

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

THIAGO BELL

### **Laboratório 3 - Maxflow**

Porto Alegre, 2017

## 1.1 Tarefa

Implementar o algoritmo de Ford-Fulkerson usando a estratégia de *fattest path* para o problema de fluxo máximo.

## 1.2 Implementação

Os algoritmos foram implementados em C++. A implementação de grafo usa listas de adjacências. Cada aresta armazena informações de capacidade e fluxo atual. Os valores dos *forward edges* e *backward edges* são calculados baseados nessas informações.

## 1.3 Ambiente de Teste

Os experimentos foram realizados usando um processador Intel i7 2600k acompanhado de 8 GiB de RAM. O sistema operacional utilizado foi Ubuntu Linux 16.10.

## 1.4 Análise de Complexidade do Algoritmo de Ford-Fulkerson

A complexidade teórica do algoritmo foi comparado com resultados experimentais. Conclui-se que a implementação respeita as previsões teóricas.

### 1.4.1 Complexidade

A complexidade do algoritmo depende do número de iterações que ele executa. Isso é, o número de vezes em que ele incrementa o fluxo. Supondo um limitante superior  $I$  para o número de iterações, a complexidade do algoritmo é de:

$$O((n \log n + m)I)$$

### 1.4.2 Formulação do Teste

Para testar o algoritmo gerou-se redes mesh com o gerador de *Washington* sempre mantendo o número de colunas igual ao de linhas. Executou-se o algoritmo para estas redes e comparou-se o tempo de execução com a previsão teórica. Para calcular-se o custo para uma determinada instância usou-se o número de iterações do algoritmo ao executar como limitante superior  $I$ .

### 1.4.3 Experimento e Comparação

Para redes com dimensões laterais variando de 100 até 500 em incrementos de 100, mediu-se o tempo de execução. A comparação destes tempos com o custo teórico pode ser vista na figura 1.1. Percebe-se que para todos os casos a razão entre esses dois valores permaneceu razoavelmente constante. Dessa forma, a implementação segue a complexidade do algoritmo.

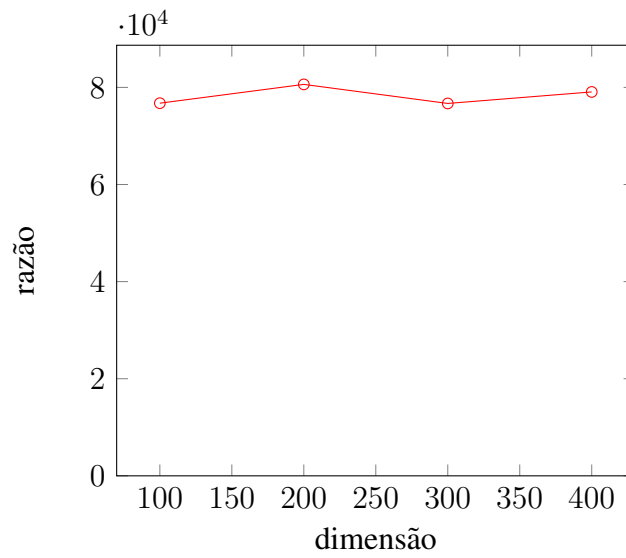


Figura 1.1: Mostra a razão entre o custo teórico esperado e o tempo de execução

## 1.5 Conclusão

Implementou-se o algoritmo de Ford-Fulkerson com a estratégia do *Fattest-Path*. Verificou-se que a implementação respeitava a complexidade do algoritmo.