

Relatório Trabalho 1 - Programação Linear

Pedro Willian Aguiar e Fernando Gbur dos Santos
Universidade Federal do Paraná – UFPR
Curitiba, Brasil

I. INTRODUÇÃO

O presente relatório tem o objetivo de apresentar a implementação e a modelagem da solução para o problema de produção de produtos químicos.

II. DESCRIÇÃO DO PROBLEMA

Uma empresa química produz n tipos diferentes de produtos. Para produzir estes produtos usa diferentes proporções de m diferentes compostos (matérias primas). Cada produto i tem valor de venda (por litro), v_i . Cada composto j usado tem um preço (por litro), p_j , e um limite diário de volume (em litros), q_j . A quantidade (em litros) de uso de cada composto j na produção de 1 litro do produto i é dada por c_{ij} . Ou seja, para produzir 1 litro do produto i são necessários c_{i1} litros do composto 1, c_{i2} litros do composto 2, e assim até c_{im} litros do composto m . A empresa assume que toda sua produção será vendida. Levando em consideração os dados, e que os demais custos de produção não dependem de quais produtos são feitos, a empresa quer maximizar os lucros.

A. Modelagem

$$\text{maximize } \sum_{i=1}^n w_i x_i \quad (1)$$

$$\text{sujeito a } \sum_{i=1}^m \sum_{j=1}^n c_{ji} x_j \leq q_i \quad (2)$$

w_i representa o coeficiente da variável de decisão x_i na função objetivo. Podemos modelar w_i como:

$$w_i = v_i - \sum_{k=1}^m c_{ki} * p_k \quad (3)$$

onde w_i é a subtração do valor de venda v_i pelo somatório do produto do limitante c_{ki} pelo preço por litro p_k .

Os limitantes são implementados dessa forma (como uma matriz transposta) pois os dados dos produtos são recebidos em relação aos compostos e é preciso a soma dos compostos em relação aos produtos.

O objetivo é encontrar os valores das variáveis de decisão $(x_1, x_2, x_3, \dots, x_n)$ que maximizam a função objetivo, sujeitas às restrições lineares fornecidas. O somatório é utilizado para representar a soma dos termos em cada equação da função objetivo e das restrições, permitindo expressar o problema de forma mais concisa e generalizada.

III. IMPLEMENTAÇÃO

A implementação foi feita em linguagem C. O programa gera um arquivo chamado *entrada_lp_solve.txt* no formato de entrada aceito pelo *lp_solve* com a descrição do programa linear que resolve o problema para uma instância dada.

Primeiramente, as variáveis n e m são recebidas e, a partir delas, são criadas as outras variáveis de armazenamento. Para armazenar os valores de produtos foi criado *valorProdutos[n]* e os limitantes foram armazenados na matriz *limitantes[n][m]*. Para os compostos foi criado a seguinte estrutura:

```
1 struct Composto {  
2     int custo;  
3     int limite;  
4 };  
5 typedef struct Composto Composto_t;
```

Código 1. Struct Composto

A partir desta estrutura, foi criado um vetor com os custos e limites chamado *compostos[m]* do tipo *Composto_t*.

Para impressão das restrições é preciso fazer a transposta da matriz de limitantes recebida, pois precisamos da soma em relação aos compostos. Sua implementação foi feita igual à equação 2, apenas acrescentando detalhes para a impressão correta para que seja aceito no *lp_solve*.

A função objetivo também foi implementada como a modelagem (1), se atentando aos detalhes da linguagem e do formato de saída.

A. Exemplo

Para o teste do programa feito foi utilizado o exemplo passado na descrição do trabalho, com as seguintes entradas:

```
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
```

3 e 4 são os n produtos e m compostos, respectivamente. 10, 7 e 3 os valores dos produtos. As duplas 1 e 1000, 2 e 2000, 5 e 500 e 10 e 2000 são os custos e limites dos compostos 1, 2, 3 e 4. Os limitantes são os próximos e últimos dados recebidos (produtos em relação aos compostos).

Para a construção da função objetivo foram feito os seguintes cálculos:

$$\begin{aligned}10 - (0.2 * 1 + 0.5 * 2 + 1 * 5 + 0.1 * 10) &= 2.8 \\7 - (1.0 * 1 + 0.1 * 2 + 0.3 * 5 + 0.1 * 10) &= 3.3 \\3 - (0.4 * 1 + 0.2 * 2 + 0.2 * 5 + 0.0 * 10) &= 1.2\end{aligned}\quad (4)$$

Resultando na função objetivo

$$2.8x_1 + 3.3x_2 + 1.2x_3 \quad (5)$$

Para as restrições é feita à transposta dos limitantes pois precisamos dos dados de compostos em relação aos produtos, sendo que a soma desses valores devem ser menores ou iguais que os limites (1000, 2000, 500, 2000) dos compostos 1, 2, 3 e 4.

O resultado final que é aceito pelo *lp_solve* é:

$$\text{max: } 2.8x_1 + 3.3x_2 + 1.2x_3$$

$$\begin{aligned}0.2x_1 + 1.0x_2 + 0.4x_3 &\leq 1000; \\0.5x_1 + 0.1x_2 + 0.2x_3 &\leq 2000; \\1.0x_1 + 0.3x_2 + 0.2x_3 &\leq 500; \\0.1x_1 + 0.1x_2 + 0.0x_3 &\leq 2000;\end{aligned}\quad (6)$$

Essa entrada utilizada como exemplo está salvo como um arquivo chamado *teste.txt* no mesmo diretório que o programa.

IV. CONCLUSÃO

O objetivo deste relatório foi descrever o problema, modelá-lo matematicamente e apresentar a implementação da solução em linguagem C, gerando um arquivo de entrada compatível com o *lp_solve* para resolver o problema em questão.

Essa abordagem permite que a empresa tome decisões informadas sobre a produção de produtos químicos, visando maximizar seus lucros. Com a implementação em linguagem C, a solução pode ser executada de forma eficiente, facilitando a resolução do problema em diferentes instâncias.