

Relatório 3º Projeto ASA 2023/2024

Nome: João Miguel Calejo Alcalde Teixeira, 106634

Grupo: al138

Formalização do modelo linear:

(toy_vars[1], toy_vars[2], ... , toy_vars[num_toys + 1]) : Este vetor contém as variáveis que representam o número de brinquedos fabricados do respetivo tipo, daí existir uma variável destas para cada brinquedo.

(package_vars[1], package_vars[2], ... , package_vars[num_packages + 1]): Este vetor contém as variáveis que representam o número de packages fabricadas do respetivo tipo, daí existir uma variável destas para cada package existente.

Sendo Pp, Tp os lucros dos pacotes e dos brinquedos respetivamente, Pv, Tv as variáveis do problema para os pacotes e para os brinquedos respetivamente, x um vetor de vetores com as Pv inseridas consoante necessário e C as capacidades dos brinquedos.

Objetivo:

$$\max \left(\sum_{n=1}^{\rho} Pp_n \cdot Pv_n + \sum_{m=1}^t Tp_m \cdot Tv_m \right)$$

Restrições:

$$\left(Tv[i] + \sum_{n=1}^s x[i][n] \right) \leq C[i] \quad , i \in [0, n]$$

$$\sum_{i=1}^n Tv_i + \sum_{c=1}^p 3 * Pv_e \leq \max_Toys_day$$

Análise Teórica:

Seja n número de brinquedos e p número de pacotes. A complexidade do programa em função destes dois parâmetros é O(n + p).

- O número de variáveis do programa linear é dado por O(n + p). As definições das variáveis dos brinquedos são definidas na leitura do input na parte dos brinquedos. Este processo é realizado num loop() que itera **n** vezes. De igual modo decorre a definição das variáveis dos pacotes num loop() que itera **p** vezes.
- O número de restrições do programa linear é dado por O(2n+p). As restrições da quantidade de brinquedos tendo em conta a capacidade máxima de cada um deles são definidas num loop() que itera **n** vezes, são ainda definidos “upper bounds” e “lower bounds” durante o processo da definição das variáveis, processo que envolve uma complexidade de O(n + p). Por fim a restrição que controla o número máximo de produção diária é feita de modo instantâneo, O(1).

Avaliação Experimental dos Resultados:

Com o auxílio de um gerador de instâncias para o programa foram gerados 14 inputs para o programa executar. Nesta tabela estão apresentados os parâmetros utilizados no gerador de modo a produzir os inputs e os seus respectivos tempos de execução.

n	p	toy min	toy max	max profit	Pok	seed	time
600	400	70	200	30	7	-	0.185
1200	800	140	400	37	14	-	0.356
1800	1200	210	600	44	21	-	0.484
2400	1600	280	800	51	28	-	0.672
3000	2000	350	1000	58	35	-	0.832
3600	2400	420	1200	65	42	-	1.095
4200	2800	490	1400	72	49	-	1.368
4800	3200	560	1600	79	56	-	1.68
5400	3600	630	1800	86	63	-	2.171
6000	4000	700	2000	93	70	-	2.354
6600	4400	770	2200	100	77	-	2.491
7200	4800	840	2400	107	84	-	2.987
7800	5200	910	2600	114	91	-	3.347
8400	5600	980	2800	121	98	-	3.662

Como foi demonstrado na avaliação teórica, a complexidade do programa aparenta ser $O(n+p)$. Procuramos provar esta afirmação produzindo um gráfico linear que produz o tempo de execução em função da sua complexidade.

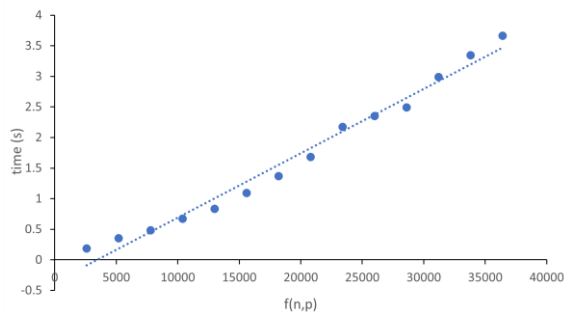


Figura1: $f(n,p) = n*3 + p*2 + 1$, neste gráfico $f(n,p)$ corresponde à soma de número máximo de restrições e número máximo de variáveis.

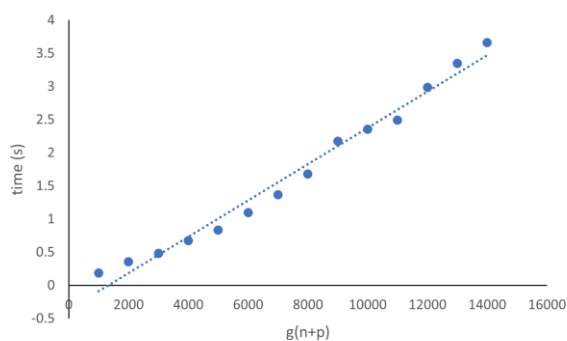


Figura2: $g(n,p) = n + p$, neste gráfico $g(n,p)$ corresponde à soma do número de brinquedos com o número de pacotes.

Como é possível verificar, $g(n,p)$ corresponde à complexidade $O(n+p)$ e $f(n,p)$ também pode ser reduzido a esse valor. Assim a partir da produção dos dois gráficos verifica-se que a complexidade do programa é a esperada, $O(n+p)$.