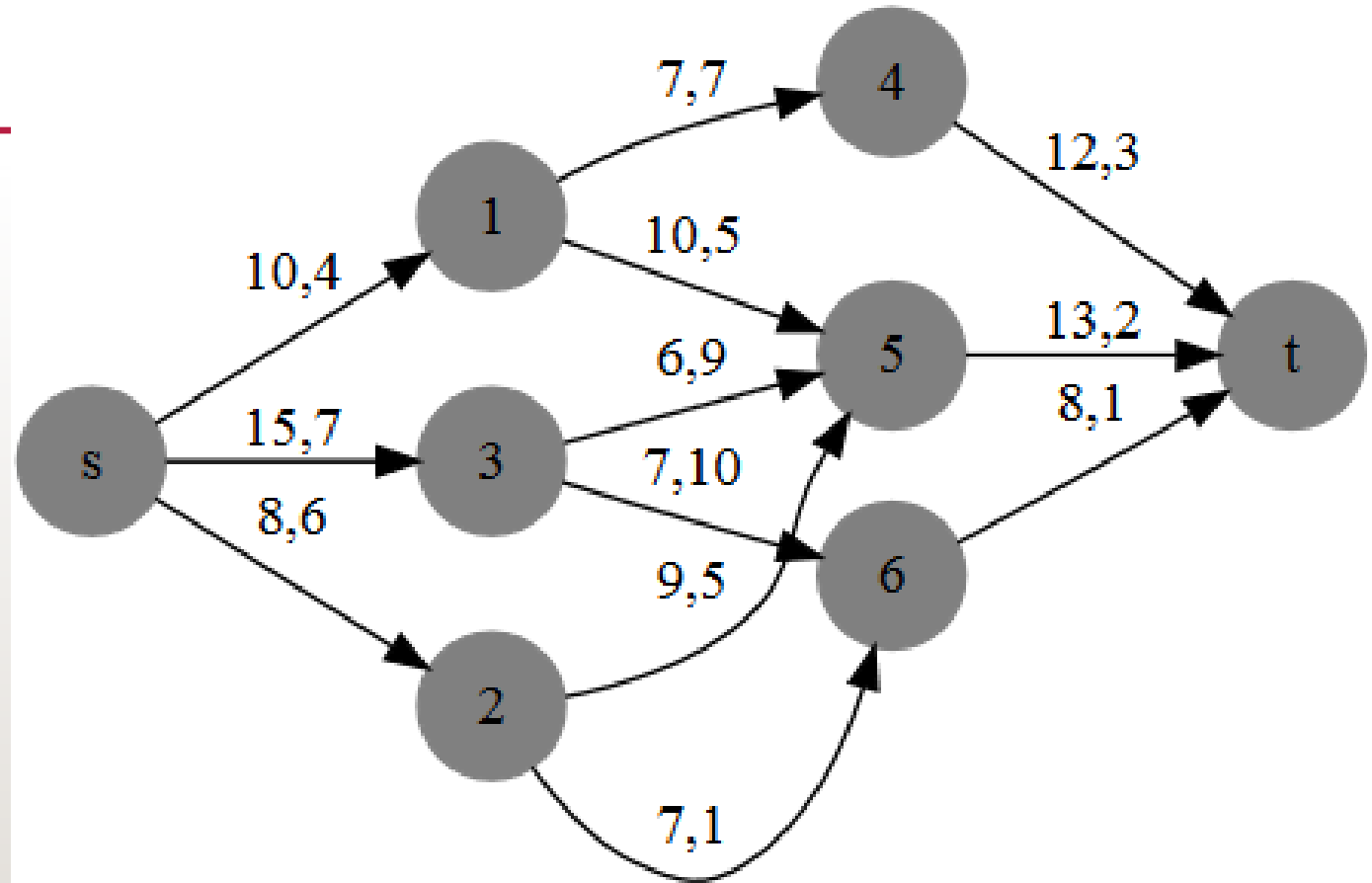


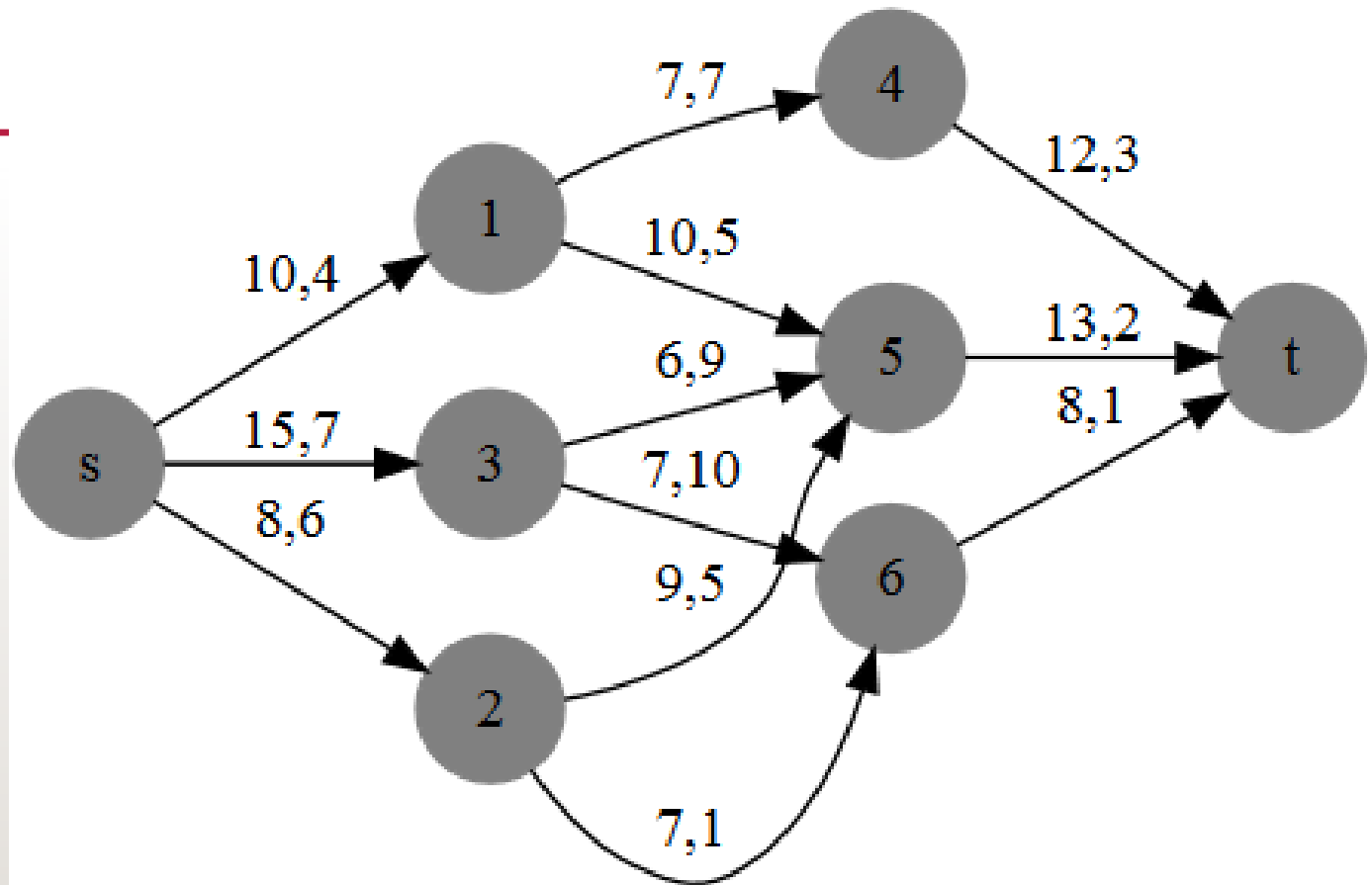
PROBLEMA DE FLUXO MÁXIMO A CUSTO MÍNIMO



-
- O problema corresponde a necessidade de circular um fluxo em rede s-t pagando o mínimo possível pelo tráfego.



- s: origem da produção
- t: consumidor
- Valores arestas:
Capacidade Fluxo da
aresta, custo na aresta



CARACTERÍSTICAS:

- Para podermos aplicar o algoritmo são necessárias algumas características:
- Não haver custo negativo após aplicada a função de custo(fc), para utilizar Dijkstra;
- Obs: Caso o algoritmo de caminho mínimo mude para Bellman-Ford não há a restrição sobre o valor em cada aresta.

PRÉ-REQUISITOS PARA ALGORITMO:

- O algoritmo para ser executado necessita de ter as seguintes informações:
- Vértice Produtor(s) e Vértice Consumidor(t)
- Grafo com informações de capacidade e custo nas arestas(G)
- Função de custo para instancia atual(fc).
- Um valor de fluxo a ser passado pelo grafo(fl)

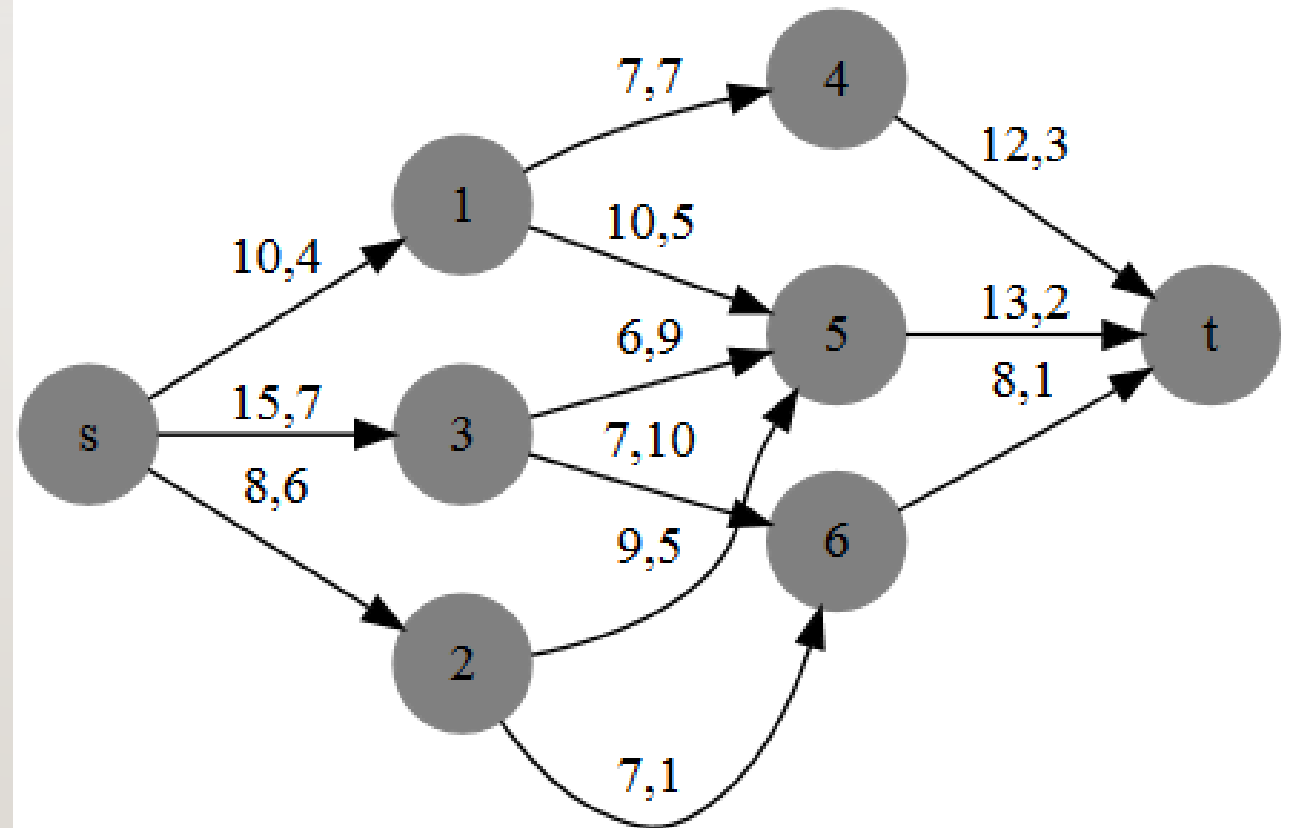
ALGORITMO:

- 1º passo:
 - Inicializar capacidade de atual como a capacidade total da aresta.
 - Definir o Grafo Residual(G_r), o vetor de pais(pais) em estado inicial
- 2º passo:
 - Verificar se existe caminho de aumento de custo mínimo(p) em G_r
 - O caminho será dado por uma aplicação do algoritmo de Dijkstra.
 - Subtrair a fluxo máximo desse caminho da capacidade atual de cada aresta do caminho.
 - Atualizar Grafo Residual.

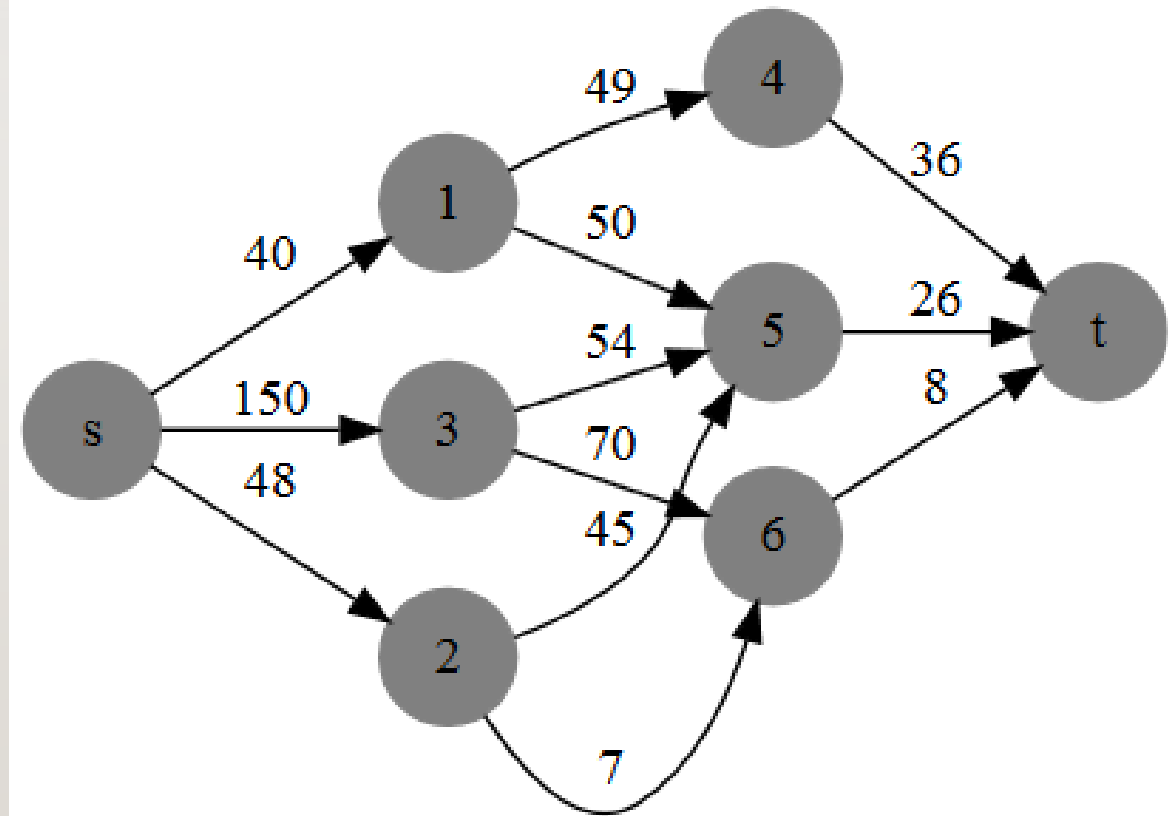
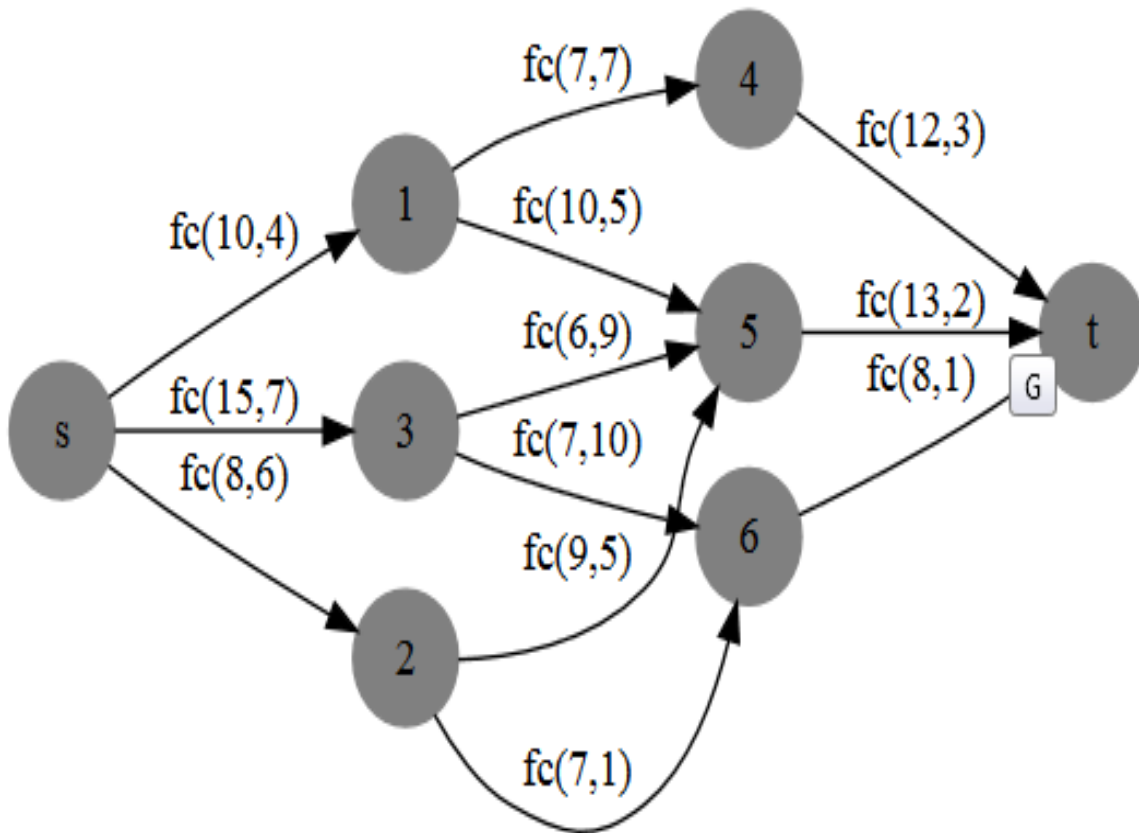
EXEMPLO EXECUÇÃO:

- Estado Inicial: Grafo Residual
 - $\text{fluxo_maximo} = 0$
 - $\text{fluxo_caminho} = 0$
- $\pi = [-1, -1, -1, -1, -1, -1, -1, -1]$

Obs: fc será capacidade * custo

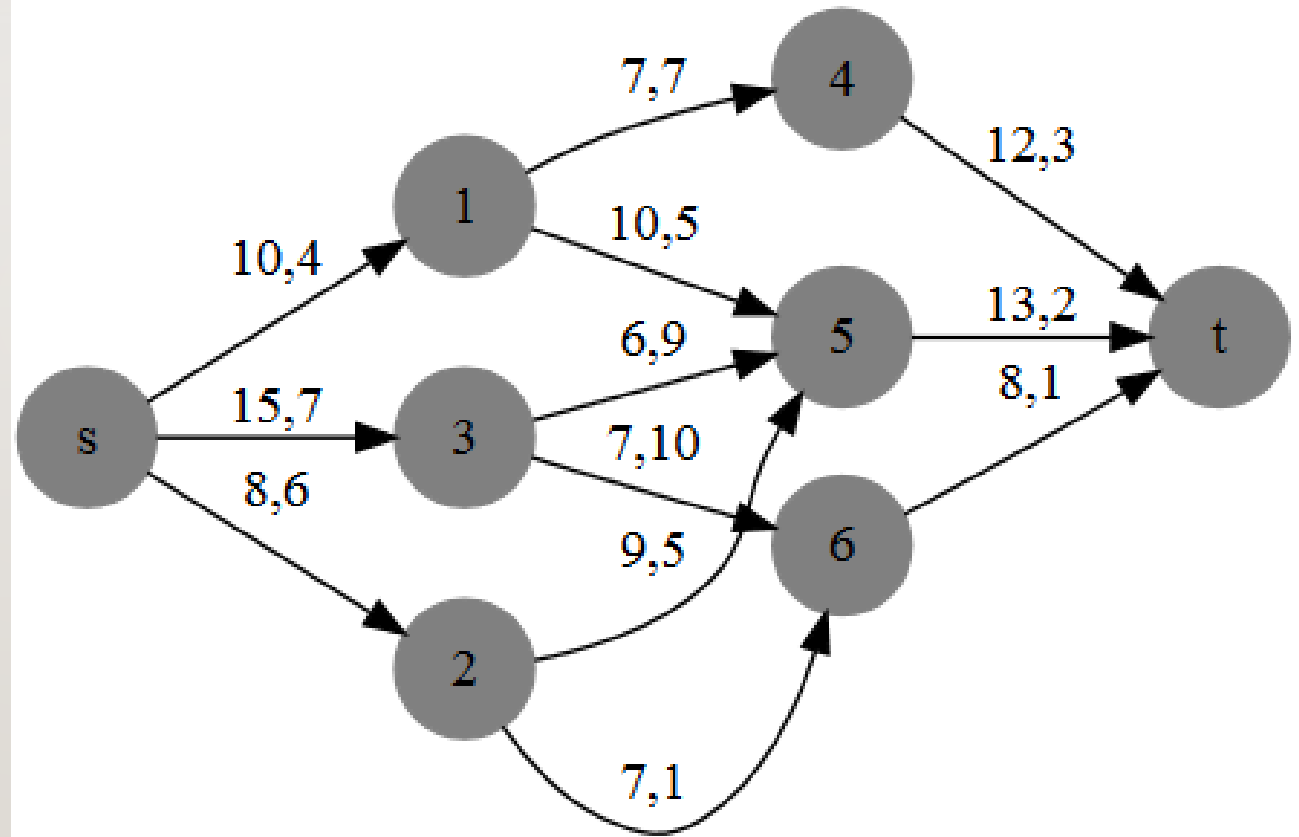


TRANSFORMAÇÃO DO GRAFO PARA BUSCA DE CAMINHO MÍNIMO

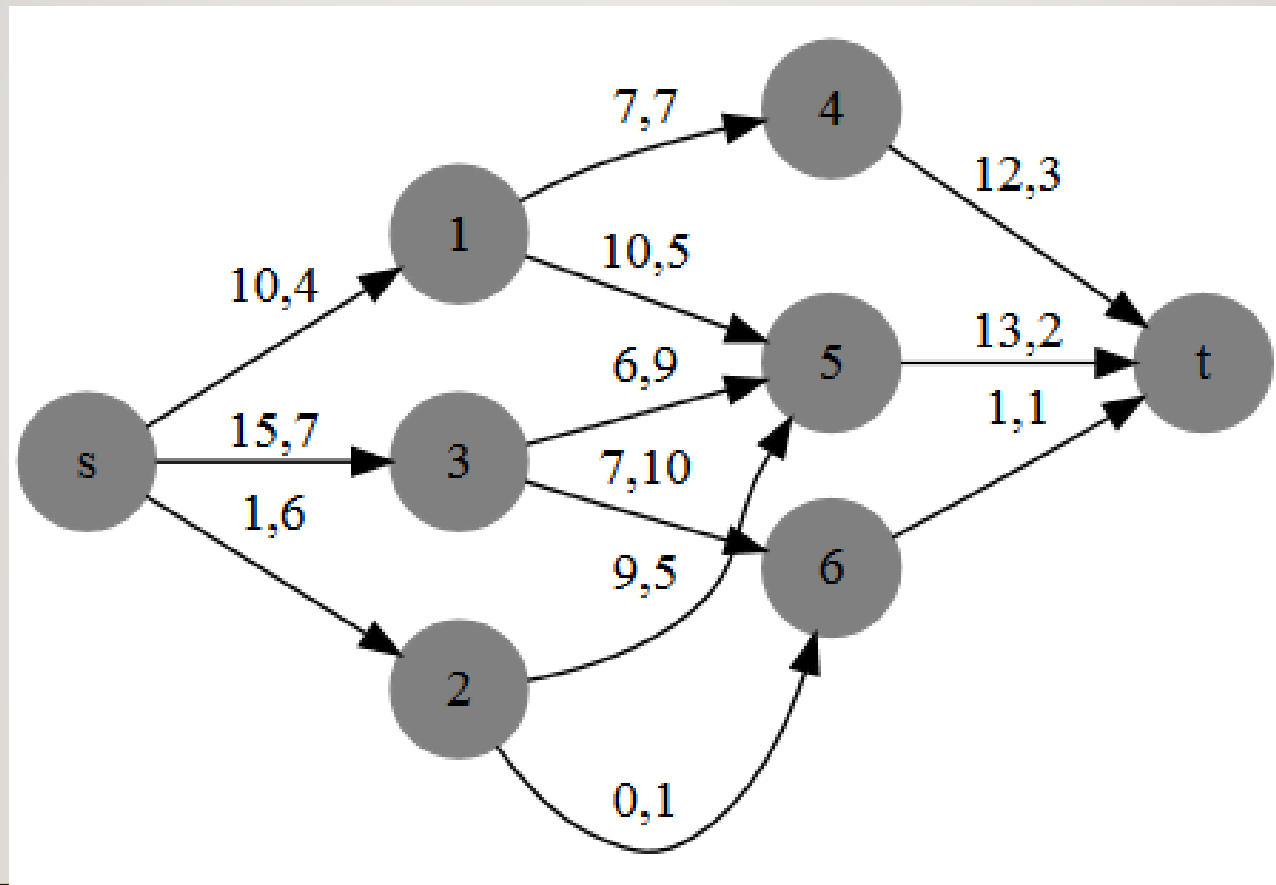


BUSCA CAMINHO MÍNIMO:

- Estado Atual: Grafo Residual
- $\text{fluxo_maximo} = 0$
- $\text{fluxo_caminho} = 7$
- $\pi = [-1, 0, 0, 0, 1, 1, 2, 6]$

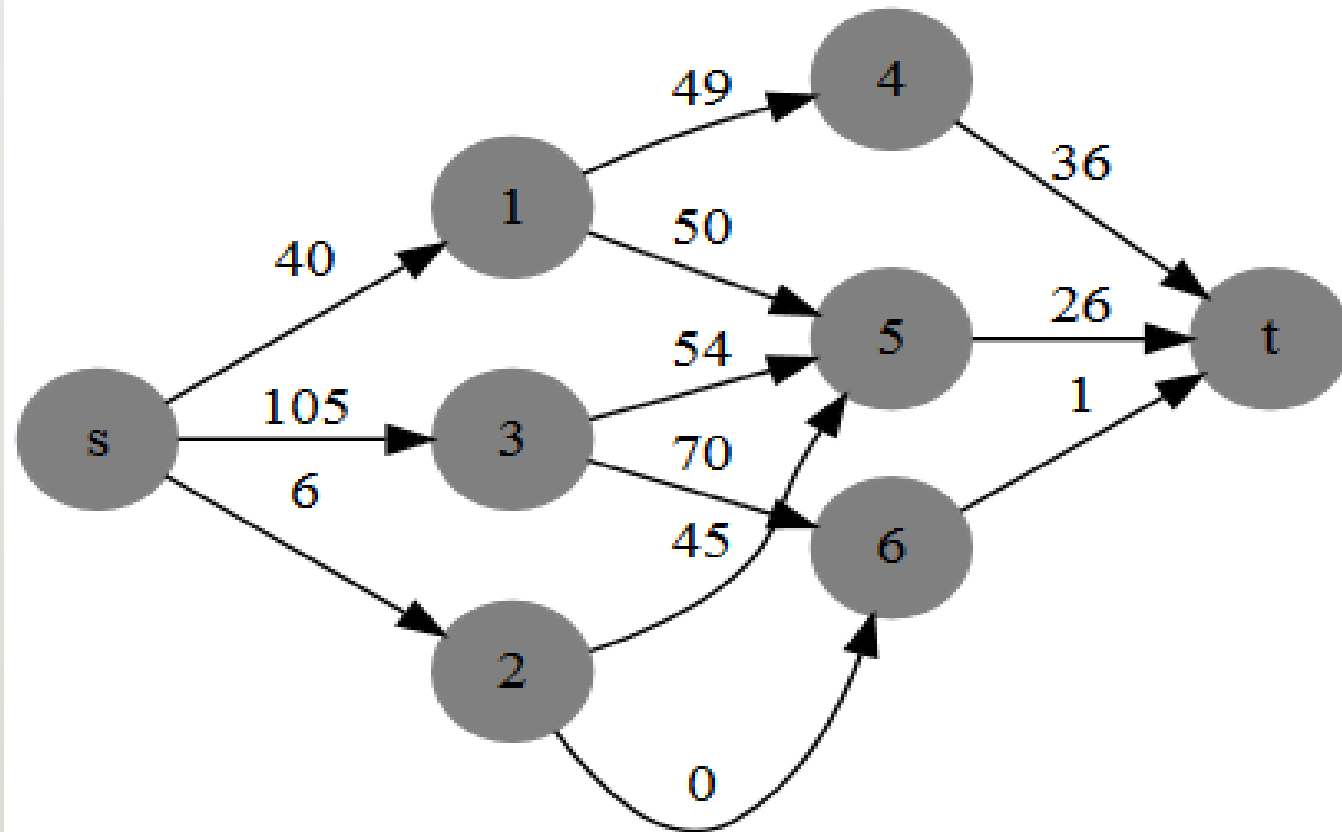


ATUALIZANDO GRAFO RESIDUAL:

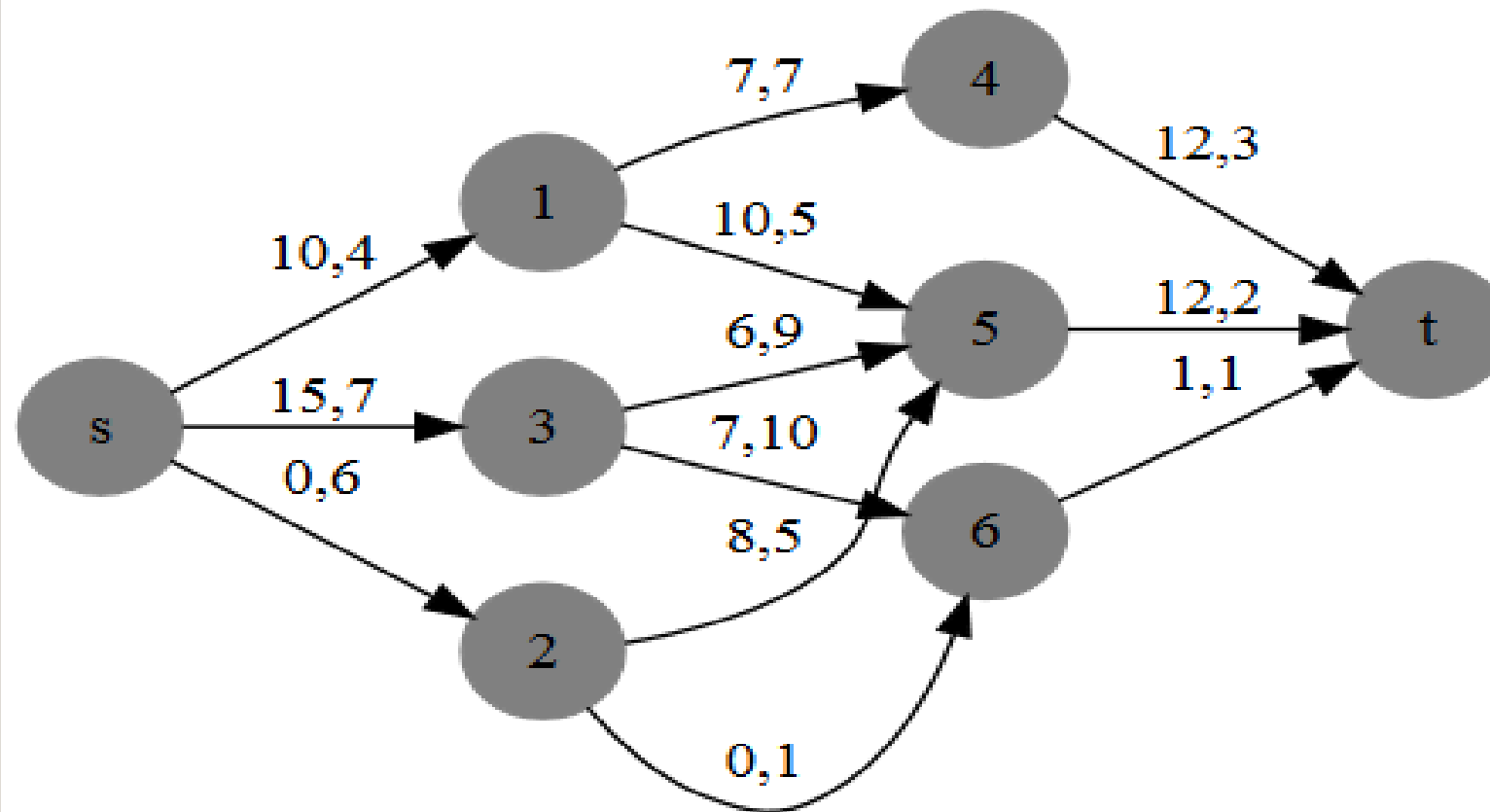


BUSCA CAMINHO MÍNIMO:

- Estado Atual: Grafo Residual
- $\text{fluxo_maximo} = 7$
- $\text{fluxo_caminho} = 1$
- $\pi = [-1, 0, 0, 0, 1, 2, 3, 5]$



GRAFO RESIDUAL ATUALIZADO:



ANALISE DE COMPLEXIDADE:

- A complexidade do algoritmo é dada por Ford-Fulkerson e Dijkstra:
- Complexidade: Ford-Fulkerson * Dijkstra
- Portanto: $|A| * (|V| * |V|)$

POSSÍVEIS TRANSFORMAÇÕES DE PROBLEMA:

- O problema de Fluxo Máximo a Custo Mínimo pode ser transformado em outros dois problemas de forma fácil:
- Fluxo Máximo: Caso os custos em todas as arestas passe a ser zero
- Caminho Mínimo: Existe duas formas:
 - Passar uma meta menor que a menor capacidade de uma aresta no grafo.
 - Colocar que a capacidade de todas aresta é infinito.

APLICABILIDADE:

- O problema de Fluxo Máximo de Custo Mínimo é aplicável em problemas de distribuição onde o objetivo é minimizar o custo final:
- Exemplos:
 - Distribuição de detritos;
 - Distribuição de água;