

Concepção e Análise de Algoritmos

Distribuição de telecomunicações

Turma 2 - Grupo C

José Peixoto	200603103	ei12134@fe.up.pt
Pedro Moura	201306843	up201306843@fe.up.pt

26 de Abril de 2015

Conteúdo

1	Formalização do problema	3
1.1	Dados de entrada	3
1.2	Limites e condições de aplicação	4
1.3	Situações de contorno	4
1.4	Resultados esperados	4
2	Descrição da solução	4
2.1	Dijkstra	5
3	Métricas de avaliação	5
4	Lista de casos de utilização	5
5	Relato das principais dificuldades encontradas no desenvolvimento do trabalho	6
6	Indicação do esforço dedicado por cada elemento do grupo	6
6.1	José Peixoto	6
6.2	Pedro Moura	6
A	Map	7

Resumo

No âmbito da unidade curricular de Concepção e Análise de Algoritmos foi-nos proposto o desenvolvimento de uma aplicação para planificação da distribuição de fibra óptica numa aldeia que evidencie o conhecimento e uso adequado de algoritmos sobre grafos. Reúnem-se dados sobre coordenadas geográficas e distâncias através do sítio da Internet *OpenStreetmap*. Após a construção de um grafo que represente a área em estudo, são-lhe aplicados algoritmos que determinam a melhor solução para o problema na minimização da distância e dos custos da instalação de fibra óptica.

1 Formalização do problema

1.1 Dados de entrada

Mapeia-se uma área, usando coordenadas geográficas que representem intersecções entre ruas ou coordenadas de casas. Para este efeito, usam-se dados extraídos do OpenStreetMap¹ sobre a área seleccionada.

Construção de um grafo a partir de vértices-casa, cruzamento ou central V e arestas E que representam as distâncias entre as casas ligadas pelas ruas.

$$G < V, E > \quad (1)$$

Área máxima A em m^2 , do círculo com centro na central da companhia Telefones, resultando na exclusão das casas fora dessa selecção.

Pretende-se guardar os dados em dois ficheiros distintos, **vertexes.csv**, onde temos os números identificativos das casas e intersecções de ruas, vértices do grafo e **edges.csv** onde se estabelecem as relações entre os vértices do ficheiro anterior e respectivas distâncias, por forma a caracterizar as arestas do grafo.

Os vértices são guardados um por cada linha e são separados por ponto e vírgula dos valores posicionais x e y e de um valor que determina se o vértice é intersecção de rua - 0, casa - 1 ou central - 2.

Exemplo: 3;264;250;1 (o vértice número 3 é uma casa com coordenadas x e y de 264 e 250 píxeis respectivamente)

As arestas são guardadas uma por cada linha, sendo os primeiros dois valores os números identificativos dos vértices e o último a distância em metros entre eles.

Exemplo: 2;3;150 (o vértice número 3 dista 150 metros do vértice 2)

¹Mapa exemplo em anexo

1.2 Limites e condições de aplicação

Na lista final, não poderá existir repetição de pares de vértices, ou seja, arestas repetidas e todos os vértices-casa têm de estar presentes nesta lista.

Como as coordenadas e distâncias exemplo são escritos em ficheiros separados por vírgulas de forma manual, torna-se morosa a representação de uma área muito grande, com muitos vértices e arestas, dos quais resulte num grafo de maiores dimensões. Os resultados da aplicação também estão dependentes da qualidade, precisão e realismo dos valores das coordenadas e das distâncias.

Como requisito do algoritmo usado, referido na secção seguinte, os grafos alvo terão de ser obrigatoriamente não dirigidos, conexos e pesados.

1.3 Situações de contorno

1.4 Resultados esperados

A aplicação determinará as coordenadas dos vértices que definem as arestas seleccionadas para instalação da cablagem com fibra óptica e a distância total mínima, soma da distância destas arestas. O subconjunto das arestas resultante é acíclico e forma uma árvore de expansão mínima.

$$Output = Set < V_i, V_j > V_i, V_j \in V_s \quad (2)$$

A função objectivo retorna a distância mínima D do somatório das distâncias parciais $dist$ das arestas formadas pelos pares de vértices escolhidos.

$$D = \min(\sum_{i,j} dist(V_i, V_j)): V_i, V_j \in Output_k \quad (3)$$

2 Descrição da solução

Após análise cuidada do enunciado, concluiu-se que um método adequado na resolução do problema passa pela utilização de um algoritmo que determine a árvore de expansão mínima, uma vez que se pretende instalar fibra óptica em todas as casas de forma o mais eficiente possível, ou seja, com menor gasto de cabo de fibra óptica, reduzindo os custos para a empresa.

O grafo alvo é construído a partir da leitura de dois ficheiros: um com vértices e outro com arestas. São feitas verificações da correcção dos valores do *id*, coordenadas x e y e o tipo de intersecção. De igual modo, na leitura das arestas e em cada linha, são verificados os *id* do vértice fonte e do vértice destino e sua distância. Para garantir que se trata de um grafo não dirigido, na adição de cada aresta, (u, v) é igualmente adicionada uma segunda no sentido inverso (v, u) .

Antes de se processar o grafo na procura de uma árvore de expansão mínima, é verificada a sua conectividade. Um grafo é conexo quando existe

sempre pelo menos um caminho entre quaisquer dois vértices distintos, salvo a excepção de o grafo apenas conter um vértice.

2.1 Kruskal

3 Métricas de avaliação

Como se vai trabalhar com grafos de pequena dimensão, no teste da aplicação, julga-se necessária a criação de testes unitários que executem de forma repetida o mesmo código numa ordem de magnitude o suficientemente grande para se efectuarem medições com valores de múltiplos de segundo.

Para a medição da complexidade espacial, ter-se-á em conta a memória necessária para armazenar todas as variáveis criadas na execução do algoritmo e crescimento com o aumento da quantidade dos dados de entrada.

4 Lista de casos de utilização

A aplicação permite a leitura de vértices e arestas contidos em dois ficheiros e construção de um grafo não dirigido que represente um mapa de uma aldeia, com o qual se pode calcular uma árvore de expansão mínima que contemple todas as casas.

- Atribuição de uma área máxima em m^2 , limitando o número de vértices que são abrangidos na árvore de expansão mínima final.
- Adição de novas centrais e cálculo do posicionamento ideal.
- Sobre os grafos de entrada e saída é possível consultar graficamente informação acerca do número de vértices e arestas e a distância total em metros, soma de todas as arestas.
- Modo de visualização dos vértices com respectivos *id* e posição dada pelos valores x e y em píxeis e o seu tipo `INTERSECTION`, `HOUSE`, ou `CENTRAL`, 0, 1 e 2 respectivamente.
- Modo de visualização das arestas com respectivos *id* de origem e de destino e distância em metros como peso da aresta.
- Multi-plataforma, testado em sistemas Linux e Windows.

5 Relato das principais dificuldades encontradas no desenvolvimento do trabalho

Sentiram-se dificuldades na compreensão do enunciado e definição de fórmulas equivalentes aos dados de entrada pretendidos.

Ao contrário de uma utilização em Windows relativamente sem problemas, o código fornecido para Linux do visualizador de grafos não mantinha o processo pai vivo após o fecho da janela resultante da chamada `fork()`, pelo que esta parte teve que ser reescrita, instalando-se um `SIGCHLD handler` que faz uma chamada da função de sistema `waitpid()`.

6 Indicação do esforço dedicado por cada elemento do grupo

Planeou-se o esqueleto do projecto e implementou-se a leitura de ficheiros em conjunto.

6.1 José Peixoto

- Responsável pela interface gráfica e da linha de comandos.
- Incorporação do código desenvolvido nas aulas práticas.
- Verificação da conectividade dos grafos.
- Limpeza das intersecções desprezáveis.
- Autor do relatório.

Graph.h

6.2 Pedro Moura

- Adaptação do algoritmo de Kruskal.

N.B. Este relatório não foi escrito ao abrigo do novo Acordo Ortográfico.

A Map

