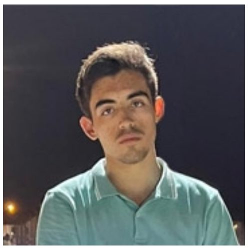
Investigação Operacional
Problema de Empacotamento

1º Trabalho

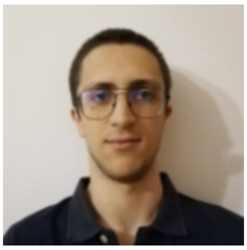
March 24, 2024

Trabalho realizado por:

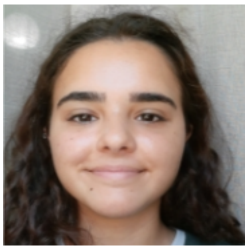
Gonçalo Rocha Sousa Freitas
Henrique Guimarães Pereira
Eduarda Mafalda Martins Vieira
Maria Leonor Carvalho da Cunha



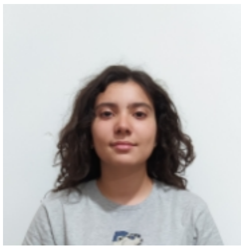
A104350
Gonçalo Rocha Sousa Freitas



A97205
Henrique Guimarães Pereira



A104098
Eduarda Mafalda Martins Vieira



A103997
Maria Leonor Carvalho da Cunha

Contents

1 Introdução

Este trabalho aborda um problema de empacotamento, *bin packing problem*, onde existem contentores de diferentes capacidades, itens de diferentes tamanhos e uma quantidade limitada de contentores e de itens. O nosso objetivo é encontrar uma solução, utilizando o método de Dyckhoff de modo a que consigamos colocar todos os itens ,de vários tamanhos, na quantidade de contentores existentes da maneira mais eficiente possível.

2 Alterações ao Problema

O maior número presente no grupo, é o 104350.
Isto significa que o x = 1, A = 0, B = 4, C = 3, D = 5, E = 0
Assim, conseguimos determinar a quantidade de contentores e itens disponíveis de cada tamanho utilizando a formula que nos foi dada no enunciado.

Contentores:

comprimento	quantidade disponível
11	ilimitada
10	5
7	6

Items:

comprimento	quantidade
1	0
2	11
3	10
4	2
5	5

Conforme os itens disponibilizados, ficamos com:
$2 \times 11 + 3 \times 10 + 4 \times 2 + 5 \times 5 + 5 \times 5 = 85$
Isso significa que o valor da solução ótima é 85

3 Formulação do Problema

Neste problema, estamos a tentar descobrir a maneira mais eficiente de colocar um grupo de itens em diferentes tamanhos de contentores, garantindo que todos os itens sejam acomodados. Temos três tamanhos de contentores disponíveis, com comprimentos de 11, 10 e 7 unidades. Para cada tamanho de contentor, temos uma quantidade específica disponível: um número ilimitado de contentores para o tamanho de 11 unidades, 5 contentores disponíveis para o tamanho de 10 unidades e 6 contentores disponíveis para o tamanho de 7 unidades. O grupos de itens é formado por: 2 itens de comprimento 1 , 4 itens de comprimento 2, 13 itens de comprimento 4 e 5 itens de comprimento 5. O objetivo é descobrir como organizar todos os itens nos contentores disponíveis de forma a minimizar o espaço desperdiçado, utilizando o modelo de 'úm-corte' de Dyckhoff.

Utilizamos como variáveis de decisão x(y,z) , e esta é inteira não negativa (x(y,z) maior igual a 0)). Onde a variável y representa o tipo de contentores usado, onde pode ser ou não um contentor residual (que foi o contentor resultante do resto de um corte a um determinado contentor), e a variável z representa o comprimento do corte que vai ser feito nesse contentor.

4 Modelo de Programação Linear

Contentores:

$$Contentortam7 : x75 + x74 + x73 + x72 - x114 - x103 - x92 <= 6$$

O número de contentores de comprimento 7 não pode exceder 6 contentores. Desta forma, subtrai-se e exclui-se aqueles contentores uma vez que, através de um corte, dão origem a um contentor residual de comprimento 7.

$$Contentortam7 : x75 + x74 + x73 + x72 - x114 - x103 - x92 >= 0$$

Os contentores de comprimento 7 têm de ser, no mínimo, nulo. Assim, subtrai-se e exclui-se aqueles contentores pois , através de um corte, dão origem a um contentor residual de comprimento 7.

$$Contentortam10 : x102 + x103 + x104 + x105 <= 5;$$

O número de contentores de comprimento 10 não pode exceder 5 contentores

$$Contentortam10 : x102 + x103 + x104 + x105 >= 0;$$

O número de contentores de comprimento 10 tem de ser no mínimo nulos, não podem ser negativos.

$$Contentortam11 : x112 + x113 + x114 + x115 >= 0;$$

Os contentores de comprimento 11 têm de ser, no mínimo, nulos.

Tipos de corte: Sabemos que os nossos cortes podem ser feitos em 2,3,4 e 5.

$$Corte1 : x32 + x43 + x54 + x65 >= 0;$$

No Corte 1, o número de contentores residuais de y=6, y=5, y=4 e y=3 ou é um número positivo, ou nulo.

$$Corte2 : x32+x42+x52+x62+x72+x82+x92+x102+x112+x42+x53+x64+x75 >= 11;$$

No Corte2, o número de cortes efetuados em contentores, quer sejam ou não residuais, de comprimento y, tem de ser superior ou igual a 11.

$$Corte3 : x43+x53+x63+x73+x83+x93+x103+x113+x52+x63+x74+x85-x32 >= 10;$$

No Corte3, o número de cortes efetuados em contentores, quer sejam ou não residuais, de comprimento y, tem de ser superior ou igual a 10.

$Corte4 : x54+x64+x74+x84+x94+x104+x114+x62+x73+x84+x95-x43-x42 >= 2;$

No Corte4, o número de cortes efetuados em contentores, quer sejam ou não residuais, de comprimento y, tem de ser superior ou igual a 2.

$Corte5 : x65+x75+x85+x95+x105+x115+x72+x83+x94+x105-x54-x53-x52 >= 5;$

No Corte5, o número de cortes efetuados em contentores, quer sejam ou não residuais, de comprimento y, tem de ser superior ou igual a 5. Uma vez que há 2 vezes o valor de x10,5 retira-se o x5,4; x5,3 e x5,2, pois num contentor de 10 ao fazer um corte de 5, originará dois contentores de 5. Retiram-se assim, os casos em que 5 não é utilizado como uma quantidade empacotada, mas sim como um contentor residual para outros cortes.

$$Corte6 : x82 + x93 + x104 + x115 >= x65 + x64 + x63 + x62;$$

No Corte6, O número de contentores que, ao serem cortados em um comprimento de y unidades, resultarão em um contentor residual de comprimento 6, precisa ser superior ao número de contentores com esse comprimento.

$$Corte8 : x102 + x113 >= x85 + x84 + x83 + x82;$$

No Corte8, o número de contentores de comprimento 10, que ao serem cortados em segmentos de comprimento 2, resultarão em um contentor residual de comprimento 8, deve ser maior do que a quantidade de contentores disponíveis com esse comprimento

$$Corte9 : x112 >= x95 + x94 + x93 + x92;$$

No Corte9, o número de contentores de comprimento 11, que ao serem cortados em segmentos de comprimento 2, resultarão em um contentor residual de comprimento 9, precisa ser maior do que a quantidade de contentores disponíveis com esse comprimento.

min: 7 x72 + 7 x73 + 7 x74 + 7 x75 - 7 x92 - 7 x103 - 7 x114 + 10 x102 + 10 x103 + 10 x104 + 10 x105 + 11 x112 + 11 x113 + 11 x114 + 11 x115;

A função objetivo tem o custo de cada contentor associado a cada variável de decisão. A título de exemplo, uma variável de decisão que exija um corte num contentor de comprimento 10 terá um custo associado de 10. Retira-se o custo associado às variáveis de decisão 7 y9,2 ; 7 y10,3 e 7 y11,4 por se tratarem de contentores residuais provenientes de cortes noutros contentores.

5 Ficheiro Input e Output

```
/* Objective function */
min: 7 x72 + 7 x73 + 7 x74 + 7 x75 - 7 x92 - 7 x103 - 7 x114 + 10 x102 + 10 x103 + 10 x104 + 10 x105 + 11 x112 + 11 x113 + 11 x114 + 11 x115;

/* Variable bounds */
Corte1: x32 + x43 + x54 + x65 >= 0;
Corte2: x32 + x42 + x52 + x62 + x72 + x82 + x92 + x102 + x112 + x42 + x53 + x64 + x75 >=  11;
Corte3: x43 + x53 + x63 + x73 + x83 + x93 + x103 + x113 + x52 + x63 + x74 + x85 - x32 >= 10;
Corte4: x54 + x64 + x74 + x84 + x94 + x104 + x114 + x62 + x73 + x84 + x95 - x43 - x42 >= 2;
Corte5: x65 + x75 + x85 + x95 + x105 + x115 + x72 + x83 + x94 + x105 - x54 - x53 - x52 >= 5;
Corte6: x82 + x93 + x104 + x115 >= x65 + x64 + x63 + x62;
Corte8: x102 + x113 >= x85 + x84 + x83 + x82;
Corte9: x112 >= x95 + x94 + x93 + x92;

Contentortam7: x75 + x74 + x73 + x72 - x114 - x103 - x92 >= 0;
Contentortam7: x75 + x74 + x73 + x72 - x114 - x103 - x92 <= 6;

Contentortam10: x102 + x103 + x104 + x105 >= 0;
Contentortam10: x102 + x103 + x104 + x105 <= 5;

Contentortam11: x112 + x113 + x114 + x115 >= 0;

int x32, x42, x43, x52, x53, x54, x62, x63, x64, x65, x72, x73, x74, x75, x82, x83, x84, x85, x92, x93, x94, x95, x102, x103, x104, x105, x112, x113, x114, x115;
```

Variables	MILP ...	result
	85	85
x72	0	0
x73	0	0
x74	0	0
x75	0	0
x92	0	0
x103	0	0
x114	0	0
x102	0	0
x104	0	0
x105	3	3
x112	5	5
x113	0	0
x115	0	0
x32	0	0
x43	0	0
x54	0	0

6 Solução Ótima

Ao analisar o *output* do programa *lpsolve* reparamos que:

- São necessários 7 contentores, 3 de comprimento 10 e 5 de comprimento 11.

7 Validação do Modelo

Como:

$$3 * 10 + 5 * 11 = 85$$

Como tal, podemos concluir que o modelo é válido, pois cumpre todas as restrições impostas pelo enunciado.