UNIVERSIDADE FEDERAL DO PARANÁ

RELATÓRIO DE OTIMIZAÇÃO

Programação linear

Luan Marko Kujavski - GRR20221236

CURITIBA

2024

I. Introdução ao problema

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

II. O problema

Uma empresa possui uma fábrica em uma cidade (origem) e um depósito em 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 $\{1, 2, \ldots, n\}$ (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 ce 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 $\{1, 2, \ldots, k\}$, e para uma rota e é necessário que a empresa tenha uma quantidade do recurso que seja maior ou igual a re 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, existem no mercado q tipos de pacotes, $\{1, 2, \ldots, q\}$. Cada pacote de recursos u tem sua oferta do recurso e custa vu.

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 $pW - C$.

III. Resolução do problema

Começando pela função objetivo, o problema é maximizar os ganhos de vendas. Isso envolve maximizar o transporte e minimizar os custos de operações, isto é:

max carga * custo_carga - custo

Por exemplo, se o valor de venda de cada tonelada é 100, então a definimos:

max 100 * carga - custo

Agora, vamos dividir essa problema em dois subproblemas:

1. Transportar a maior carga possível
2. Minimizar os custos de operações.

O primeiro subproblema será resolvido seguindo o problema do transporte do livro de referência. Nessa problema, todo o peso que vem a um vértice deve sair dele.

Seja um vértice v que tem um vértice u de entrada. A carga pode passar de v a u ou de u a v. Dessa forma, denotamos que a carga máxima passada pela aresta uv pode ser positiva ou negativa

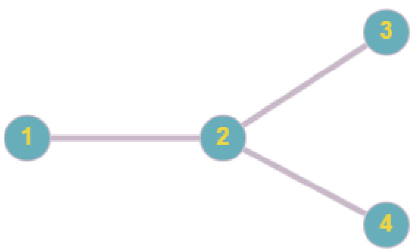
Por exemplo, seja a seguinte rota e seja a carga máxima de 5 toneladas

Definimos que:

-5 <= carga12 <= 5

Fazemos isso para cada uma das arestas da instância do problema.

Sendo assim, trabalhos com algo parecido a grafos não direcionados (já que os valores podem ser negativos). Por exemplo, temos que no seguinte grafo:



Definimos que:

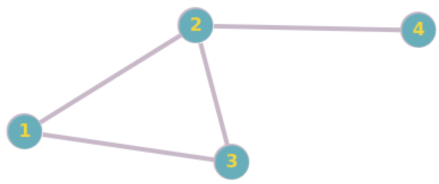
carga12 = carga23 + carga24

Isso porque, para resolver o problema do transporte, fazemos o seguinte:

Seja G um grafo montado a partir de dados de entrada. Para cada v = V(G), existe um número como label de v ∈ [1..V(G)]. Os vértices de entrada de v são os vértices cujas labels têm número menor que v e têm caminho até v. Os vértices de saída de v são os vértices cujas labels têm número maior que v e têm caminho saindo de v. Sendo assim, definimos que

A soma de todos os vértices u de entrada de v é igual à soma de todos os vértices s de saída de v.

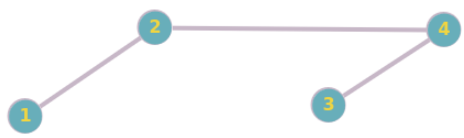
É importante ressaltar que se um vértice v não tem vértice de saída, então toda os vértices de entrada são igualados a 0. Por exemplo:



Nesse caso, seja v = 3, definimos o seguinte:

u13 + u23 = 0

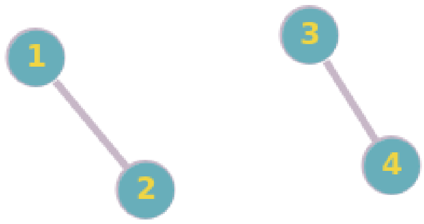
O mesmo acontece para o contrário. Seja um caso em que um vértice v não tem vértices de entrada mas tem vértices de saída, como o seguinte:



Nesse caso, seja v = 3, definimos que:

0 = s34

Também existe o caso em que o grafo é desconexo, como o seguinte:



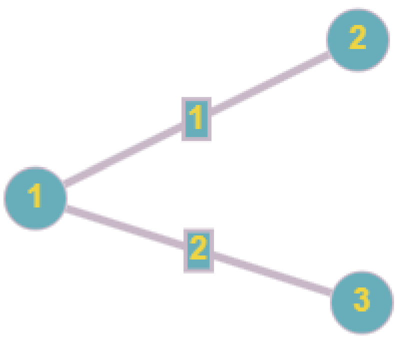
Nesse caso, a carga é zerada, já que:

seja v = 2, u12 = 0
seja v = 3, 0 = s34

Sendo assim, a função objetivo é zerada.

Já o segundo subproblema segue o problema da dieta do livro de referência, em que a ideia é atingir um valor mínimo de recursos minimizando o custo. Por exemplo:

Supondo que exista um gráfico não direcionado ponderado como o seguinte:



Supondo que existam dois pacotes disponíveis para somente um recurso e que as quantidades compradas de cada um são respectivamente x1 e x2.
Sendo também que o primeiro pacote dá 2 recursos e o segundo pacote dá 4 recursos. Ademais, os pesos de cada aresta denotam o número de recursos por unidade de carga necessários.

Definimos que:

2 * x1 + 4 * x2 >= 1 * carga12 2 * carga13

Isso quer dizer basicamente que o número de recursos comprados deve ser maior ou igual ao número de recursos usados.
Devemos repetir essa equação para todos os recursos que estão disponíveis. Para cada recurso, os pesos das arestas mudam, já que diferentes recursos têm diferentes valores mínimos para serem usados na rota e o valor do recurso que cada pacote entrega também.

No exemplo da especificação, temos 2 pacotes disponíveis e temos 3 recursos.
Sendo assim, devemos ter 3 equações com duas variáveis no lado esquerdo das equações, que é o valor comprado de cada um dos pacotes multiplicado pelo valor de cada recurso que consta em cada pacote. O lado direito das equações refere-se às rotas (problema do transporte), portanto não importam nessa explicação.

Definimos que:

4x1 + 5x2 >= carga12 + carga13 + carga23 + carga24 + carga34
2x1 + 2x2 >= carga12 + carga13 + carga23 + carga24 + carga34
0x1 + 1x2 >= carga12 + carga13 + carga23 + carga24 + carga34

Por fim, enfrentamos um último problema que é o módulo de um número em uma instância de problema de programação linear.

Para isso, seguimos as orientações do problema do sorvete do livro de referência. Na solução deste, substituímos a variável desejada por duas outras variáveis positivas. Por exemplo:

carga12 = a12 - b12

A partir disso, usamos a12 + b12 nas contas posteriores. No exemplo dado anteriormente, as três equações ficariam assim:

4x1 + 5x2 >= 1a12 + 1b12 + 2a13 + 2b13 + 1a23 + 1b23 + 2a24 + 2b24 + 3a34 + 3b34
2x1 + 2x2 >= 1a12 + 1b12 + 2a13 + 2b13 + 2a23 + 2b23 + 1a24 + 1b24 + 4a34 + 4b34
0x1 + 1x2 >= 5a12 + 5b12 + 0a13 + 0b13 + 0a23 + 0b23 + 0a24 + 0b24 + 6a34 + 6b34

Ao fazermos desse jeito, como a e b são positivas, ou a ou b tem o valor que queremos.

O motivo de precisarmos do módulo de um número é para que a carga (que pode ser negativa) não seja negativa ao ser multiplicada pelo ganho de cada tonelada de carga na função objetivo.

Por exemplo, na seguinte função objetivo:

max 100carga12 + 100carga13 - 10x1 - 20x2

Como carga12 e carga13 sempre serão positivas

Referências

1. Gärtner, B., Matousek, J.: Understanding and using linear programming. Universitext, Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-30717-4