UNIVERSIDADE FEDERAL DO PARANÁ
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Discente: Matheus Luiz de Oliveira Souza (GRR20210259)
Docente: André Guedes
Disciplina: Otimização (CI1238)
Data: 28/05/2023

MODELAGEM E IMPLEMENTAÇÃO PARA O PROBLEMA DE PRODUÇÃO DE PRODUTOS QUÍMICOS

1. Resumo

Neste relatório estarei apresentando o problema da "Produção de Produtos Químicos" e como modelei, implementei (em linguagem C) e resolvi esse problema utilizando Programação Linear.

2. Introdução

Os algoritmos foram todos realizados em linguagem C e, para maior organização e agilidade no processo de compilação dos arquivos headers, foi utilizado um Makefile.

Dentre os códigos desenvolvidos, utilizei como referência os ensinamentos passados em aula e o livro "*Understanding and Using Linear Programming*", Jiří Matoušek, Bernd Gärtner - 2007.

Para este trabalho, foi utilizado o lp_solve como ferramenta para resolver o problema modelado.

3. Contexto geral do problema

Uma empresa fabricante de produtos químicos quer maximizar seus lucros levando em consideração o preço de venda de cada produto e o custo referente aos compostos utilizados para produzir cada um desses produtos.

Essa empresa produz N tipos de produtos e utiliza diferentes proporções de M tipos de compostos para produzir tais produtos.

4. Modelagem do problema

Primeiramente, após analisar o problema, cheguei na seguinte fórmula que representa a função objetivo desse problema em específico:

$$\sum_{i=1}^{n} \left(V_i - \sum_{j=1}^{m} C_j \cdot Q_{ij}\right) \cdot P_i$$

onde n é a quantidade de tipos diferentes de produtos, Vi é o preço de venda de um litro do produto i, m é a quantidade de compostos diferentes, Cj é o custo do litro do composto j, Qij é a quantidade necessária do composto j (em litros) para produzir um litro do produto i e Pi são as variáveis da função objetivo que representam a quantidade de cada produto que irão maximizar o lucro dessa empresa.

Após encontrar a função objetivo, cheguei na fórmula que caracteriza as restrições deste problema:

$$\sum_{i=1}^{n} Q_{ij} \cdot P_i \leq L_j \text{, } tal\, que\ j \in (1,2,3,\ldots,m)$$

onde n é a quantidade de tipos diferentes de produtos, Qij é a quantidade necessária do composto j (em litros) para produzir um litro do produto i, Pi são as variáveis que representam a quantidade de cada produto, Lj refere-se ao limite diário de cada composto, j é um número natural que pertence ao conjunto que vai de 1 até m, onde m é a quantidade de compostos diferentes disponíveis para fabricação dos produtos químicos.

Por fim, cabe ressaltar que as quantidades de cada produto são sempre maiores ou iguais a zero, ou seja:

$$P_1, P_2, P_3, \ldots, P_n \geq 0$$

5. Implementação

Como já foi dito anteriormente, para essa implementação utilizei a linguagem C, de modo que após rodar o comando make, será gerado um executável de nome "producao".

É esperado um conjunto de dados que representam as quantidades de produtos e de compostos, o valor de venda de cada produto, o custo de cada composto e sua quantidade limite diária e a quantidade, em litros, de cada composto para produzir determinado produto.

Assim, pela entrada padrão (STDIN), eu leio esses valores, por meio da função "capturaDados()" e modelo o problema, que vai servir de entrada para o lp_solve, com a função "modelaPL()" e mando para a saída padrão (STDOUT).

6. Exemplo de execução

Para utilizar esse programa de modelagem e resolução do problema da “Produção de Produtos Químicos”, seguiremos os seguintes passos:

1. Executar o comando “*make*” dentro do diretório do projeto.
2. Temos algumas opções para receber o resultado:
   a. Executar o seguinte comando: ./producao < dados_entrada.txt | lp_solve, onde dados_entrada.txt é um arquivo que contém os dados do problema em questão que servirão de base para a modelagem. Logo após a modelagem do problema, o resultado desse passo será lançado para o lp_solve, onde o problema realmente será resolvido e nos dará combinação das quantidades de cada produto que maximizam o lucro da empresa.
   b. Executar este outro comando: ./producao | lp_solve, onde agora, ao invés de passar todos os dados necessários por meio de um arquivo, será esperado que o usuário digite os dados no terminal. Após terminar de digitar os dados e teclar ENTER e depois CTRL-D, o resultado da modelagem será passado para o lp_solve para nos entregar a solução ótima.
3. Além disso, podemos pedir para gerar um arquivo com o problema modelado. Isso pode ser feito passando a flag “-f” e o nome do arquivo. Por exemplo:
   a. ./producao -f nome_arquivo < dados_entrada.txt | lp_solve
   b. ./producao -f nome_arquivo | lp_solve

*Obs: o caractere “|” sempre ao lado do “lp_solve” é o “*pipe*”.

Prints de exemplo para dois tipos de execução citadas:

```
→  t1_otimizacao ./producao < teste.in | lp_solve

Value of objective function: 3755.31914894

Actual values of the variables:
p1                         212.766
p2                         957.447
p3                               0
```

```
→  t1_otimizacao ./producao -f arquivo_saida < teste.in | lp_solve

Value of objective function: 3755.31914894

Actual values of the variables:
p1                         212.766
p2                         957.447
p3                               0
→  t1_otimizacao cat arquivo_saida.out
2.80p1 + 3.30p2 + 1.20p3;

0.20p1 + 1.00p2 + 0.40p3 <= 1000.00;
0.50p1 + 0.10p2 + 0.20p3 <= 2000.00;
1.00p1 + 0.30p2 + 0.20p3 <= 500.00;
0.10p1 + 0.10p2 + 0.00p3 <= 2000.00;
```

7. Alguns exemplos de teste

1. **Caso em que há lucro (valor de venda é maior que o custo de produção)**

```
2 3
10 3
2 2000
3 3000
4 5000
0.8 0.7 1.0
1.0 0.8 0.9
```

```
max: 2.30p1 + -5.00p2;

0.80p1 + 1.00p2 <= 2000.00;
0.70p1 + 0.80p2 <= 3000.00;
1.00p1 + 0.90p2 <= 5000.00;
```

```
Value of objective function: 5750.00000000

Actual values of the variables:
p1                        2500
p2                           0
```

Nesse caso, pelo menos um dos coeficientes da função objetivo é positivo.

2. **Caso em que não há lucro nem prejuízo (valor de venda e custo de produção iguais)**

```
2 2
6 5
5 2000
10 3000
0.2 0.5
0.4 0.3
```

```
0.00p1 + 0.00p2;


0.20p1 + 0.40p2 <= 2000.00;
0.50p1 + 0.30p2 <= 3000.00;
```

```
Value of objective function: 0

Actual values of the variables:
p1                           0
p2                           0
```

Nesse caso, os coeficientes da função objetivo são nulos.

3. **Caso em que daria prejuízo caso a empresa desejasse produzir (valor de venda menor que o custo de produção)**

```
2 2
1 1
5 2000
10 3000
0.2 0.5
0.4 0.7
```

```
max: -5.00p1 + -8.00p2;


0.20p1 + 0.40p2 <= 2000.00;
0.50p1 + 0.70p2 <= 3000.00;
```

```
Value of objective function: 0

Actual values of the variables:
p1                           0
p2                           0
```

Nesse caso, todos os coeficientes da função objetivo são negativos.

8. Referências

- Livro "*Understanding and Using Linear Programming*", Jiří Matoušek, Bernd Gärtner - 2007.