MO417 – Ata do Exercício 26.2-8

Fabian N.C. van 't Hooft

17 de junho de 2010

Enunciado

Mostre que um fluxo máximo em uma rede $G = (V, E)$ sempre pode ser encontrado por uma sequência de no máximo $|E|$ caminhos em ampliação. (*Sugestão:* Determine os caminhos *depois* de encontrar o fluxo máximo).

Resolução

As subfunções apresentadas aqui somente explicam em forma textual o funcionamento:

FIND-PATH-WITH-POSITIVE-FLOW(G, s, t)
    return a path from $s, t \in E[G]$ where flow larger then zero

GET-EDGE-WITH-MIN-FLOW(P)
    return the edge with the lowest flux, given a set of edges

NR-OF-PATHS($G, s, t, maxflow$)

```
c ← 0
mf ← maxflow
while mf > 0
      P ← FIND-PATH-WITH-POSITIVE-FLOW(G, s, t)
      minEdge ← GET-EDGE-WITH-MIN-FLOW(P)
      for each edge (u, v) ∈ P
            edge.flow ← edge.flow − minEdge.flow
      mf ← mf − minEdge.flow
      c ← c + 1
return c
```

No algoritmo apresentado o parâmetro maxflow é f o fluxo máximo do grafo G. A Figura 1 mostra os passos do algoritmo.

Corretude

Está garantido que, após cada iteração do laço-WHILE (linha 3), uma aresta que era capaz de carregar um fluxo é desconsiderada na função **FIND-PATH-WITH-POSITIVE-FLOW**. A garantia esta na linha 7. Nós podemos afirmar que após no máximo $|E|$ iterações todos os caminhos foram encontrados. Portanto o ultimo caminho s, v_1, v_3, v_t invalida as ultimas arestas que foram capazes de carregar um fluxo com valor 11. A variável mf é inicializada com o maxflow, 23 no caso do nosso exemplo. Cada fez que um caminho é encontrado mf é diminuído por o valor $minEdge.flow$ (Linha 8). O contador c é inicializado com zero para informar o numero de caminhos que foram encontrados após a terminação do procedimento (Linha 9).

Exemplo

Seguinte figuras mostram os passos. Cada caminho que foi encontrado diminui o fluxo por o valor $minEdge.flow$.

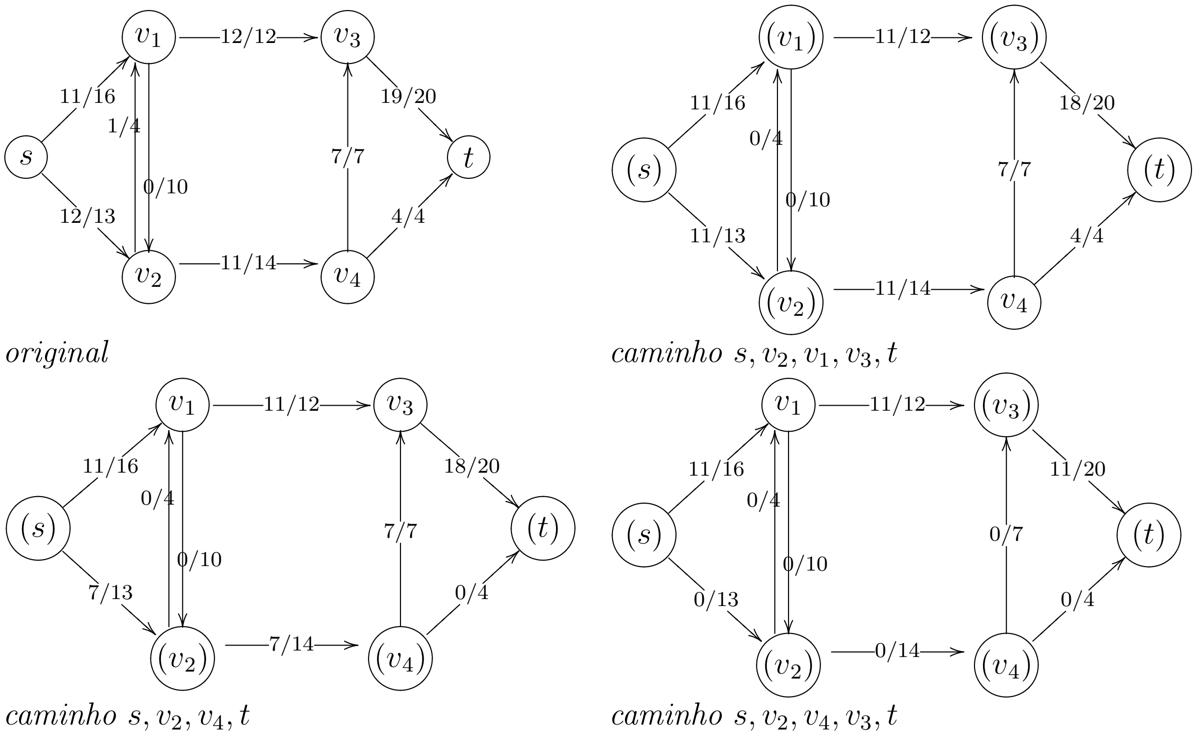


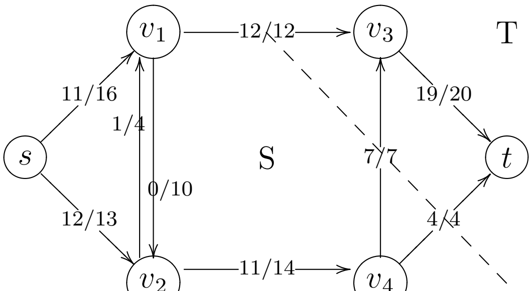
Figura 1: Exemplo de passos para o algoritmo.

O resultado do algoritmo é exato para a entrada mostrado na Figura 1.

Caminho	Fluxo-minimo
s, v_2, v_1, v_3, t	1
s, v_2, v_4, t	4
s, v_2, v_4, v_3, t	7
s, v_1, v_3, v_t	11
Fluxo total	1+4+7+11 = 23

Discussão

O algoritmo apresentado anteriormente resolve o problema enunciado, mas na aula havia uma discussão sobre a corretude.



O caminho s, v_1, v_3, v_t através do corte(S,T) pode carregar até 12 unidades. O algoritmo anteriormente enunciado só calcula 11. O caminho s, v_1, v_3, v_t compartilha uma parte do caminho s, v_2, v_1, v_3, t. Portanto existem múltiplas soluções dentro das possibilidades do conjunto S. No corte mínimo os fluxos são bem definidos.

Uma tentativa foi feita no quadro pelo colega Pedro, iniciando com o caminho s, v_2, v_1, v_3, t atribuindo um fluxo máximo através do caminho aumentante. Isto não funcionou. Ordenando os caminhos através do fluxo-minimo em ordem decrescente faz o nosso exemplo funcionar.

Caminho	Fluxo-minimo
s, v_1, v_3, v_t	12
s, v_2, v_4, v_3, t	7
s, v_2, v_4, t	4
s, v_2, v_1, v_3, t	Não pode aumentar o fluxo - descarto