# Novos Algoritmos de Aproximação para Problemas de Empacotamento Bidimensional

**Glauber F. Cintra**[1]

**Resumo**

Investigamos o problema de empacotamento bidimensional (PEB) e introduzimos o conceito de padrões "semi-homogêneos". Fazendo uso de tais padrões desenvolvemos um algoritmo polinomial cuja razão de aproximação absoluta é 4, e mostramos que esta razão é "justa". Ainda utilizando padrões semi-homogêneos, desenvolvemos um algoritmo que resolve uma variante do PEB na qual os recipientes e os itens são quadrados. Provamos que esse algoritmo tem razão de aproximação assintótica entre 2,4166 e 2,6875. Até onde sabemos, esses são os primeiros algoritmos de aproximação propostos para tais problemas. Desenvolvemos ainda um algoritmo para o PEB binário com rotações e provamos que esse algoritmo possui razão de aproximação assintótica não maior que 4.

**Palavras-chave:** problemas de empacotamento, algoritmos de aproximação, razão assintótica

**Abstract**

We investigate the two-dimensional packing problem (PEB) and we introduce the concept of "semi-homogeneous" patterns. Making use of such patterns we developed a polynomial 4-approximation algorithm for this problem, and we prove that this absolute performance ratio is "tight". Using such patterns, we design an algorithm for a variant of the PEB, in which the bins and the items are squares. We prove that this algorithm has an asymptotic performance ratio between 2,4166 and 2,6875. To our knowledge, these are the first approximation algorithms proposed for these problems. We also exhibit an algorithm for the binary PEB with rotations and we prove that its asymptotic performance ratio is at most 4.

**Keywords:** packing problems, two-dimensional packing, approximation algorithms, asymptotic ratio

[1]Faculdade 7 de Setembro — Rua Alm. Maximiano da Fonseca 1395 GP 15 — CEP 60811-020 — Fortaleza, CE, glauber@fa7.edu.br

# Introdução

Existe uma grande diversidade de situações em que nos deparamos com o seguinte desafio: colocar uma coleção de objetos pequenos (itens) dentro de objetos grandes (recipientes), obtendo algum tipo de vantagem econômica. Frequentemente os recipientes e os itens têm apenas duas dimensões relevantes e possuem forma retangular. Os problemas que envolvem tais restrições são chamados genericamente de problemas de empacotamento bidimensional (PEB). Tais problemas constituem o assunto principal desse artigo.

Desenvolvemos algoritmos de aproximação para algumas variantes do PEB, fornecendo suas respectivas razões de aproximação. Até onde sabemos, alguns deles são os primeiros algoritmos de aproximação desenvolvidos para tais variantes.

Este texto está organizado da seguinte maneira. Na próxima seção procuramos despertar o interesse do leitor pelo estudo dos problemas de empacotamento. Apresentamos motivações de caráter teórico, citando algumas conexões com outras áreas de pesquisa e a complexidade computacional intrínseca a esses problemas. Além disso, apresentamos alguns resultados de inaproximabilidade relacionados com alguns problemas de empacotamento. Fornecemos também motivações de ordem prática, apresentando diversas situações em que problemas de empacotamento surgem naturalmente.

A seguir, na Seção 1, introduzimos alguns conceitos sobre algoritmos de aproximação e estabelecemos parte da notação que será utilizada nas seções seguintes.

Na Seção 3, abordamos PEB. Introduzimos o conceito de padrões semi-homogêneos e mostramos como utilizá-los no desenvolvimento de um algoritmo de aproximação para este problema cuja razão de aproximação absoluta é igual a 4. Propomos também um algoritmo de aproximação para a variante do PEB em que os recipientes e os itens são quadrados. Mostramos que tal algoritmo tem razão de aproximação assintótica não maior que 2,6875. Propomos ainda um algoritmo de aproximação para o PEB binário com rotações e mostramos que sua razão de aproximação assintótica não é maior que 4.

Na última seção, tecemos algumas considerações sobre nossas contribuições e discutimos também alguns possíveis desdobramentos de nossa pesquisa.

# 1 Preliminares

Os conceitos aqui explicados estão baseados em [GJ79] e [FMC$^+$01]. Dado um problema de minimização $\mathcal{P}$, denotamos por $\mathrm{OPT}(I)$ o valor de uma solução ótima de uma instância $I$ de $\mathcal{P}$. Dado um algoritmo $\mathcal{A}$ para $\mathcal{P}$, denotamos por $\mathcal{A}(I)$ o valor obtido pelo algoritmo $\mathcal{A}$ para a instância $I$ de $\mathcal{P}$.

Dizemos que um algoritmo $\mathcal{A}$ é um *algoritmo de aproximação* para um problema de minimização $\mathcal{P}$ se $\mathcal{A}$ é polinomial e existem funções $\alpha$ e $\beta$ tais que

$$\mathcal{A}(I) \leq \alpha\ \mathrm{OPT}(I) + \beta, \forall\ I \in \mathcal{P}. \tag{1}$$

A *razão de aproximação absoluta* do algoritmo $\mathcal{A}$, denotada por $R_{\mathcal{A}}$, é definida da seguinte forma:

$$R_{\mathcal{A}} = inf\{r \mid \frac{\mathcal{A}(I)}{\mathrm{OPT}(I)} \leq r, \forall\ I \in \mathcal{P}\}.$$

Se em (1) temos que $\beta = 0$, então dizemos que o algoritmo $\mathcal{A}$ é uma $\alpha$-aproximação absoluta para o problema $\mathcal{P}$ e que $\alpha$ é *uma* razão de aproximação absoluta do algoritmo $\mathcal{A}$. Dizemos que a razão $\alpha$ é *justa* se para todo $\epsilon > 0$, existe uma instância $I \in \mathcal{P}$ tal que $\frac{\mathcal{A}(I)}{\mathrm{OPT}(I)} > \alpha - \epsilon$. Neste caso, temos que $R_{\mathcal{A}} = \alpha$.

Definimos a *razão de aproximação assintótica* do algoritmo $\mathcal{A}$, denotada por $R^{\infty}_{\mathcal{A}}$, da seguinte maneira[2]:

$$R^{\infty}_{\mathcal{A}} = \limsup_{n \to \infty} (max\{\frac{\mathcal{A}(I)}{\mathrm{OPT}(I)} \mid I \in \mathcal{P} \text{ e } \mathrm{OPT}(I) = n\}).$$

Se em (1) temos que $\beta$ é uma constante, dizemos que o algoritmo $\mathcal{A}$ é uma $\alpha$-aproximação assintótica para o problema $\mathcal{P}$ e que $\alpha$ é *uma* razão de aproximação assintótica do algoritmo $\mathcal{A}$. Dizemos que a razão $\alpha$ é *justa* se para todo $\epsilon > 0$ e todo $n > 0$, existe uma instância $I \in \mathcal{P}$ tal que $\mathrm{OPT}(I) > n$ e $\frac{\mathcal{A}(I)}{\mathrm{OPT}(I)} > \alpha - \epsilon$. Neste caso, temos que $R^{\infty}_{\mathcal{A}} = \alpha$.

Chamamos de *esquema de aproximação assintótica de tempo polinomial* (*polynomial time asymptotic approximation scheme*), ou simplesmente PAAS, um conjunto de algoritmos $\mathcal{A}$ tal que, para todo $\epsilon > 0$, existe uma constante $\beta$ e um algoritmo $\mathcal{A}_\epsilon \in \mathcal{A}$, polinomial em $\epsilon$ e no tamanho da entrada, tal que

[2]Esta definição aparece em [CGJ82].

$$\mathcal{A}_\epsilon(I) \leq (1+\epsilon)\ \mathrm{OPT}(I) + \beta.$$

Chamamos de *esquema de aproximação assintótica de tempo completamente polinomial* (*fully polynomial time asymptotic approximation scheme*), ou simplesmente FPAAS, um conjunto de algoritmos $\mathcal{A}$ tal que, para todo $\epsilon > 0$, existe uma constante $\beta$ e um algoritmo $\mathcal{A}_\epsilon \in \mathcal{A}$, polinomial em $\frac{1}{\epsilon}$ e no tamanho da entrada, tal que

$$\mathcal{A}_\epsilon(I) \leq (1+\epsilon)\ \mathrm{OPT}(I) + \beta.$$

Todos os algoritmos de aproximação discutidos nesse artigo são para problemas de minimização. Por este motivo, não fornecemos as definições correspondentes para problemas de maximização. Tais definições podem ser obtidas em [GJ79].

# 2 Motivação

Devido à grande variedade de situações do mundo real onde surgem problemas de empacotamento, um número crescente de pesquisadores de diversas áreas, tais como computação, economia, engenharia e matemática, têm se dedicado ao estudo desses problemas. Nesses estudos têm sido aplicadas diversas técnicas de resolução de problemas tais como *programação linear*, *programação dinâmica*, *branch-and-bound* etc. Além disso, avanços significativos têm sido obtidos em outras áreas do conhecimento como fruto de pesquisas envolvendo problemas de empacotamento. Destacamos em especial os avanços obtidos nas áreas de teoria da complexidade computacional e algoritmos de aproximação.

O próprio termo *approximation algorithm* foi introduzido por Johnson [Joh74] ao propor algoritmos para o problema de empacotamento unidimensional (PEU). Os primeiros resultados provando a inexistência, sob a hipótese de que $\mathcal{P} \neq \mathcal{NP}$, de algoritmos *on-line* com razão de aproximação menor que certas constantes, envolveram problemas de empacotamento. Além disso, os primeiros PAAS e FPAAS para problemas fortemente $\mathcal{NP}$-completos foram desenvolvidos para o PEU [FL81, KK82].

Os problemas de empacotamento costumam ser fáceis de entender e formular. No entanto, sua aparente simplicidade costuma esconder sua natureza complexa, em termos computacionais. Queremos dizer com isto que a maioria dos problemas de empacotamento tratados na literatura não podem ser resolvidos por algoritmos polinomiais, a menos que $\mathcal{P} = \mathcal{NP}$. Muitos problemas de empacotamento têm como caso particular o problema da 3-partição (P3P).

O P3P é o seguinte problema: dado um número inteiro $B$ e um conjunto $A$ com $3m$ elementos, onde cada elemento $a \in A$ possui valor $\frac{B}{4} < v_a < \frac{B}{2}$, e $\sum_{a\in A} v_a = mB$, queremos particionar $A$ em subconjuntos disjuntos $A_1, \ldots, A_m$ tais que $\sum_{a\in A_i} v_a = B$ $(i = 1, \ldots, m)$. Garey e Johnson [GJ75] provaram que o P3P é $\mathcal{NP}$-difícil. É fácil mostrar que o P3P é um caso particular do PEU, que, obviamente, é um caso particular do PEB. Sendo assim, o PEB também é $\mathcal{NP}$-difícil.

Sabe-se que algumas variantes dos problemas de empacotamento não são aproximáveis, em termos absolutos, abaixo de certas constantes, supondo que $\mathcal{P} \neq \mathcal{NP}$. Por exemplo, Garey Johnson [GJ79] provaram que não existe algoritmo de aproximação para o PEU com razão de aproximação absoluta menor que 1,5. Brown, Baker e Katseff [BBK82] mostraram que qualquer algoritmo *on-line* para o problema de empacotamento em faixa binário (PEFB) tem que ter razão de aproximação absoluta maior ou igual a 2.

Existem ainda resultados envolvendo inaproximabilidade, em termos assintóticos, de problemas de empacotamento. Por exemplo, Van Vliet [Vli92] provou que não existe algoritmo *on-line* para o PEU com razão de aproximação assintótica menor que 1,5401. Csirik e Woeginger [CW97] demonstraram que qualquer algoritmo *on-line* para o PEFB, baseado em níveis, tem que ter razão de aproximação assintótica maior ou igual a 1,691. Sabe-se ainda que qualquer algoritmo *on-line* para o PEB binário tem que ter razão de aproximação assintótica maior ou igual a 1,907 [BvW96].

O interesse por problemas de empacotamento é também explicado por sua grande aplicabilidade prática, especialmente nas indústrias. Pequenas melhorias nos processos que envolvem empacotamento podem levar a ganhos substanciais, dependendo da escala de produção, e representar uma vantagem decisiva na competição com outras empresas do setor.

As indústrias alimentícias, farmacêuticas, de cosméticos, de bens de consumo duráveis etc., precisam empacotar seus produtos, geralmente em caixas de papelão, e guardá-los em armazéns. As empresas de transporte rodoviário, ferroviário, marítimo e aéreo precisam colocar as cargas em contêineres e caminhões-baú. Estes contêineres muitas vezes precisam ser empilhados em navios. Em todas estas situações surgem problemas de empacotamento bi e tridimensional.

Existe ainda uma infinidade de aplicações práticas para os problemas de empacotamento. Por exemplo, determinar como arranjar os processos na memória principal de um computador de modo a diminuir a necessidade de fazer *swap* entre a memória principal e a memória secundária envolve um problema de empacotamento. Um outro exemplo seria o planejamento da sequência de exibição de anúncios durante os intervalos comerciais na programação de uma emissora de rádio ou televisão. Enfim, existe um largo espectro de aplicações práticas para os problemas de empacotamento.

# 3 Novos Algoritmos de Aproximação

Nesta seção propomos algoritmos de aproximação para o problema de empacotamento bidimensional (PEB). Vejamos a definição deste problema: dada uma quantidade ilimitada de recipientes retangulares de largura $L$ e altura $A$, e uma lista de $m$ itens retangulares, cada item $i$ com largura $l_i \leq L$, altura $a_i \leq A$ e demanda $d_i$, queremos determinar como empacotar $d_i$ cópias de cada item $i$ da lista utilizando o menor número possível de recipientes.

Apresentaremos também um algoritmo de aproximação para a variante onde todas os recipientes e itens são quadrados, que é chamada de problema de empacotamento de quadrados em quadrados (PEQ). Até onde sabemos, esses são os primeiros algoritmos de aproximação propostos para estes problemas. Finalmente, propomos um algoritmo de aproximação para a variante do PEB na qual rotações são permitidas e todos os itens possuem demanda igual a 1, que chamaremos de PEBB$^r$.

Ao projetar um algoritmo para o PEB, temos que ter cuidado com a representação da solução gerada pelo algoritmo. Para descrever eficientemente uma solução, precisamos nos limitar a uma quantidade polinomial de padrões[3] e especificar a quantidade de vezes que cada padrão deve ser utilizado. Mas isto ainda não é suficiente. Observe que, se as dimensões dos itens forem muito pequenas em relação às dimensões dos recipientes, poderemos ter num padrão uma quantidade exponencial de retângulos. Se, ao descrever um padrão, precisarmos gastar memória para representar cada item contido no padrão, poderemos precisar de uma quantidade exponencial de memória para representar o padrão, e então o algoritmo precisaria de tempo exponencial.

Portanto, qualquer algoritmo polinomial para o PEB tem que satisfazer às seguintes exigências: (a) utilizar uma quantidade polinomial de padrões (e especificar a quantidade de vezes que cada padrão deve ser utilizado) e (b) representar cada padrão com uma quantidade polinomial

[3] Chamamos de *padrão de empacotamento* (ou simplesmente *padrão*) cada possível forma de empacotar itens num recipiente.

de memória.

Para atender estes dois requisitos podemos fazer uso de padrões homogêneos. Dizemos que um padrão é *homogêneo* se ele contém apenas itens de mesmas dimensões. Denotamos por $H(L, A, l, a, d)$ o padrão homogêneo contendo $d$ cópias de um item de largura $l$ e altura $a$ num recipiente de largura $L$ e altura $A$. Empacotamos os itens em $H(L, A, l, a, d)$ da seguinte maneira. Fazemos $x = \lfloor \frac{L}{l} \rfloor$, para $j$ variando de 0 até $d-1$, e empacotamos a $j$-ésima cópia do item na posição $(l(j \bmod x), a\lfloor \frac{j}{x} \rfloor)$. Observe que o padrão terá $\lceil \frac{d}{x} \rceil$ níveis, cada nível contendo $x$ itens (exceto possivelmente o último).

Podemos representar padrões da forma $H(L, A, l, a, d)$ com uma quantidade de memória polinomial, visto que as posições dos retângulos no padrão não precisam ser armazenadas já que podem ser calculadas de acordo com o procedimento que acabamos de descrever. O algoritmo que chamaremos de H consiste em atender a demanda de cada item utilizando apenas padrões homogêneos. A seguir descrevemos o algoritmo H.

---

**Algoritmo 3.1** H

---

*Entrada*: Uma instância $I = (L, A, l, a, d)$ do PEB, onde $l = (l_1, \ldots, l_m)$,
$a = (a_1, \ldots, a_m)$ e $d = (d_1, \ldots, d_m)$.

*Saída*: Uma solução para $I$.

**1** Para $i = 1$ até $m$

**1.1** $x_i = \lfloor \frac{L}{l_i} \rfloor$, $y_i = \lfloor \frac{A}{a_i} \rfloor$ e $z_i = \lfloor \frac{d_i}{x_i y_i} \rfloor$.

**1.2** Se $z_i > 0$ então empacote $x_i y_i z_i$ cópias do item $i$ utilizando $z_i$ vezes o padrão $H(L, A, l_i, a_i, x_i y_i)$.

**1.3** Se $d_i > x_i y_i z_i$ então empacote $d_i - x_i y_i z_i$ cópias do item $i$ utilizando uma vez o padrão $H(L, A, l_i, a_i, d_i - x_i y_i z_i)$.

**2** Devolva o empacotamento.

---

Sobre a razão de aproximação desse algoritmo, vale o seguinte resultado[4].

**Teorema 3.1.** $\mathrm{H}(I) \leq 4\ \mathrm{OPT}(I) + m$, *para toda instância* $I$ *do PEB.*

Pelo Teorema 3.1 percebemos que o valor da solução produzida pelo algoritmo H pode ser bem distante do valor de uma solução ótima. Considere a seguinte família de instâncias: $m$ itens, todos eles com largura e altura (distintas) menores que 1 e demanda igual a 1, e

[4]Por questão de espaço, omitimos as provas dos teoremas. Tais provas podem ser obtidas em [Cin04].

recipientes de largura e altura $m$. Claramente é possível empacotar todos os itens utilizando um único recipiente. No entanto, o algoritmo H produz uma solução de valor $m$. Observe que $\lim_{m\to\infty} \frac{\mathrm{H}(I)}{\mathrm{OPT}(I)} = \infty$. Isto significa que o algoritmo H pode produzir uma solução de valor infinitamente maior que o valor de uma solução ótima.

De modo a obter um algoritmo com razão de aproximação absoluta igual a 4, introduzimos o conceito de padrões "semi-homogêneos". Chamamos de *bloco homogêneo* um retângulo minimal que contém todos os itens de mesmas dimensões dentro do padrão. Dizemos que um padrão bidimensional é *semi-homogêneo* se os blocos homogêneos contidos no padrão não se sobrepõem. Na Figura 1 temos à esquerda um padrão semi-homogêneo (a) e à direita um padrão que não é semi-homogêneo (b).

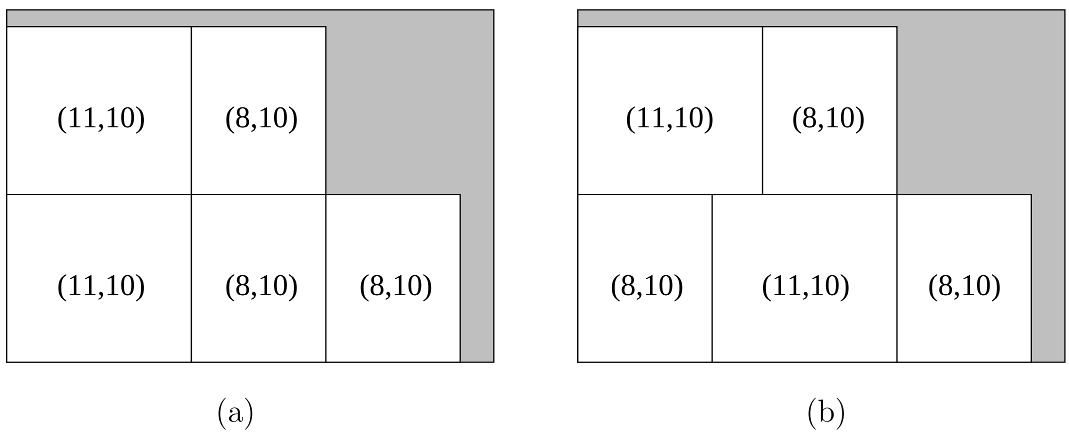


Figura 1: (a) Um padrão semi-homogêneo e (b) um padrão que não é semi-homogêneo

Observe que num padrão semi-homogêneo, cada bloco homogêneo é equivalente a um padrão homogêneo. Sendo assim, cada bloco homogêneo pode ser descrito com uma quantidade de memória polinomial. Ademais, a quantidade de blocos homogêneos num padrão semi-homogêneo é limitada por $m$, a quantidade de itens. Portanto, um padrão semi-homogêneo também pode ser descrito com uma quantidade de memória polinomial.

A idéia básica do algoritmo SH é primeiramente utilizar padrões homogêneos que aproveitem pelo menos $\frac{1}{4}$ da área do recipiente. A seguir, para cada item cuja demanda não tenha sido totalmente atendida com os padrões homogêneos, utilizamos no máximo 2 blocos homogêneos. Finalmente, estes blocos homogêneos são empacotados nos recipientes utilizando-se o algoritmo HFF, proposto por Chung, Garey e Johnson [CGJ82]. Note que teremos no máximo $2m$ blocos homogêneos. Isto garante que o HFF vai requerer tempo polinomial em $m$. Eis a descrição do algoritmo SH.

**Algoritmo 3.2** SH

*Entrada*: Uma instância $I = (L, A, l, a, d)$ do PEB, onde $l = (l_1, \ldots, l_m)$,

$a = (a_1, \ldots, a_m)$ e $d = (d_1, \ldots, d_m)$.

*Saída*: Uma solução para $I$.

**1** $n = 0$.

**2** Para $i = 1$ até $m$

**2.1** $x_i = \lfloor \frac{L}{l_i} \rfloor$, $y_i = \lfloor \frac{A}{a_i} \rfloor$ e $z_i = \lfloor \frac{d_i}{x_i y_i} \rfloor$.

**2.2** Se $z_i > 0$ então empacote $x_i y_i z_i$ cópias do item $i$ utilizando $z_i$ vezes o padrão $H(L, A, l_i, a_i, x_i y_i)$.

**2.3** $d'_i = d_i - x_i y_i z_i$.

**2.4** Se $l_i a_i d'_i \geq \frac{LA}{4}$ então empacote $d'_i$ cópias do item $i$ utilizando uma vez o padrão $H(L, A, l_i, a_i, d'_i)$.

**2.5** Se $d'_i > 0$ e $l_i a_i d'_i < \frac{LA}{4}$

**2.5.1** $y'_i = \lfloor \frac{d'_i}{x_i} \rfloor$, $n = n + 1$, $l'_n = l_i x_i$ e $a'_n = a_i y'_i$.

**2.5.2** Se $x_i y'_i < d'_i$ então $n = n + 1$, $l'_n = l_i(d'_i - x_i y'_i)$ e $a'_n = a_i$.

**3** Se $n > 0$ então resolva a instância $I' = (L, A, l' = (l'_1, \ldots, l'_n), a' = (a'_1, \ldots, a'_n))$ com o algoritmo HFF.

**4** Devolva o empacotamento.

---

O seguinte teorema estabelece a razão de aproximação absoluta do algoritmo SH.

**Teorema 3.2.** SH$(I) \leq 4$ OPT$(I)$, *para toda instância $I$ do PC. Ademais, esta razão é justa.*

Podemos introduzir algumas modificações no SH de modo a obter um algoritmo para o PEQ com razão de aproximação melhor que 4. Desta vez, objetivamos que a área ocupada em cada recipiente (com exceção de alguns deles) seja pelo menos $\frac{4L^2}{9}$. Chamamos de QSH o algoritmo específico para o PEQ. cuja descrição aparece a seguir.

Sobre o desempenho o desempenho assintótico do QSH, é possível provar o seguinte teorema.

**Teorema 3.3.** $2,4166 \approx \frac{29}{12} \leq R^{\infty}_{\text{QSH}} \leq \frac{43}{16} = 2,6875.$

Propomos ainda o algoritmo *First Fit Decreasing Height using Rotations* (FFDHR) para o problema de empacotamento bidimensional binário com rotações (PEBB$^r$). Neste problema, cada item possui demanda igual a 1. O FFDHR utiliza a estratégia do algoritmo FFDH, proposto por Coffman, Garey, Johnson e Tarjan [CGJT80], com duas modificações. Ao empacotar um item, procuramos o nível de menor índice no qual ele caiba, ou na sua orientação original,

**Algoritmo 3.3** QSH

*Entrada*: Uma instância $I = (L, l, d)$ do PEQ, onde $l = (l_1, \dots, l_m)$ e $d = (d_1, \dots, d_m)$.

*Saída*: Uma solução para $I$.

**1** $n = 1$.

**2** Para $i = 1$ até $m$

**2.1** $x_i = \lfloor \frac{L}{l_i} \rfloor$ e $z_i = \lfloor \frac{d_i}{x_i^2} \rfloor$.

**2.2** Se $z_i > 0$ então empacote $x_i^2 z_i$ cópias do item $i$ utilizando $z_i$ vezes o padrão $H(L, L, l_i, l_i, x_i^2)$.

**2.3** $d'_i = d_i - x_i^2 z_i$.

**2.4** Se $l_i^2 d'_i \geq \frac{4L^2}{9}$ empacote $d'_i$ cópias do item $i$ utilizando o padrão $H(L, A, l_i, a_i, d'_i)$.

**2.5** Se $d'_i > 0$ e $l_i^2 d'_i < \frac{4L^2}{9}$

**2.5.1** $y'_i = \lfloor \frac{d'_i}{x_i} \rfloor$.

**2.5.2** Se $y'_i > 0$ e $l_i x_i \geq \frac{L}{2}$ e $l_i y'_i \geq \frac{L}{2}$

**2.5.2.1** $l'_n = l'_{n+1} = l_i \lceil \frac{x_i}{3} \rceil$, $a'_n = a'_{n+1} = l_i \lceil \frac{y'_i}{2} \rceil$, $n = n + 2$.

**2.5.2.2** $l'_n = l'_{n+1} = l_i \lceil \frac{x_i}{3} \rceil$, $a'_n = a'_{n+1} = l_i \lfloor \frac{y'_i}{2} \rfloor$, $n = n + 2$.

**2.5.2.3** Se $x_i - 2\lceil \frac{x_i}{3} \rceil > 0$

**2.5.2.3.1** $l'_n = l'_{n+1} = l_i(x_i - 2\lceil \frac{x_i}{3} \rceil)$, $a'_n = l_i \lceil \frac{y'_i}{2} \rceil$, $a'_{n+1} = l_i \lfloor \frac{y'_i}{2} \rfloor$, $n = n + 2$.

**2.5.3** Se $y'_i > 0$ e $l_i x_i \geq \frac{L}{2}$ e $l_i y'_i < \frac{L}{2}$

**2.5.3.1** $l'_n = l'_{n+1} = l_i \lceil \frac{x_i}{3} \rceil$, $a'_n = a'_{n+1} = l_i$, $n = n + 2$.

**2.5.3.2** Se $x_i - 2\lceil \frac{x_i}{3} \rceil > 0$

**2.5.3.2.1** $l'_n = l_i(x_i - 2\lceil \frac{x_i}{3} \rceil)$, $a'_n = l_i$, $n = n + 1$.

**2.5.4** Se $x_i y'_i < d'_i$

**2.5.4.1** Se $l_i(d'_i - x_i y'_i) \geq \frac{L}{2}$

**2.5.4.1.1** $l'_n = l'_{n+1} = l_i \lceil \frac{d'_i - x_i y'_i}{3} \rceil$, $a'_n = a'_{n+1} = l_i$, $n = n + 2$.

**2.5.4.1.2** Se $x_i - 2\lceil \frac{d'_i - x_i y'_i}{3} \rceil > 0$

**2.5.4.1.2.1** $l'_n = l_i(x_i - 2\lceil \frac{d'_i - x_i y'_i}{3} \rceil)$, $a'_n = l_i$, $n = n + 1$.

**2.5.4.2** Se $l_i(d'_i - x_i y'_i) < \frac{L}{2}$

**2.5.4.2.1** $l'_n = l_i(d'_i - x_i y'_i)$, $a'_n = l_i$, $n = n + 1$.

**3** $n = n - 1$.

**4** Se $n > 0$, resolva a instância $I' = (L, L, l' = (l'_1, \dots, l'_n), a' = (a'_1, \dots, a'_n))$ com o algoritmo HFF.

**5** Devolva o empacotamento.

---

ou após ter sofrido uma rotação. Se existir tal nível, o item é empacotado nesse nível, dando prioridade para o uso da orientação original. Se não existir tal nível, é criado um novo nível, onde esse item é empacotado na sua orientação original.

A outra modificação é que os níveis são criados nos recipientes e não numa faixa de altura ilimitada, como no FFDH. Vamos considerar que os recipientes estão indexados a partir de 1. Suponha que ao empacotar um item, haja necessidade de criar um novo nível. Neste caso, procuramos o recipiente de menor índice no qual é possível criar um novo nível onde caiba o item na sua orientação original. Observe que somente realizamos uma rotação num item se for para evitar a criação de um novo nível. Apresentamos a seguir a descrição do FFDHR.

---

**Algoritmo 3.4** FFDHR

---

*Entrada*: Uma instância $I = (L, A, l, a)$ do PEBB$^r$, onde $l = (l_1, \ldots, l_m)$ e
$a = (a_1, \ldots, a_m)$.

*Saída*: Uma solução para $I$.

Ordene os itens em ordem decrescente de altura, obtendo $a_1 \geq \ldots \geq a_m$.

Faça $\mathcal{N} = 0$, $\mathcal{P} = 0$.

Para $i = 1$ até $m$

  $d = l_i$.

  $k = min(\{j \mid 1 \leq j \leq \mathcal{N} \text{ e } (l_i \leq w_j \text{ ou } (a_i \leq w_j \text{ e } l_i \leq h_j))\} \cup \{\mathcal{N} + 1\})$.

  Se $k \leq \mathcal{N}$ e $l_i > w_k$ faça $d = a_i$. `/* Item deve sofrer rotação */`

  Se $k > \mathcal{N}$ `/* É preciso criar um novo nível */`

    $r = min(\{j \mid 1 \leq j \leq \mathcal{P} \text{ e } t_j + a_i \leq A\} \cup \{\mathcal{P} + 1\})$.

    $p_k = r$, $w_k = L$.

    Se $r > \mathcal{P}$ `/* É preciso utilizar um novo recipiente */`

      $\mathcal{P} = \mathcal{P} + 1$ e $t_r = 0$.

    $h_k = l_i$, $b_k = t_r$ e $t_r = t_r + l_i$.

  Se $d = a_i$

    Empacote o item $i$ na posição $(L - w_k, b_k)$ do recipiente $p_k$, após o item sofrer rotação.

  Senão

    Empacote o item $i$ na posição $(L - w_k, b_k)$ do recipiente $p_k$.

  $w_k = w_k - d$.

Devolva o empacotamento.

---

Sobre a razão de aproximação assintótica do FFDHR, temos o seguinte resultado.

**Teorema 3.4.** FFDHR$(I) \leq 4$ OPT$(I) + 2$, *para toda instância $I$ do PEBB$^r$.*

# Considerações Finais

Em nossos estudos, percebemos que não existiam algoritmos de aproximação propostos na literatura para o PEB. É verdade que existem diversos algoritmos para o caso particular do PEB onde a demanda de todos os itens é igual a 1, no entanto estes algoritmos podem requerer tempo exponencial se usados para resolver instâncias do PEB.

Desenvolvemos então o algoritmo polinomial H, que utiliza apenas padrões homogêneos, e provamos que $\mathrm{H}(I) \leq 4\ \mathrm{OPT}(I) + m$, para toda instância $I$ do PC, onde $m$ representa a quantidade de itens da instância. No entanto, podemos construir instâncias com itens muito pequenos onde $\mathrm{H}(I)$ pode ser tão grande quanto se queira e $\mathrm{OPT}(I) = 1$. Disto decorre que $R_{\mathrm{H}}^{\infty} = \infty$. Introduzimos então o conceito de bloco homogêneo e, baseado neste conceito, definimos o que são padrões semi-homogêneos. Desenvolvemos o algoritmo SH, que utiliza padrões semi-homogêneos, e provamos que sua razão de aproximação absoluta é 4, e que esta razão é justa. Acreditamos ser o H e o SH os primeiros algoritmos de aproximação propostos para o PEB.

Ainda utilizando a idéia de padrões semi-homogêneos, desenvolvemos o algoritmo QSH, que resolve a variante do PEB na qual os recipientes itens são quadrados, que é chamada de PEQ. Provamos que o QSH tem razão de aproximação assintótica entre 2,4166 e 2,6875. Até onde sabemos, este é o primeiro algoritmo de aproximação proposto para o PEQ. Desenvolvemos ainda um algoritmo para o PEB binário com rotações, o FFDHR. Provamos que tal algoritmo possui razão de aproximação assintótica não maior que 4.

Diversas idéias parecem promissoras. Uma delas é modificar o algoritmo FFDHR da seguinte forma. Ao empacotar um item, podemos procurar um nível de menor altura e maior largura disponível no qual ele caiba (em alguma das orientações viáveis). Ao criar um novo nível, podemos procurar um recipiente de menor altura disponível no qual ele caiba (novamente em alguma das orientações viáveis). Provalmente este algoritmo teria desempenho melhor que o FFDHR, mas talvez determinar sua razão de aproximação (justa) seja uma tarefa bastante árdua.

Outra idéia que poderia ser explorada com o objetivo de aperfeiçoar o algoritmo QSH é subdividir a demanda residual de cada item de forma a obter blocos homogêneos quadrados. Cada um destes blocos homogêneos iria conter uma quantidade de itens que seria um quadrado perfeito. Observe que todo número inteiro $n$ é igual à soma de $k$ quadrados perfeitos, onde $k \leq \log n$. Isto garante que a instância residual teria tamanho polinomial em $m$. Poderíamos então resolvê-la com algoritmos que possuem melhor razão de aproximação que o HFF. Por exemplo, Kohayakawa, Miyazawa, Raghavan e Wakabayashi [KMRW01] apresentaram um algoritmo para a variante do PEQ na qual todos os itens têm demanda igual a 1. Tal algoritmo tem razão de aproximação assintótica tão próxima de 1,555... quanto se queira. Com a modificação que explicamos e utilizando o algoritmo de Kohayakawa *et al.* (ou mesmo o HFF) talvez seja possível melhorar a razão de aproximação do algoritmo QSH.

## Referências


[BBK82] D. J. Brown, B. S. Baker, and H. P. Katseff, *Lower bounds for the on-line two dimensional packing algorithms*, Acta Informatica **18** (1982), 207–225.

[BvW96] D. Blitz, A. van Vliet, and G. J. Woeginger, *Lower bounds on the asymptotic worst-case ratio of online bin packing algorithms*, Unpublished manuscript (1996).

[CGJ82] F. R. K. Chung, M. R. Garey, and D. S. Johnson, *On packing two-dimensional bins*, SIAM J. Algebraic Discrete Methods **3** (1982), 66–76.

[CGJT80] E. G. Coffman Jr., M. R. Garey, D. S. Johnson, and R. E. Tarjan, *Performance bounds for level oriented two-dimensional packing algorithms*, SIAM Journal on Computing **9** (1980), 808–826.

[Cin04] G. F. Cintra, *Algoritmos para problemas de corte de guilhotina bidimensional*, Ph.D. thesis, Instituto de Matemática e Estatística, São Paulo, 2004.

[CW97] János Csirik and Gerhard J. Woeginger, *Shelf algorithms for on-line strip packing*, Information Processing Letters **63** (1997), no. 4, 171–175.

[FL81] W. Fernandez de la Vega and G. S. Lueker, *Bin packing can be solved within 1+$\epsilon$ in linear time*, Combinatorica **1** (1981), 349–355.

[FMC+01] Cristina G. Fernandes, Flávio K. Miyazawa, Márcia Cerioli, Paulo Feofiloff, and et al., *Uma introdução sucinta a algoritmos de aproximação*, Publicações Matemáticas do IMPA. [IMPA Mathematical Publications], Instituto de Matemática Pura e Aplicada (IMPA), Rio de Janeiro, 2001, 23º Colóquio Brasileiro de Matemática. [23rd Brazilian Mathematics Colloquium].

[GJ75] M. R. Garey and D. S. Johnson, *Complexity results for multiprocessor scheduling under resource constraints*, SIAM J. Comput. **4** (1975), no. 4, 397–411.

[GJ79] M. R. Garey and D. S. Johnson, *Computers and intractability: A guide to the theory of $\mathcal{NP}$-completeness*, W. H. Freeman and Co., San Francisco, 1979.

[Joh74] D. S. Johnson, *Fast algorithms for bin packing*, J. Comput. Syst. Sci. **8** (1974), 272–314.

[KK82] N. Karmarkar and R. M. Karp, *An efficient approximation scheme for the one dimensional bin packing problem*, Proceedings, 23rd Ann. Symp. on Foundations of Computer Science (Los Angeles), IEEE Computer Society, 1982, pp. 312–320.

[KMRW01] Y. Kohayakawa, F.K. Miyazawa, P. Raghavan, and Y. Wakabayashi, *Multidimensional cube packing*, Brasilian Symposium on Graphs and Combinatorics. Electronic Notes of Discrete Mathematics (GRACO'2001), Elsevier Science, 2001.

[Vli92] A. Van Vliet, *An improved lower bound for online bin packing algorithms*, Inform. Process. Lett. **43** (1992), 277–284.