

# ***Simulated Annealing e Iterated Greedy para o Problema do Empacotamento Colorido***

**Daniela A. Sousa<sup>1</sup>**

<sup>1</sup>Departamento de Informática – Universidade Federal de Viçosa (UFV)

daniela.assis@ufv.br

## **1. Introdução**

O Problema do Empacotamento (BPP - *Bin Packing Problem*) é um problema NP-difícil clássico, amplamente estudado e consiste em alocar itens de diferentes tamanhos (1-dimensional) em uma quantidade mínima de recipientes, restrita a capacidade máxima dos recipientes. Uma das generalizações desse problema é o Problema do Empacotamento Colorido (CBPP - *Colorful Bin Packing Problem*), proposto em 2014 e estudado do ponto de vista de algoritmos *online* e algoritmos de aproximação [da Silva et al. 2022]. O CBPP consiste em, dado um conjunto de itens, cada um com tamanho e cor, e  $n$  recipientes de mesma capacidade, o objetivo é alocar todos os itens sem exceder a capacidade máxima dos recipientes e sem alocar itens de mesma cor lado a lado de forma que  $n$  seja mínimo.

O CBPP pode ser aplicado, por exemplo, para escalar músicas de diferentes gêneros em uma estação de rádio com vários canais [Böhm et al. 2014]. Suponha que uma estação de rádio não queira transmitir músicas do mesmo gênero uma após a outra. O CBPP pode ser aplicado para criar está escala de músicas, onde os itens são as músicas, as cores são os gêneros e os canais são os recipientes, sendo sua capacidade o tempo máximo de transmissão de músicas.

Como o CBPP é uma generalização do BPP, também é um problema NP-difícil. Uma abordagem comum para tratar problemas dessa classe é a utilização de meta-heurísticas. Dessa forma, a proposta desse trabalho é aplicar um método que integra as meta-heurísticas *Simulated Annealing* e *Iterated Greedy* para encontrar uma solução para o CBPP.

## **2. Trabalhos relacionados**

[Böhm et al. 2014] apresenta o problema *Online Colored Bin Packing* como uma generalização do problema *Online Black and White Bin Packing*. No problema *Online Black and White Bin Packing*, é dado uma lista de itens de tamanho no intervalo  $[0, 1]$ , onde cada item é preto ou branco. Os itens chegam um por um e devem ser alocados em um recipiente, sendo ordenados no recipiente por ordem de chegada, sem exceder sua capacidade e dois itens de mesma cor não devem ser colocados lado a lado. O objetivo é minimizar o número de recipientes utilizados.

[Borges and Schouery 2020] apresenta algoritmo pseudo polinomial para o CBPP baseado em modelos propostos para o BPP. Uma das propostas é o Fluxo em Arcos, que consiste no uso de um digrafo  $G' = (V', A')$ , sendo  $V'$  o conjunto de vértices que representa a união entre um único vértice de origem 0 e um conjunto de vértices  $V_q$ , que representa os pontos de empacotamento para cada item de cor  $q$  em um certo recipiente.

O conjunto de arcos  $A'$  representa a ligação de um vértice de cor  $q$  a um item de cor  $q'$ , onde  $q$  é diferente de  $q'$ . A solução para o CBPP é o menor número de caminhos que cobre os arcos de todos os itens.

[da Silva et al. 2022] propuseram quatro heurísticas para o CBPP, que posteriormente são melhoradas com VNS. Duas heurísticas são adaptações de heurísticas utilizadas para BPP: *Minimum Bin Slack'* (MBS') e *Best Fit Decreasing* (BFD). As duas outras heurísticas foram propostas para o CBPP: *Hard BFD* e *Two by Two*. Posteriormente, propuseram uma abordagem math-heurística dividida em três estágios: execução da VNS para encontrar bons padrões (recipiente que satisfaz as restrições de capacidade e de cores); resolução com relaxação linear para filtrar os padrões do estágio anterior e encontrar uma solução ótima; execução do GRASP por um determinado período de tempo, em que a solução ótima obtida com a relaxação é utilizada na construção de uma solução.

### 3. Metodologia

Formulando o CBPP, dado recipientes de capacidade  $L$  e um conjunto de itens  $I = \{1, \dots, n\}$ , onde cada item  $i$  possui um tamanho  $s_i$  maior que 0 e menor ou igual a  $L$  e uma cor  $c_i$ . O objetivo é empacotar cada item  $i$  pertencente a  $I$  nos recipientes  $\{B_1, B_2, \dots, B_k\}$ , com  $k$  mínimo de forma a satisfazer as restrições de capacidade e de cor. Para a restrição de capacidade, temos que a soma dos tamanhos dos itens no recipiente  $B_r$  deve ser menor ou igual a  $L$ , com  $r$  pertencente ao intervalo  $[1, k]$ . Já a restrição de cor é formulada como: existe uma permutação  $p$  dos itens em  $B_r$ , tal que para todos os pares de itens  $(i, j)$ , se  $c_i = c_j$ , então  $p(i) \neq p(j) + 1$ .

O primeiro passo para construção do método é a definição da função de avaliação. O objetivo em CBPP é minizar o número de recipientes utilizados para alocar todos os itens, porém avaliar a solução usando apenas o número de recipientes pode levar a estagnação, visto que pode existir diversas permutações com o mesmo número de recipientes. Dessa forma, foi utilizada a função de avaliação definida por [Hyde et al. 2011].

$$Fitness = 1 - \left( \frac{\sum_{i=1}^n (fullness_i / C)^2}{n} \right)$$

**Figura 1. Função de avaliação**

Temos que  $n$  é o número de recipientes,  $fullness_i$  é a soma de todos os itens no recipiente  $i$  e  $C$  é capacidade máxima dos recipientes. Essa função retorna valores entre zero e um, onde quanto mais próximo de zero, mais totalmente preenchido estão os recipientes. Ou seja, o objetivo é minizar a função de *Fitness*.

Para a construção da solução inicial, foi adotada a heurística *HardBFD* descrita por [da Silva et al. 2022], visto que foi proposta especificamente para o CBPP. Essa heurística consiste em processar os itens em ordem não-crescente de tamanho, onde o item  $i$  é alocado no recipiente que ficar com menor capacidade residual sem violar as restrições de cor. Se não for possível alocar em nenhum recipiente, procura-se um item  $j$  de cor diferente do item atual  $i$  e um recipiente  $r$  mais cheio no qual  $i$  e  $j$  podem ser

alocados juntos. Caso exista tal recipiente  $r$ , escolhe-se o  $j$  que irá gerar o recipiente mais preenchido. Caso contrário, é criado um novo recipiente, onde  $i$  será alocado.

A aplicação do *Simulated Annealing* foi feita com o intuito de minimizar a função de avaliação. Para isso, a vizinhança utilizada para BPP *Swap from Lowest Bin* descrita por [Hyde et al. 2011] foi adaptada para o CBPP. Nesta vizinhança, o maior item do recipiente menos preenchido é escolhido para troca com um item menor de um recipiente escolhido aleatoriamente se a troca não exceder a capacidade dos recipientes envolvidos e a respeitar a restrição de cor. Caso não exista tal item no recipiente escolhido aleatoriamente, busca-se em outro. Dessa forma, foi adotado *Preserving Strategies*, onde a operação de *swap* não será realizada se violar as restrições do problema.

Dos parâmetros utilizados para o *Simulated Annealing*, temos temperatura máxima (T\_MAX), temperatura mínima (T\_MIN) e taxa de resfriamento (R), onde testes foram realizados adotando valores expostos na Tabela 1, onde os valores de temperaturas adotadas visam analisar como o modelo se comporta ao variar entre intensificação e exploração.

Experimento	T_MAX	T_MIN	R
0	1000	0.000001	0.1
1	1000	0.000001	0.3
2	1000	0.000000001	0.1
3	1000	0.000000001	0.3
4	10000	0.000001	0.1
5	10000	0.000001	0.3
6	10000	0.000000001	0.1
7	10000	0.000000001	0.3
8	10000000	0.000001	0.1
9	10000000	0.000001	0.3
10	10000000	0.000000001	0.1
11	10000000	0.000000001	0.3

**Tabela 1. Parâmetros considerados para *Simulated Annealing***

Note que, ao utilizar essa vizinhança o número de recipientes permanece o igual ao número de recipientes da solução. Tentando contornar esse problema, é aplicada a meta-heurística *Iterated Greedy* (IG), onde todas as vezes que uma solução vizinha a solução corrente tem uma melhor avaliação que a melhor solução encontrada até aquele momento, a solução vizinha é adotada como melhor solução e é submetida ao IG. Se o número de recipientes diminuir após a execução do IG, a solução obtida é adotada como melhor solução encontrada e solução corrente. Na tentativa de diminuir o número de recipientes, aqueles com capacidade residual equivalente 10% ou mais da capacidade máxima são obrigatoriamente escolhidos para reconstrução. Além disso, no máximo, apenas outros 30% dos recipientes restantes são escolhidos aleatoriamente para reconstrução para evitar que bons padrões sejam destruídos. Os itens dos recipientes escolhidos são desalocados e os recipientes descartados. A reconstrução consiste em alocar o maior item ainda não alocado no primeiro recipiente em que as restrições de cor e capacidade sejam respeitadas, caso contrário, é criado um novo recipiente para que o item seja alocado. No caso do IG,

a avaliação é feita com base no número de recipientes e o critério de parada é número de iterações sem diminuição do número de recipientes, adotando o valor de 2000 iterações.

Os casos de teste foram adaptados de instâncias para o BPP geradas aleatoriamente (Randomly Generated Instances - BPP)<sup>1</sup>. Para cada instância, na primeira linha tem-se o número  $n$  de itens, na segunda linha a capacidade  $c$  dos recipientes e nas próximas  $n$  linhas, o tamanho dos  $n$  itens. A geração das instâncias giram em torno dos seguintes parâmetros:

- $n = \{50, 100, 200, 300, 400, 500, 750, 1000\}$
- $c = \{50, 75, 100, 120, 125, 150, 200, 300, 400, 500, 750, 1000\}$
- Tamanho mínimo do item =  $\{0.1c, 0.2c\}$
- Tamanho máximo do item =  $\{0.7c, 0.8c\}$

Para cada combinação de parâmetros são geradas 10 instâncias aleatórias, totalizando mais de 3000 mil instâncias.

Para realização dos testes do método desenvolvido, foi considerado as instâncias com:

- $n = \{100, 500\}$
- $c = \{50, 200, 400, 750, 1000\}$
- Tamanho mínimo do item =  $0.1c$
- Tamanho máximo do item =  $0.7c$

Das 10 instâncias geradas para cada combinação, uma é escolhida aleatoriamente para gerar a instância do CBPP, onde é atribuído cor para cada item da instância de forma aleatória. O número de cores é controlado pelo parâmetro  $Q = \{2, 5, 7\}$ . Ao todo, foram geradas 57 instâncias para o CBPP.

## 4. Resultados

Para a quantidade de recipientes, o modelo não apresentou melhora significativa se comparado a solução inicial, sendo que na maioria dos casos, a diferença foi de apenas um para todos os parâmetros adotados para o SA. Nos casos em que a diminuição foi maior do que um, a solução inicial com *HardBFD* foi avaliada com *Fitness* maior que 0,1, sendo que, em média possui valor de aproximadamente 0,07. Os resultados referentes a recipiente estão nas tabelas 3 e 4. Em vermelho são as execuções nas quais o número de recipientes não diminuiu, representando 28% das instâncias. Já em azul são os casos em que diminuiu-se apenas um recipiente, representando 49% das instâncias. Por fim, em verde temos os casos com diminuição mais significativa de recipientes, para 23% das instâncias.

Para a função de avaliação, a partir de análise visual do gráfico em 2 é possível notar que 1 não está em interseção com os demais e apresenta resultados piores. O experimento 1 é referente a solução inicial com *HardBFD*. Para os demais experimentos não é possível concluir qual o melhor. Ou seja, a aplicação do modelo com diferentes parâmetros para o SA gera soluções com melhor aproveitamento da capacidade do recipiente.

---

<sup>1</sup><https://site.unibo.it/operations-research/en/research/bpplib-a-bin-packing-problem-library>

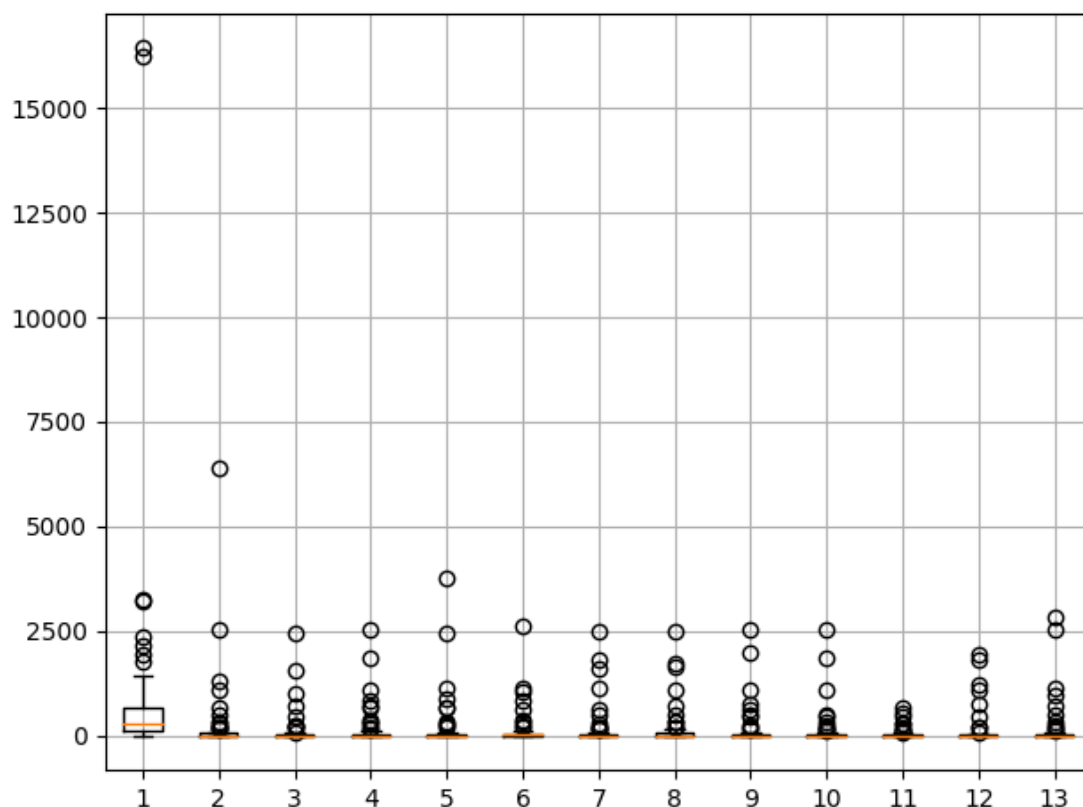


Figura 2. Gráfico Boxplot para resultado relativo ao melhor conhecido

Instâncias	HardBFD	0	1	2	3	4	5	6	7	8	9	10	11
100_50_2_0	44	40	41	40	40	41	41	41	40	41	41	40	40
100_50_2_1	49	40	40	40	40	40	40	40	40	40	39	40	40
100_50_2_2	41	41	40	41	40	40	40	40	40	40	41	40	40
100_50_5_0	44	43	43	43	43	43	43	43	43	43	43	43	43
100_50_5_1	44	43	43	43	43	43	43	43	43	43	43	43	43
100_50_5_2	43	43	43	43	43	43	43	43	43	43	43	43	43
100_50_7_0	43	42	42	42	42	42	42	42	42	42	42	42	42
100_50_7_1	43	42	42	42	42	42	42	42	42	42	42	42	42
100_50_7_2	43	42	42	42	42	42	42	42	42	42	42	42	42
100_200_2_0	47	42	42	43	42	42	42	42	42	42	42	42	42
100_200_2_1	43	43	43	43	43	43	43	43	43	43	43	43	43
100_200_2_2	43	43	43	43	43	43	43	43	43	43	43	43	43
100_200_5_0	43	42	42	42	42	42	42	42	42	42	42	42	42
100_200_5_1	42	42	42	42	42	42	42	42	42	42	42	42	42
100_200_5_2	42	42	42	42	42	42	42	42	42	42	42	42	42
100_200_7_0	41	40	40	40	40	40	40	40	40	40	40	40	40
100_200_7_1	41	40	40	40	40	40	40	40	40	40	40	40	40
100_200_7_2	41	41	41	41	41	41	41	41	41	41	41	41	41
100_400_2_0	42	41	41	41	41	41	41	41	41	41	41	41	41
100_400_2_1	51	42	42	41	41	42	42	41	41	41	42	41	42
100_400_2_2	42	41	41	41	41	41	41	41	41	41	41	41	41
100_400_5_0	45	44	44	44	44	44	44	44	44	44	44	44	44
100_400_5_1	44	43	43	43	43	43	43	43	43	43	43	43	43
100_400_5_2	45	44	44	44	44	44	44	44	44	44	44	44	44
100_400_7_0	43	41	41	41	41	41	41	41	41	41	41	41	41
100_400_7_1	41	40	40	40	40	40	40	40	40	40	40	40	40
100_400_7_2	42	41	41	41	41	41	41	41	41	41	41	41	41
100_750_2_0	45	42	43	42	42	42	42	42	43	43	43	42	42
100_750_2_1	43	42	42	42	42	42	42	42	42	42	42	42	42
100_750_2_2	45	43	42	42	42	42	42	42	42	42	42	42	42

Figura 3. Tabela com número final de recipientes para cada instância (Parte 1)

## 5. Conclusão

Diante dos resultados apresentandos, o modelo é capaz de reorganizar os itens nos recipientes de forma a melhorar o aproveitamento de sua capacidade a partir de uma heurística

100_750_5_0	41	41	41	41	41	41	41	41	41	41	41	41	41
100_750_5_1	42	42	41	41	41	41	42	42	42	42	42	41	42
100_750_5_2	41	41	40	41	41	41	41	41	41	41	41	40	41
100_750_7_0	43	42	42	42	42	42	42	42	42	42	42	42	42
100_750_7_1	42	41	41	41	41	41	41	42	41	41	41	41	41
100_750_7_2	41	41	40	41	40	41	41	41	41	41	40	40	41
100_1000_2_0	44	43	42	43	43	42	42	43	43	43	43	43	43
100_1000_2_1	52	43	44	44	43	45	43	44	42	43	43	44	43
100_1000_2_2	44	42	42	42	42	42	42	42	42	42	42	42	42
100_1000_5_0	43	42	42	42	42	42	42	42	42	42	42	42	42
100_1000_5_1	43	43	43	43	43	43	43	43	43	43	43	43	43
100_1000_5_2	43	42	42	42	42	42	42	42	42	42	42	42	42
100_1000_7_0	39	38	38	38	38	38	38	38	38	38	38	38	38
100_1000_7_1	39	39	39	39	39	39	39	39	39	39	39	39	39
100_1000_7_2	39	38	38	38	38	38	38	38	38	38	38	38	38
500_50_2_0	206	204	204	204	204	204	204	204	204	204	204	204	204
500_50_2_1	210	204	204	204	205	204	204	205	204	204	204	204	204
500_50_5_0	207	206	206	206	206	206	206	205	206	205	206	206	206
500_50_5_1	207	206	206	206	206	206	206	206	206	206	206	206	206
500_50_7_0	207	205	205	205	205	205	205	205	205	205	205	205	205
500_50_7_1	207	206	206	206	207	206	206	207	206	206	206	206	206
500_200_2_0	207	205	205	205	205	205	206	205	205	205	205	205	205
500_200_2_1	207	206	206	206	206	206	206	206	206	206	206	206	206
500_200_5_0	207	206	206	206	206	206	206	206	206	206	206	206	206
500_200_5_1	206	206	205	206	206	206	205	206	205	205	205	205	206
500_200_7_0	206	206	206	206	206	206	206	206	206	206	206	206	206
500_200_7_1	205	205	205	205	205	205	205	205	205	205	205	205	205

**Figura 4. Tabela com número final de recipientes para cada instância (Parte 2)**

(*HardBFD*) que gera bons resultados para o CBPP [da Silva et al. 2022], porém não é possível concluir se é o melhor resultado que pode ser gerado para a função de avaliação e também para o número de recipientes, pois não foi possível comparar com outros métodos. Além disso, se o objetivo é encontrar uma solução com número mínimo possível de recipientes de forma rápida, utilizar apenas a heurística *HardBFD* já seria suficiente, visto que o modelo para esse caso, os valores entre a heurística e o modelo são próximos e quanto maior a instância, maior o tempo de execução, sendo que a heurística entrega uma solução em menor tempo de execução quando comparada ao modelo proposto.

## Referências

- Borges, Y. and Schouery, R. (2020). Modelos pseudo polinomiais para o problema do empacotamento colorido. In *Anais do V Encontro de Teoria da Computação*, pages 37–40, Porto Alegre, RS, Brasil. SBC.
- Böhm, M., Sgall, J., and Veselý, P. (2014). Online colored bin packing.
- da Silva, R. F. F., Borges, Y. G. F., and Schouery, R. C. S. (2022). Heurísticas para o problema do empacotamento colorido. *ANAIS DO SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL*.
- Hyde, M. R., Ochoa, G., Vázquez-Rodríguez, J. A., and Curtois, T. (2011). A hyflex module for the one dimensional bin packing problem.