MO417 – Ata do Exercício 24.1-4

Ewerton Almeida Silva

6 de junho de 2010

Enunciado: Modifique o algoritmo de Bellman-Ford de modo que ele defina $d[v]$ como $-\infty$ para todos os vértices v para os quais existe um ciclo de peso negativo em algum caminho a partir da origem até v.

Solução: A modificação efetuada no algoritmo de Bellman-Ford (Algoritmo 1) foi pequena. Ela consistiu em se alterar, conforme pedido, o valor dos vértices para os quais, após todos os relaxamentos, um ciclo negativo fora achado num caminho a partir da origem até os mesmos. No geral, as alterações efetuadas foram as seguintes:

I A variável booleana *ciclo* foi acrescentada ao algoritmo. Ela guarda a informação relativa a ocorrência de ciclos negativos no grafo, assumindo o valor *true* neste caso. O objetivo desta variável é a manutenção do retorno do algoritmo de Bellman-Ford, que retorna *true* se o grafo não possui nenhum ciclo de peso negativo, e *false* caso contrário.

II Porque o valor $d[v]$ de alguns vértices poderá ser redefinido como $-\infty$, o professor Meidanis sugeriu uma modificação no algoritmo de forma que somas com esses valores sejam evitadas. Assim, antes de qualquer comparação de $d[v]$ com a adição entre $d[u]$ e $w(u, v)$ (o peso da aresta que liga o vértice u ao vértice v), o algoritmo checa se $d[u]$ é igual a $-\infty$. Em caso afirmativo, não realizamos a comparação e por consequência a soma não é efetuada, e ainda temos certeza de que $d[v]$ assumiu o valor $-\infty$ previamente (pois é v alcançável através de u).

III Quando identificamos um vértice v que deve ser marcado com $-\infty$, "setamos" *ciclo* como *false* e referenciamos o procedimento CAMINHO(G, v) que altera $d[v]$ para o valor em questão, além de efetuar outras tarefas. A seguir, é explanada a idéia geral por trás deste novo procedimento.

O procedimento CAMINHO(G, v) (Algoritmo 4) foi criado com a finalidade de "propagar" recursivamente o valor $-\infty$ para todos os vértices que são alcançáveis por meio de um caminho com um dado ciclo negativo entre os vértices u e v. Em suma, para um vértice v identificado como sendo alcançável por um caminho detentor de um ciclo negativo, definimos $d[v]$ como $-\infty$ e chamamos o procedimento recursivamente para todas as arestas (v, p) (em outras palavras, para todos os vértices adjacentes a v) nas quais $v[p] \neq -\infty$, repetindo o processo de maneira recursiva para p e assim por diante.

Algoritmo 1 MODIFIED-BELLMAN-FORD(G, w, s)

INITIALIZE-SINGLE-SOURCE(G, s)
for $i \leftarrow 1$ **to** $|V[G]| - 1$ **do**
    for cada aresta $(u, v) \in E[G]$ **do**
        RELAX(u, v, w)
    end for
end for
$ciclo \leftarrow false$
for cada aresta $(u, v) \in E[G]$ **do**
    if $d[u] \neq -\infty$ **and** $d[v] > d[u] + w(u, v)$ **then**
        $ciclo \leftarrow true$
        CAMINHO(G, v)
    end if
end for
return not *ciclo*

Algoritmo 2 INITIALIZE-SINGLE-SOURCE(G, s)

for cada vértice $v \in V[G]$ **do**
    $d[v] \leftarrow \infty$
    $\pi[v] \leftarrow$ NIL
end for
$d[s] \leftarrow 0$

Algoritmo 3 RELAX(u, v, w)

if $d[v] > d[u] + w(u, v)$ **then**
    $d[v] \leftarrow d[u] + w(u, v)$
    $\pi[v] \leftarrow u$
end if

Algoritmo 4 CAMINHO(G, v)

```
d[v] ← −∞
for cada aresta (v, p) ∈ E[G] do
    if d[p] ≠ −∞ then
        CAMINHO(G, p)
    end if
end for
```