

Grafos**1º Semestre de 2018**

Trabalho – Entrega: / /18

Peso: 3 pontos

O Problema do Fluxo Máximo

Considere uma parte da Internet modelada por meio de um grafo orientado G em que cada vértice representa um computador, cada aresta (u, v) representa uma conexão de comunicação unidirecional de u para v , e o peso de cada aresta (u, v) diz respeito à largura de banda do canal, isto é, o número de bits que podem ser transmitidos de u para v por segundo. Se desejarmos uma transmissão em alta velocidade de um computador s (origem ou fonte) em G para outro computador t (destino) em G , a maneira mais rápida é dividir a transmissão em pacotes e fazer um roteamento através de G de acordo com o fluxo máximo, como mostra a Figura 1.

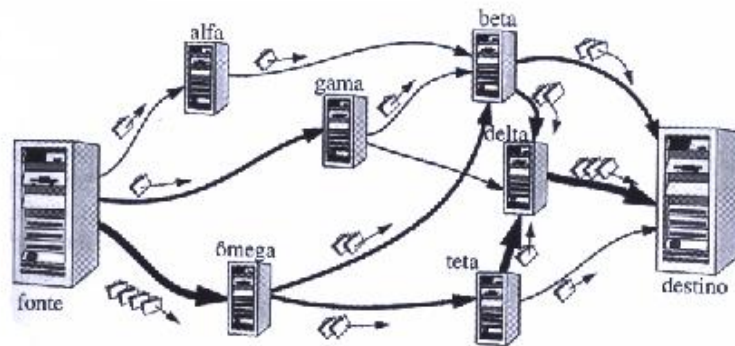


Figura 1. Um exemplo de fluxo em um grafo que representa uma rede de computadores.

Este trabalho tem por objetivo realizar a implementação do algoritmo de fluxo máximo em redes usando a busca em largura para encontrar o caminho aumentante na rede residual, bem como determinar o corte mínimo.

A Figura 2 ilustra um arquivo texto a ser usado como entrada do algoritmo. A primeira linha do arquivo indica o vértice origem de G . Na segunda linha existem dois valores. O primeiro, indica o número de vértices de G (note-se que este será o vértice destino). Já o segundo valor indica o número de arestas de G . As linhas seguintes do arquivo texto representam as arestas de u para v , bem como as respectivas capacidades.

A saída deve ser fácil de compreender. Para ganhar bônus (1 ponto extra) na nota, você deve implementar a conversão do arquivo de entrada (ver Figura 2) para o padrão de arquivos de grafo GraphML descrito [neste link](#). O GraphML é um padrão aberto e livre para uso tanto acadêmico quanto comercial.

```
1          //vértice origem
13 21      //vértice destino e número de arestas
1 2 3      //aresta de 1 para 2 e capacidade da aresta
1 3 6
1 4 3
2 5 3
3 6 1
3 8 3
3 9 1
3 12 3
4 7 2
5 8 1
5 10 1
6 5 1
7 9 2
7 11 3
8 10 4
8 11 2
9 10 2
9 11 1
10 13 5
11 13 5
12 7 2
```

Figura 2: Exemplo de arquivo de entrada.