

Fluxo de Redes

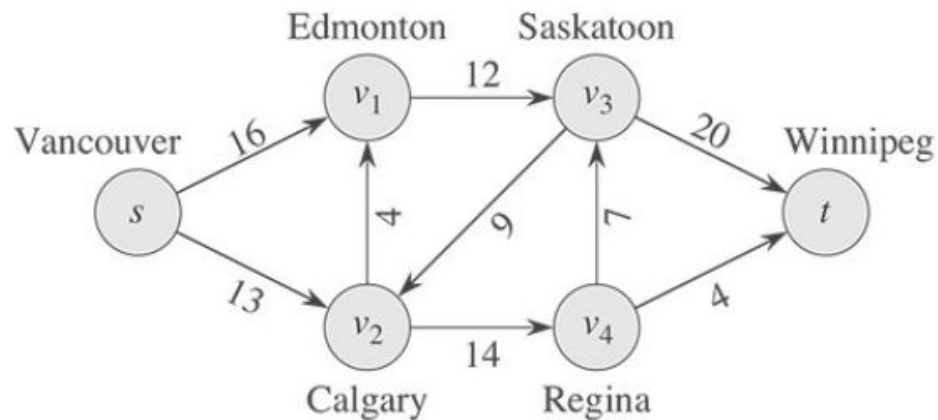
João Canavarro e Renan Cunha

Agenda

1. Rede de Fluxo
2. Fluxo e Fluxo Máximo
3. Aplicações
4. Método Ford-Fulkerson
5. Implementação

Rede de Fluxo

- Grafo conexo, direcionado e anti-simétrico.
- Cada aresta possui uma capacidade/taxa de transmissão positiva.
- Possui uma *fonte* e um *sumidouro*



Fluxo

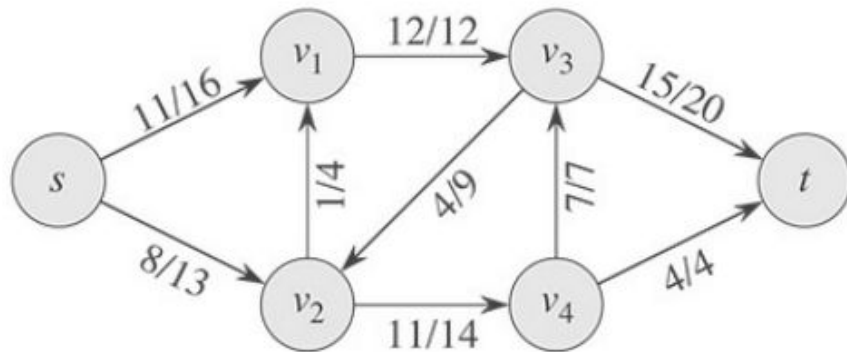
- Quantidade transmitida ou taxa de transmissão entre a fonte e o sumidouro em uma rede de fluxo.
- Restrição de capacidade:

$$0 \leq f(u, v) \leq c(u, v)$$

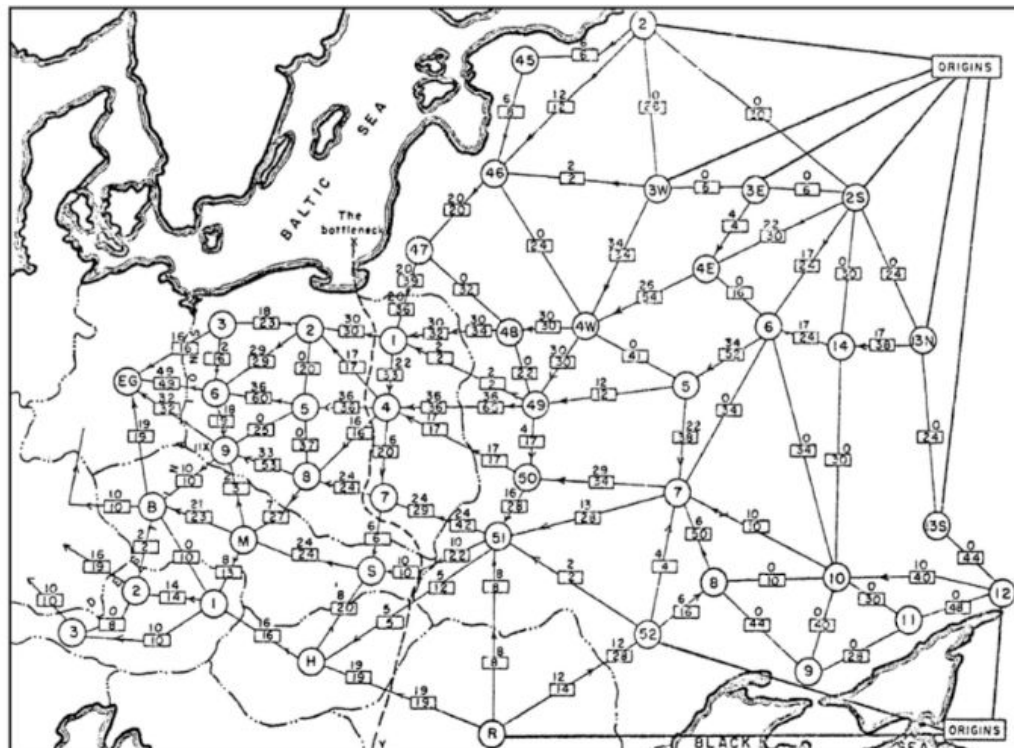
- Conservação de Fluxo:

$$\sum_{v \in V} f(v, u) = \sum_{v \in V} f(u, v)$$

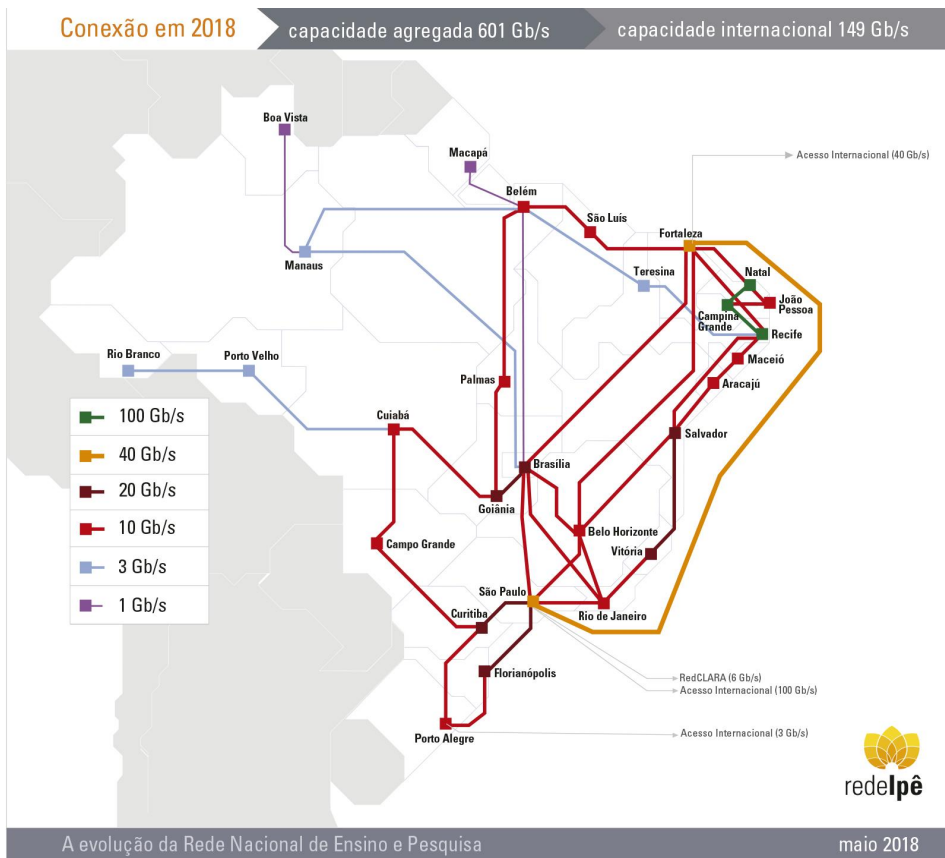
- **Fluxo máximo** é o número máximo de fluxo que a rede suporta.



Aplicações



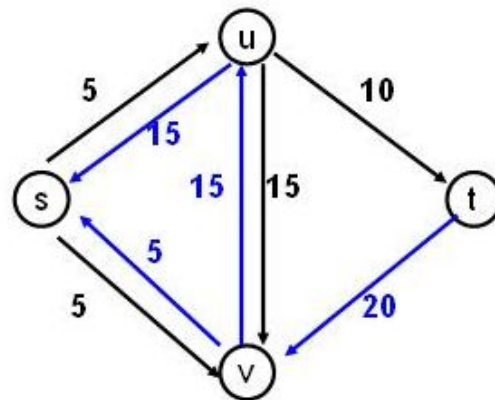
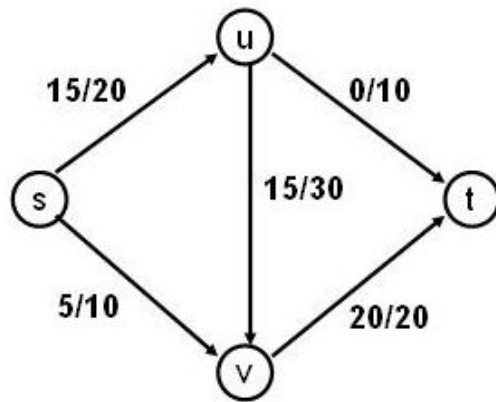
Aplicações



Método Ford-Fulkerson

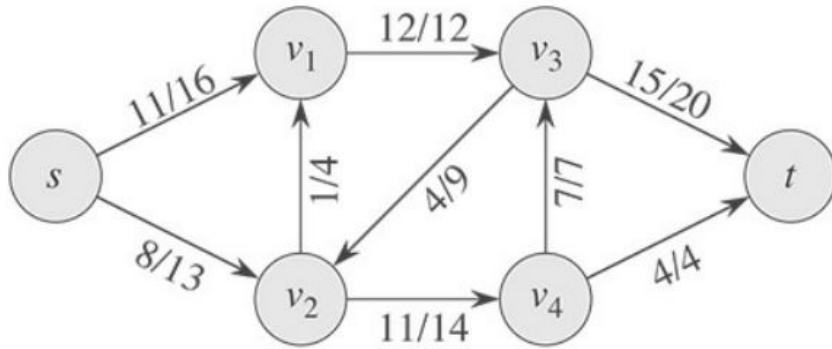
- Redes Residuais
- Caminho Aumentante
- Corte

Método Ford-Fulkerson - Redes Residuais

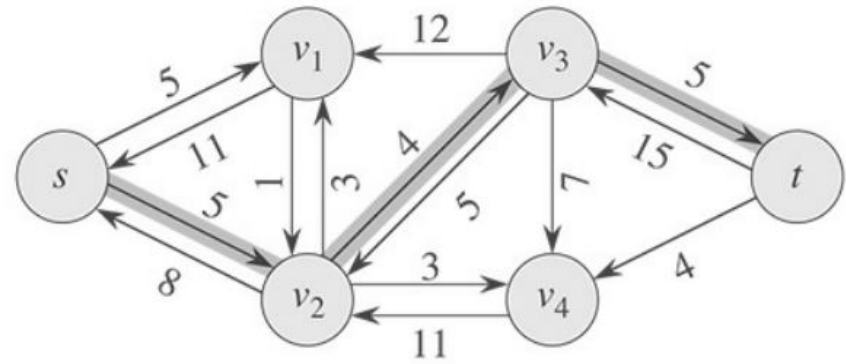


$$c_f(u, v) = c(u, v) - f(u, v)$$

Método Ford-Fulkerson - Caminho Aumentador



(a)



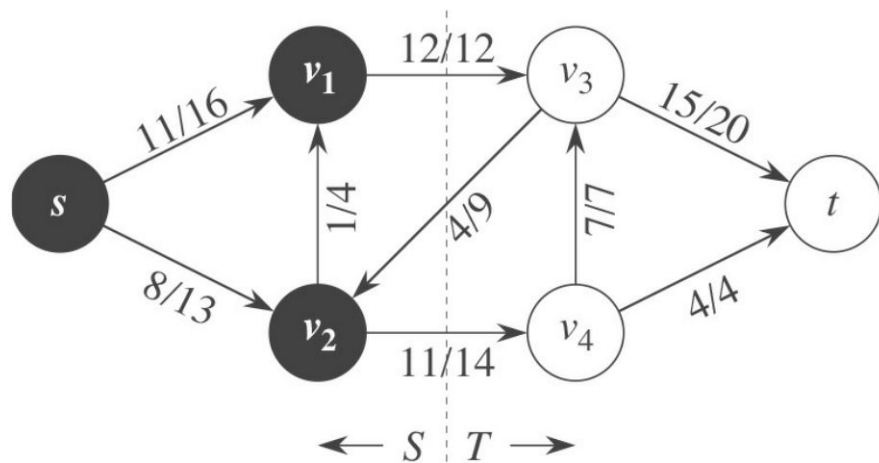
(b)

Pode incrementar o fluxo de acordo com a capacidade residual mínima

Método Ford-Fulkerson - Corte

Corte de rede de fluxo divide os vértices S e t .

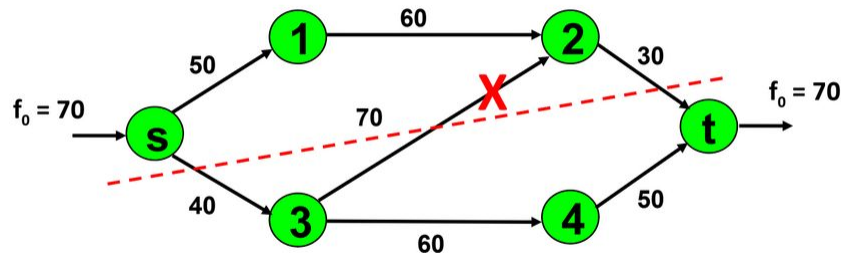
Capacidade do corte é a soma das capacidades das arestas de S para t .



Método Ford-Fulkerson - Corte Mínimo

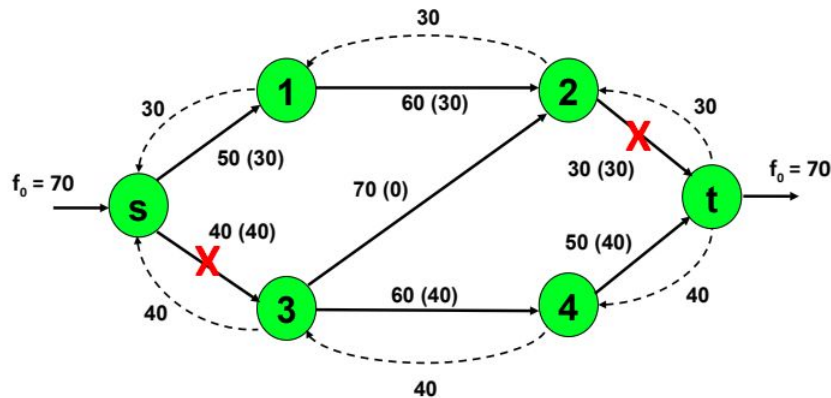
Um **corte mínimo** de uma rede é um corte cuja capacidade é mínima em relação a todos os cortes da rede.

O **teorema do fluxo máximo/corte mínimo** diz que o valor do fluxo máximo é igual a capacidade do corte mínimo.



Método Ford-Fulkerson - Corte Mínimo

O **teorema do fluxo máximo/corte mínimo** diz que um fluxo é máximo se e somente se sua rede residual não contém nenhum caminho aumentador.

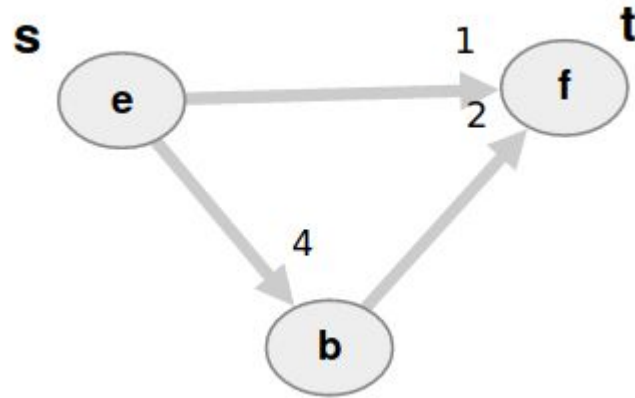


Método Ford-Fulkerson - Pseudocódigo

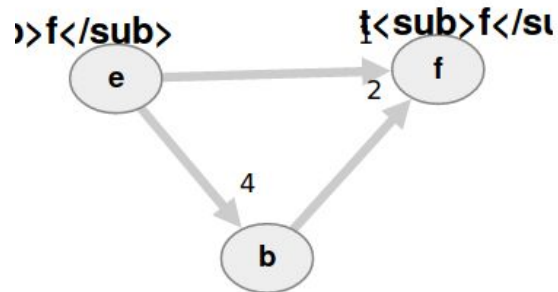
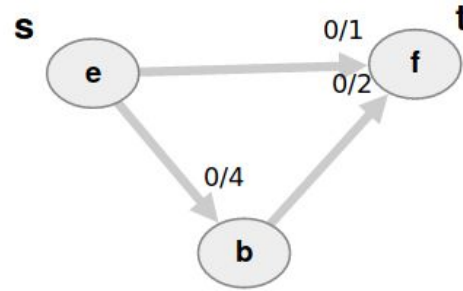
FORD-FULKERSON(G, s, t)

```
1  for cada aresta  $(u, v) \in G.E$ 
2       $(u, v).f = 0$ 
3  while existir um caminho  $p$  de  $s$  a  $t$  na rede residual  $G_f$ 
4       $c_f(p) = \min\{c_f(u, v) : (u, v) \text{ está em } p\}$ 
5      for cada aresta  $(u, v)$  em  $p$ 
6          if  $(u, v) \in E$ 
7               $(u, v).f = (u, v).f + c_f(p)$ 
8          else  $(v, u).f = (v, u).f - c_f(p)$ 
```

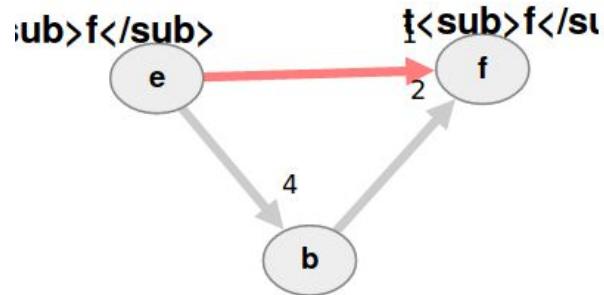
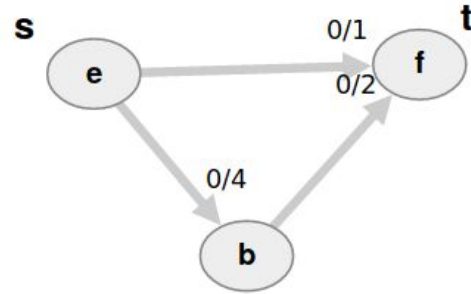
Método Ford-Fulkerson - Exemplo



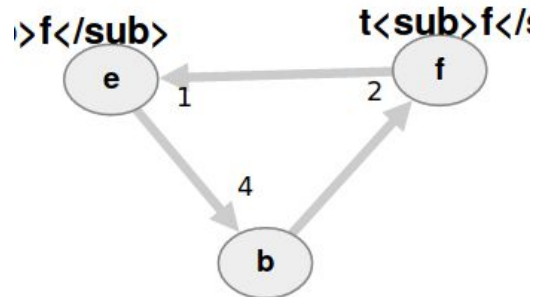
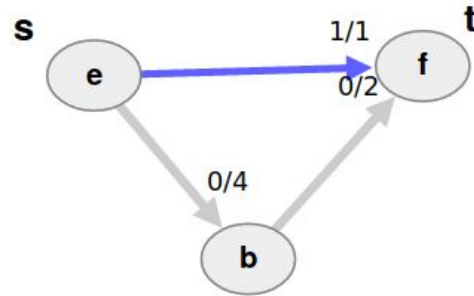
Método Ford-Fulkerson - Exemplo



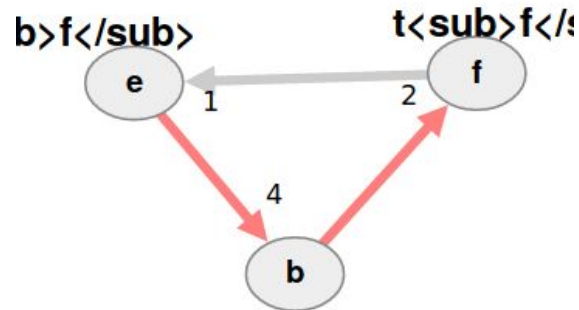
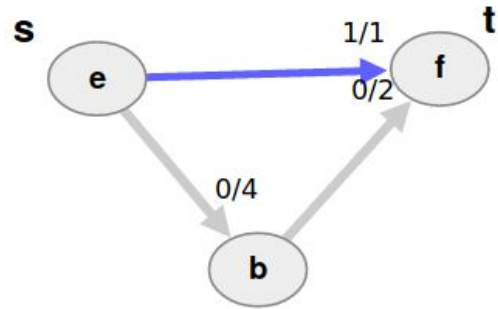
Método Ford-Fulkerson - Exemplo



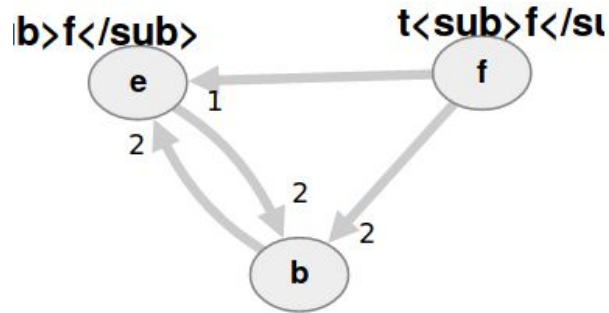
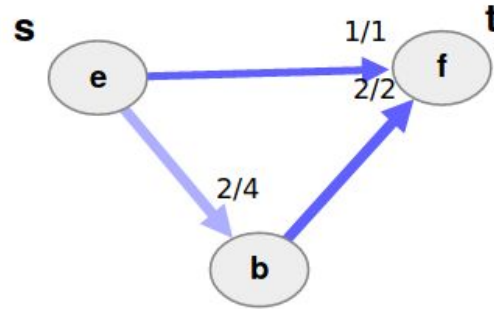
Método Ford-Fulkerson - Exemplo



Método Ford-Fulkerson - Exemplo



Método Ford-Fulkerson - Exemplo



Implementação

```
5  class FordFulkerson:
6      def __init__(self, graph: nx.DiGraph, source_vertice: int,
7                  sink_vertice: int):
8          self.graph = graph
9          self.source_vertice = source_vertice
10         self.sink_vertice = sink_vertice
11         self.residual_graph = graph.copy()
12
13         while nx.has_path(self.residual_graph, self.source_vertice,
14                           self.sink_vertice):
```

Implementação

```
13         while nx.has_path(self.residual_graph, self.source_vertice,
14                             self.sink_vertice):
15
16             # choose a path
17             path = next(nx.all_simple_paths(self.residual_graph,
18                                             self.source_vertice,
19                                             self.sink_vertice))
20
21             # take the minimal additional flow of this path and the edges
22             path_flow, path_edges = self.__min_residual_flow(path)
23
24             # update flow values, remove edge and add with new capacity
25             for edge in path_edges:
```

Implementação

[illegible]

Fim

1. Rede de Fluxo
2. Fluxo e Fluxo Máximo
3. Modelagem de Redes de Fluxo
4. Aplicações
5. Método Ford-Fulkerson
6. Implementação

Referências:

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. *Introduction to Algorithms, Third Edition* (3rd ed.). The MIT Press.