

Conceção e Análise de Algoritmos

9 de Abril de 2017

Tema 8: Routing multimodal para transporte coletivo

Turma 2MIEIC04

Grupo A

José Marques

Pedro Santos 201200764

Pedro Ferreira (2MIEIC05) 201103084

Explicação do Problema

O facto de nas grandes cidades o transporte individual ser demasiado utilizado, em detrimento dos Coletivos causa uma preocupação nos dias de hoje.

Um problema presente na maior parte das cidades com vários serviços de transportes públicos é o facto de não haver plataformas que facilitem ao utilizador a escolha do melhor itinerário tendo em conta os vários tipos de transporte. Estas plataformas teriam em conta os transportes com paragens em comum e os horários dos vários transportes e linhas diferentes.

Nesta plataforma, caberia ao utilizador apenas introduzir a origem e o destino do percurso. A plataforma calcularia os vários itinerários possíveis, e mostraria ao utilizador o itinerário multimodal mais curto/rápido.

Para avaliar o itinerário podem ser utilizados vários critérios: tempo de viagem, distância.

Descrição do Problema

Input

Construção de um grafo, $G = (V, E)$, de estações e linhas no qual:

- V – vértices – representam todas as paragens/estações dos meios de transportes disponíveis;
- E – arestas – representam todas as distâncias, tempos de viagens e custos;
- Nó de início de viagem e nó de destino.

Introdução de dados

Um ficheiro das estações com respetivas linhas/tipos de transportes que nelas passam, tempo médio entre estações tal como custo e duração.

Output

Todas as estações em que o utilizador passou por forma a otimizar o seu trajeto e o valor final da distância, custo ou duração da viagem dependendo da preferência do utilizador.

Objetivo

Facilitar aos utilizadores a escolha dos melhores trajetos consoante os critérios por estes preferidos.

Restrição No mesmo caminho não se passa pela mesma estação mais do que uma vez.

Formalização do Problema

Input

$G < V, E >$,

V : estações/paragens

E : ligações entre pontos

(tempo de viagem);

(custo da viagem);

(distância da viagem).

I : ponto inicial

F : ponto final

Output

$Caminho = \{V_i\}, I = 1 \dots n$

$Valor$

Objetivo

$Min(valor)$:

$$valor = \sum_{i=1}^n (E_{ij}), \quad ij \in Caminho$$

Restrição

$$\forall i \in \textit{Caminho} = \{V\}$$

$$\sim \exists j \in \textit{Caminho} : Vi = Vj \wedge i \neq j$$

Solução

Algoritmos

Depois de uma discussão chegamos à conclusão que o melhor algoritmo a ser usado neste trabalho é o **algoritmo de Dijkstra**, que encontra o caminho mais curto num grafo dirigido ou não dirigido em **tempo computacional $O([arestas + vértices]\log n)$** , este algoritmo não serve para grafos com pesos negativos, mas visto que este não é o caso, o algoritmo é perfeitamente aplicável.

Este algoritmo é um algoritmo ganancioso, tomando decisões que parecem óptimas no momento, determinando assim o conjunto de melhores caminhos intermediários. O valor de cada aresta está associado a uma classe com os campos “distância”, “preço” e “tempo”, portanto são estes valores que são utilizados para o cálculo de melhor caminho, dependendo das preferências do utilizador.

Esquema do programa

Começamos por introduzir dados de graphos, vértices (estações) e arestas (linhas). De seguida, agrupamos os graphos em dois tipos de estruturas de dados. Uma das estruturas é utilizada para representação gráfica e outra delas é utilizada para cálculos de rotas ideais, cálculos esses efetuados pelo algoritmo descrito em cima. As duas estruturas de dados apresentam ligeiras diferenças, de forma a tornar a representação gráfica mais “*user friendly*”.

Depois de carregar todos os dados, é altura do utilizador escolher o tipo de optimização de viagem, a estação de origem e a estação de destino, seguido da indicação de rota (diferente cor de arestas) na representação gráfica e uma lista das estações que constituem a rota otimizada.

```
O percurso pode ser calculado tendo como critérios:  
Escolha o criterio pelo qual deseja que o percurso seja calculado
```

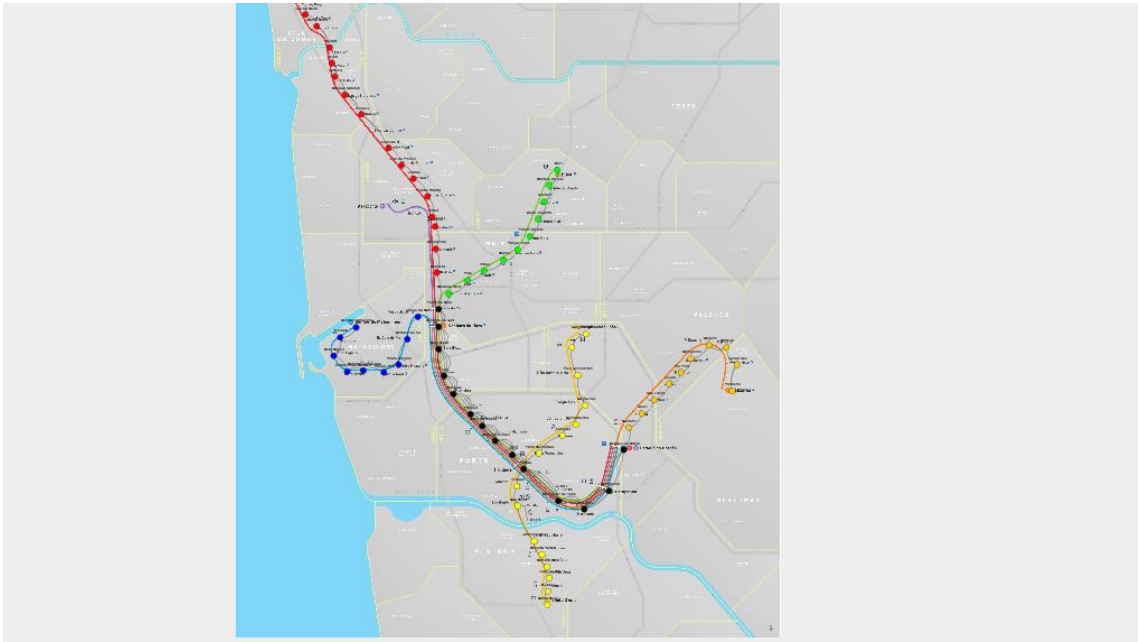
1. O tempo total de viagem
2. A distancia da viagem

```
109. Nsoni (linha E)  
110. Nau Vitoria (linha E)  
111. Levada (linha E)  
112. Rio Tinto (linha E)  
113. Campainho (linha E)  
114. Baguim (linha E)  
115. Carreira (linha E)  
116. Venda Nova (linha E)  
117. Fanzeres (linha E)
```

```
Escolhe o numero da estacao de origem: 1  
Escolhe o numero da estacao de destino: 4  
O caminho mais curto:  
Estadio do Dragao (linha A)  
Campanha (linha A)  
Heroismo (linha A)  
Campo 24 de Agosto (linha A)
```

```
A distancia percorrida e de cerca de: 4 minutos
```

```
<
```

Casos de Utilização

- Leitura de dados de ficheiros relativos a redes de transportes;
- Escolha do melhor percurso em termos de tempo de viagem;
- Escolha do melhor percurso em termos de distância percorrida;
- Visualização através do GraphViewer de toda a rede de transportes;
- Visualização do melhor percurso através do GraphViewer.

Métricas de avaliação

Avaliação empírica do seu desempenho

Para avaliar e testar a complexidade temporal dos algoritmos vamos medir os diferentes tempos de execução do programa utilizando diferentes dados de entrada.

Complexidade temporal

Como já foi referido anteriormente, a complexidade temporal do algoritmo que estamos a pensar utilizar é de $O([arestas + vértices] \log n)$.

Complexidade espacial

Na resolução do problema será utilizada programação dinâmica, que tem como complexidade temporal $O(|V| * T_{total})$.

Principais dificuldades

Ao longo da realização deste trabalho encontramos algumas dificuldades, nomeadamente com um problema com a compilação no IDE eclipse, para compilar o nosso ficheiro de conexão era necessário adicionar à linha de comandos do compilador: `-lws2_32`.

Também o facto de todos os elementos do grupo terem unidades curriculares diferentes e um ser trabalhador, nos dificultou a conjugação de horários para a realização do trabalho e como consequência só implementamos na nossa rede de transportes, o metro do Porto, pretendemos no entanto continuar a melhor o trabalho para a apresentação, se possível adicionando diferentes meios de transporte.

Conclusão

Toda a nossa rede de transportes foi baseada na cidade do Porto. Por questões de tempo apenas implementamos o metro do Porto, tendo em conta que a alteração de uma linha para a outra (transbordo) implica o tempo de deslocação entre estações e tempo de espera.

A realização deste trabalho serviu para ficarmos mais inteirados da matéria em questão, particularmente com o modo de funcionamento do Algoritmo de Dijkstra.

Quanto ao esforço todos os elementos do grupo se esforçaram igualmente para a sua realização.