

UNIVERSIDADE FEDERAL DO PARANÁ

Discente: Yuri Junqueira Tobias – GRR20211767

Docente: André Guedes

Data de entrega: 09/05/2023

Disciplina: Otimização – CII238

Modelagem e implementação, por programação linear, de uma solução para um problema em específico.

1. Introdução

O presente trabalho consiste na modelagem e implementação, por programação linear, de uma solução para o seguinte problema (Produção de Produtos Químicos): uma empresa química produz n tipos de produtos e, para produzir os produtos, utiliza diferentes porções de m diferentes compostos. Cada produto i tem um valor de venda (por litro), v_i . Cada composto j tem um preço (por litro), p_j e um limite diário de volume q_j . A quantidade de uso do composto j utilizado na produção do produto i é dada por c_{ij} . A empresa assume que toda sua produção será vendida e, com isso, deseja maximizar seu lucro.

Vale ressaltar também que um problema de programação linear consiste em maximizar uma dada função linear dentre um conjunto de todos os vetores que satisfazem um determinado sistema de equações lineares e desigualdades. Além disso, cada programa linear pode ser facilmente transformado para a forma:

$$\text{maximize } c^T x \text{ sujeito a } Ax \leq b$$

Sendo $c^T x$ a função linear a ser maximizada, também conhecida como **função objetivo**; Ax a matriz que representa o lado esquerdo do sistema de equações e b o lado direito. Ou seja, c é um vetor com os coeficientes das variáveis presentes no vetor x , A é a matriz com os coeficientes das variáveis presentes também em x e, por fim, b é o vetor que contém o resultado das desigualdades para cada equação do sistema linear.

2. Modelagem

Levando em consideração que a empresa deseja maximizar seu lucro e o mesmo varia de acordo com o produto que está sendo vendido, então a ideia da modelagem que será abordada no presente trabalho, seguindo a forma apresentada acima (introdução), consiste em:

- c : lucro por produto;

- x : vetor contendo as variáveis que formam uma bijeção para os produtos;
- A : matriz contendo os coeficientes das restrições, tais coeficientes representam a quantidade de uso de determinado composto em determinado produto;
- b : vetor contendo o limite de uso diário dos compostos;

A partir disso, é possível observar que iremos chegar ao seguinte modelo:

$$\begin{aligned} & \bullet \quad \text{Max } \sum_{i=1}^n \sum_{j=1}^m (v_i - p_j * c_{ij}) * x_i \\ & \text{subject to } \sum_{i=1}^n (c_{ij} * x_i) \leq q_j \quad \text{para todo } j \in [1..m] \\ & n, m, i, j \in \mathbb{Z}^+ \\ & v_i, p_j, q_j, c_{ij} \in \mathbb{R}^+ \end{aligned}$$

Com isso, podemos chegar à seguinte conclusão: iremos maximizar o somatório do lucro que cada produto dá, ou seja, o valor de venda do produto (v_i) menos o custo de produzir aquele produto ($p_j * c_{ij}$ para todo $j \in [1..m]$), vezes a quantidade do produto (x_i).

Ademais, a única restrição que temos é a de limitar a quantidade de uso diário dos compostos, e daí temos: todo composto está sujeito ao somatório da sua quantidade de uso em determinado produto (c_{ij}) vezes a quantidade daquele produto (x_i) para todos os produtos, sendo o limite do composto dado por q_j (valores presentes em b). Neste trabalho assume-se que n e m (quantidade de produtos e compostos) são inteiros positivos e, os demais valores pertencem aos reais positivos.

3. Implementação

O objetivo principal da implementação consiste em gerar uma saída para ser usada pelo *lp_solve* a partir de uma entrada que segue determinado padrão - que será apresentado a seguir. Os formatos de entrada e saída utilizam a entrada e a saída padrões (stdin e stdout), e a entrada pode conter números separados por 1 ou mais espaços, tabs ou fim de linha, tudo conforme solicitado.

A entrada deve seguir o seguinte padrão:

- Inicia com dois números inteiros **n** e **m** indicando a quantidade de produtos e compostos, respectivamente;
- Em seguida temos **n** números indicando os preços de venda (de 1 litro, v_i) de cada produto;
- Após temos **m** linhas, com 2 números em cada, indicando o custo (de 1 litro, p_j) e o limite de produção (q_j) de cada composto usado como matéria prima;
- Logo após temos os **$n * m$** números indicando a quantidade de cada composto usada na produção de 1 litro de cada produto. As quantidades de cada composto

para a produção do primeiro produto ($c_{11}, c_{12}, \dots, c_{1m}$), estão na primeira linha, depois as quantidades para o produto 2, e assim até o produto n .

O trabalho está dividido em três arquivos: *producao.c*, *makefile* e *input.txt*. Sendo que o primeiro contém as funções auxiliares e o programa principal para receber a entrada, tratar e gerar a saída para o *lp_solve*. O segundo é apenas para compilar o primeiro e gerar o executável *producao* conforme solicitado. E o terceiro vem com um exemplo – presente no enunciado e que será apresentado a seguir –, e serve apenas para facilitar na hora de gerar entradas, uma vez que basta o usuário redirecionar o arquivo para o input de *producao*.

O arquivo *producao.c* contém, além da função principal (*main*), 1 estrutura (*struct*) e 4 funções auxiliares:

- **Struct:** contém todos os valores/atributos necessários para a produção e contém a descrição de cada um no próprio código;
 - **leEntrada(...):** lê os valores da entrada padrão (*stdin*) e faz as respectivas atribuições e alocações de memória;
 - **calculaLucros(...):** calcula quanto cada produto dá de lucro e armazena os valores no vetor de lucros presente na estrutura (*struct*);
 - **imprimeLPSolve(...):** gera a saída no formato de entrada para o *lp_solve*;
 - **deletaProducao(...):** libera o espaço de memória alocado para os atributos da estrutura (*struct*);
 - **main():** instancia uma estrutura (*struct*) e chama todas as funções auxiliares na sequência apresentada acima;
- Obs: todas as funções auxiliares recebem apenas a estrutura (*struct*);

4. Exemplo

Considere uma entrada com $n = 3$ produtos e $m = 4$ compostos, da seguinte forma:

		compostos				valor
	n/m	1	2	3	4	
pro du tos	1	0,2	0,5	1,0	0,1	10
	2	1,0	0,1	0,3	0,1	7
	3	0,4	0,2	0,2	0,0	3
custo		1	2	5	10	
limite		1000	2000	500	2000	

Figura 1: tabela de exemplo presente no enunciado do trabalho

A mesma surge a partir da seguinte entrada (padrão do arquivo input.txt):

- 3 4
10 7 3
1 1000
2 2000
5 500
10 2000
0.2 0.5 1.0 0.1
1.0 0.1 0.3 0.1
0.4 0.2 0.2 0.0

O programa `producao.c` para esta instância irá gerar a seguinte saída:

max: $+ 2.8x_1 + 3.3x_2 + 1.2x_3$;

$+ 0.2x_1 + 1.0x_2 + 0.4x_3 \leq 1000.0$;

$+ 0.5x_1 + 0.1x_2 + 0.2x_3 \leq 2000.0$;

$+ 1.0x_1 + 0.3x_2 + 0.2x_3 \leq 500.0$;

$+ 0.1x_1 + 0.1x_2 + 0.0x_3 \leq 2000.0$;

Quando aplicarmos a saída acima como entrada do `lp_solve`, o mesmo irá gerar seguinte saída:

- Value of objective function: 3755.31914894

Actual values of the variables:

x1	212.766
x2	957.447
x3	0

Para este exemplo um plano ótimo tem ganho de R\$ 3.755,31, onde são produzidos 212,766 litros do produto 1, 957,447 litros do produto 2 e 0 litros do produto 3 (de acordo com o enunciado). Como podemos perceber, a saída do `lp_solve` para a saída do executável *producao* corresponde ao resultado esperado.

5. Referências:

[1]: Understanding and Using Linear Programming. Jiří Matoušek, Bernd Gärtner. 2007.