s)
<b>APRP1</b>	Ótima	20	4	1	1	5,41
<b>APRP2</b>	Ótima	20	6	1	1	5,45
<b>APRP3</b>	Ótima	20	8	1	1	5,63
<b>APRP4</b>	Ótima	20	10	1	1	5,98
<b>APRM1</b>	Ótima	20	4	2	1/1	5,98
<b>APRM2</b>	Ótima	20	6	1	2	6,19
<b>APRM3</b>	Ótima	20	8	1	3	13,04
<b>APRM4</b>	Ótima	20	10	1	3	65,02
<b>APRG1</b>	Ótima	20	4	1	4	6,65
<b>APRG2</b>	Ótima	20	6	2	2/1/3	211,88
<b>APRG3</b>	Factível	20	8	3	1/3/4	TL
<b>APRG4</b>	Factível	20	10	6	2/1/2/3/2/3	TL

## 6. Conclusões e trabalhos futuros

Neste trabalho apresentamos um modelo de programação não-linear inteira mista para o problema de empacotamento de cilindros em níveis. O modelo proposto mostrou-se adequado, principalmente para a resolução de instâncias pequenas e de raio único pequeno.

Como próximos passos da pesquisa, pretendemos aumentar a quantidade de itens para cada classe de instâncias criada, encontrando a quantidade máxima de itens que o modelo é capaz de resolver na otimalidade, em um tempo de execução de até 3600 segundos. Pretendemos ainda, explorar melhor o caso em que a altura dos itens é única e os raios são variáveis e estudar uma maneira de fornecer o ponto inicial para o *solver*.

## Agradecimentos

Agradecemos o apoio da FAPESP (processos 2016/02469–4, 2015/24987–4 e 2013/07375–0) para realização deste trabalho.



## Referências

- Birgin, E. G. e Gentil, J. M. (2010). New and improved results for packing identical unitary radius circles within triangles, rectangles and strips. *Computers & Operations Research*, 37(7):1318–1327.
- Chung, F. R., Garey, M. R., e Johnson, D. S. (1982). On packing two-dimensional bins. *SIAM Journal on Algebraic Discrete Methods*, 3(1):66–76.
- Coffman, E. G., Jr, Garey, M. R., Johnson, D. S., e Tarjan, R. E. (1980). Performance bounds for level-oriented two-dimensional packing algorithms. *SIAM Journal on Computing*, 9(4):808–826.
- Fowler, R. J., Paterson, M., e Tanimoto, S. L. (1981). Optimal packing and covering in the plane are np-complete. *Information Processing Letters*, 12(3):133–137.
- Frenk, J. G. e Galambos, G. (1987). Hybrid next-fit algorithm for the two-dimensional rectangle bin-packing problem. *Computing*, 39(3):201–217.
- George, J. A., George, J. M., e Lamar, B. W. (1995). Packing different-sized circles into a rectangular container. *European Journal of Operational Research*, (84):693–712.
- Gilmore, P. e Gomory, R. E. (1965). Multistage cutting stock problems of two and more dimensions. *Operations research*, 13(1):94–120.
- Gomes, A. M. e Oliveira, J. F. (2002). A 2-exchange heuristic for nesting problems. *European Journal of Operational Research*, 141(2):359 – 370.
- Kang, J. e Park, S. (2003). Algorithms for the variable sized bin packing problem. *European Journal of Operational Research*, 147(2):365–372.
- Litvinchev, I. e Espinosa, E. L. O. (2014). Integer programming formulations for approximate packing circles in a rectangular container. *Mathematical Problems in Engineering*, 2014:1–6.
- Lodi, A., Martello, S., e Monaci, M. (2002a). 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. (2002b). Recent advances on two-dimensional bin packing problems. *Discrete Applied Mathematics*, 123(1):379–396.
- Ortmann, F. G., Ntene, N., e Van 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(2):306–315.
- Puchinger, J. e Raidl, G. R. (2007). Models and algorithms for three-stage two-dimensional bin packing. *European Journal of Operational Research*, 183(3):1304–1327.
- Stoyan, Y., Semkin, V., e Chugay, A. (2016). Modeling close packing of 3d objects. *Cybernetics and Systems Analysis*, 52(2):296–304.
- Stoyan, Y. G. e Yas'kov, G. (1998). Mathematical model and solution method of optimization problem of placement of rectangles and circles taking into account special constraints. *International Transactions in Operational Research*, 5(1):45–57.
- Zhao, X., Bennell, J. A., Bektaş, T., e Dowsland, K. (2014). A comparative review of 3d container loading algorithms. *International Transactions in Operational Research*, 23(1-2):287–320.