

CI1238 - Otimização

Primeiro Trabalho Prático

14 de março de 2024

1 Introdução

O trabalho consiste em modelar e implementar, por programação linear, uma solução para o problema de “Transporte de carga com pacotes de recursos”.

A resolução do trabalho, ou seja, a descrição do problema, da modelagem e da implementação, deve estar em um texto claro em formato de um artigo e em pdf. Este texto deve conter o nome dos autores (alunos da equipe), uma introdução com o problema, a modelagem e sua explicação (de por que essa modelagem resolve o problema). Todas as referências que forem usadas devem estar citadas corretamente no texto.

Não se espera a implementação do método simplex. A tarefa principal de implementação consiste em gerar uma saída para ser usada pelo resolvidor `lp.solve`. Seu programa deve compilar e executar nas servidoras do DINF. A implementação deve estar descrita no texto que contém a resolução do trabalho e deve apresentar exemplos de uso e outras informações que os autores julguem necessário.

O trabalho deve ser entregue com um `makefile` de forma que ao digitar o comando `make` o executável `transporte` seja construído no diretório corrente.

Você deve entregar um arquivo compactado (no formato `tar.gz`) com os nomes dos alunos da equipe (ou logins) com os seguintes arquivos no diretório raiz:

- texto (em pdf);
- os fontes (podem estar em subdiretórios);
- `makefile`;
- exemplos usados no texto (podem estar em subdiretórios).

A entrega deve ser feita por e-mail para `andre@inf.ufpr.br`, em um arquivo compactado com todos os arquivos do trabalho, com assunto “Otimização-trabalho 1” (exatamente). O trabalho pode ser feito individualmente ou em dupla.

2 O problema

Transporte de carga com pacotes de recursos

Uma empresa tem uma fábrica, em uma cidade (origem), e um depósito, em uma outra cidade (destino). Esta empresa produz um tipo de produto que se vende em peso (toneladas).

Considerando que a empresa vai conseguir vender toda a sua produção e terá um ganho fixo (descontado o custo de produção) de p para cada tonelada transportada até o depósito, a empresa quer enviar o máximo possível de seu produto para o depósito (por dia).

Para isso precisa usar uma rede de transporte. Esta rede é formada por n cidades $\{1, 2, \dots, n\}$ (incluindo as cidades origem, 1, e destino, n) e m rotas (não importa o meio de transporte) ligando duas cidades. Cada rota $e = (i, j)$ liga as cidades i e j da rede e tem uma capacidade de carga diária limitada em c_e toneladas por dia.

Para atravessar uma rota é exigido um certo conjunto mínimo de recursos que a empresa deve ter. Ou seja, existe um conjunto de k recursos $\{1, 2, \dots, k\}$ e para uma rota e é necessário que a empresa tenha uma quantidade do recurso ℓ que seja maior ou igual a $r_{e\ell}$ por tonelada transportada. Após usar a rota e , essa quantidade do recurso é usada.

Os recursos só podem ser comprados pela empresa (no mercado de recursos) em pacotes. Cada tipo de pacote tem uma certa quantidade de cada recurso e tem um custo (em dinheiro). Ou seja, existe no mercado q tipos de pacotes, $\{1, 2, \dots, q\}$. Cada pacote de recursos u tem $s_{u\ell}$ do recurso ℓ e custa v_u .

O lucro final da empresa será o ganho da venda do produto menos o custo com os pacotes de recursos. Ou seja, se a empresa conseguir enviar W toneladas do produto e tiver gasto C na compra dos pacotes de recursos, terá um lucro de $pW - C$.

2.1 Formato de entrada e saída

Os formatos de entrada e saída, são descritos a seguir e devem ser usados a entrada e a saída padrões (STDIN e STDOUT).

A entrada é formada de um conjunto de números inteiros. Os números podem estar separados por 1 ou mais espaços, tabs ou fim de linha.

Entrada: Inicia com 5 números n , m , k , q e p indicando a quantidade de cidades, rotas, recursos, e pacotes e o ganho por tonelada transportada, respectivamente. Em seguida temos m linhas, cada uma com os dados de uma rota. Cada rota é descrita com $k + 3$ números indicando as cidades i e j (extremos da rota), a capacidade e os valores dos recursos necessários $r_{i,j,t}$, com $t = 1, 2, \dots, k$. Em seguida temos q linhas, cada uma com os dados de um pacote de recursos. Cada pacote é descrito com $k + 1$ números indicando o valor do pacote (v_u) e as quantidades de cada recurso $s_{u,t}$, com $t = 1, 2, \dots, k$.

Saída: um arquivo no formato de entrada do `lp_solve` com a descrição do programa linear que resolve o problema para a instância dada. O formato de entrada do `lp_solve` está descrito na URL abaixo:

<http://lpsolve.sourceforge.net/5.5/lp-format.htm>

2.2 Exemplo de entrada

Considere $n = 4$ cidades, $m = 5$ rotas, $k = 3$ recursos, $q = 2$ pacotes e $p = 100$ de ganho. As rotas e os pacotes são descritos na tabela abaixo:

recursos	rotas					pacotes		
	1, 2	1, 3	2, 3	2, 4	3, 4	1	2	
1	1	2	1	2	3	4	5	
2	1	2	2	1	4	2	2	
3	5	0	0	0	6	0	1	
	5	2	5	2	5	10	20	
			capacidades				custos	

O arquivo de entrada seria como abaixo.

```
4 5 3 2 100
1 2 5 1 1 5
1 3 2 2 2 0
2 3 5 1 2 0
2 4 2 2 1 0
3 4 5 3 4 6
10 4 2 0
20 5 2 1
```

Para este exemplo um plano ótimo tem ganho de 150, onde são comprados 5 pacotes do tipo 1 e passando 2 toneladas, de 1 para 3, de 3 para 2 e de 2 para 4 e 0 nas outras rotas.

2.3 Exemplo de entrada do `lp_solve`

Um exemplo, tirado de outro problema, pode ser visto abaixo.

$$\text{min : } 100x_{31} + 100x_{32};$$

$$x_{11} + x_{21} + x_{31} = 10;$$

$$x_{12} + x_{22} + x_{32} = 20;$$

$$x_{11} + x_{12} \leq 5;$$

$$x_{21} + x_{22} \leq 10;$$

$$x_{31} + x_{32} \leq 50;$$

$$x_{21} = 0;$$