

Análise e Síntese de Algoritmos

2º Projeto

Tomás Cunha, nº 81201, Grupo 15

1. Introdução

Este projeto tem como objetivo encontrar um ponto de encontro entre várias filiais de forma a minimizar o custo total das rotas, se existir. O problema pode ser reduzido a encontrar a menor soma dos custos dos caminhos mais curtos de todas as filiais para cada localidade, representando os caminhos como arestas de um grafo e os vértices como as localidades. Na resolução do problema utilizei a descrição do algoritmo de Johnson e da estrutura de dados Min-Heap disponíveis no livro *Introduction to Algorithms*[1].

2. Descrição da solução

A solução encontrada consiste em realizar uma variação do algoritmo Johnson tomando como vértices de fonte todas as filiais. Em vez de guardar todos os caminhos mais curtos numa matriz, é apenas guardada a soma dos caminhos até cada localidade num vetor, reduzindo o espaço ocupado. No final, este vetor é percorrido para encontrar a soma mínima. Após encontrar o ponto de encontro correto, é calculado o grafo transposto do original e realiza-se o algoritmo Dijkstra a partir do ponto de encontro, de forma a obter os custos individuais dos caminhos de cada filial até ao ponto de encontro.

O algoritmo pode ser representado em pseudocódigo da seguinte forma:

Algorithm 1: Descobrir o ponto de encontro, se existir

```
1 function get-result( $G, F, w$ )
2   foreach  $f \in F/G$  do
3     | reachable[f]  $\leftarrow$  true;
4     | sum[v]  $\leftarrow$  0;
5    $G' = G \cup s$ ;
6    $d[v] \leftarrow 0 \ \forall v \in V[G]$ ;
7   bellman-ford( $G', s, w$ );
8    $h(v) = \delta(s, v)$  calculado pelo bellman-ford  $\forall v \in V[G]$ ;
9    $w'(u, v) = w(u, v) + h(u) - h(v) \ \forall (u, v) \in E[G]$ ;
10  foreach  $u$  in  $F/G$  do
11    | dijkstra( $G, u, w'$ );
12    | foreach  $v \in \delta'(u, v)$  calculados por dijkstra do
13      | if  $\delta'(u, v) = \infty$  then
14        | | reachable[v]  $\leftarrow$  false;
15        | | sum[v]  $\leftarrow$  sum[v] +  $\delta'(u, v) + h(v) - h(u)$ ;
16  meeting-place  $\leftarrow$   $\min(\{s: \text{reachable}[s] \ \forall s \in \text{sum}\})$ ;
17  if meeting-place  $\neq \emptyset$  then
18    | paths = dijkstra( $G^T, \text{meeting-place}, w'$ );
```

Referências

- [1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, *Introduction to Algorithms*, 3rd Edition, September 2009