Graph algorithms - Bellman-Ford algorithm

Problem

Given a graph with no negative costs cycles and two vertices *s* and *t*, find a minimum cost walk from *s* to *t*.

Idea

The algorithm keeps two mappings:

- *dist[x]* = the cost of the minimum cost walk from *s* to *x* known so far
- *prev[x]* = the vertex just before *x* on the walk above.

Initially, *dist[s]*=0 and *dist[x]*=∞ for $x \neq s$; this reflects the fact that we only know a zero-length walk from *s* to itself.

Then, we repeatedly performs a *relaxation* operation defined as follows: if *(x,y)* is an edge such that *dist[y] > dist[x] + c(x,y)*, then we set:

- dist[y] = dist[x] + c(x,y)
- prev[y] = x

The idea of the relaxation operation is that, if we realize that we have a better walk leading to *y* by using *(x,y)* as its last edge, compared to what we know so far, we update our knowledge.

The algorithm

```
Input:
    G : directed graph with costs
    s, t : two vertices
Output:
    dist : a map that associates, to each accessible vertex, the cost of the minimum
            cost walk from s to it
    prev : a map that maps each accessible vertex to its predecessor on a path from s to it
Algorithm:
    for x in X do
        dist[x] = ∞
    end for
    dist[s] = 0
    changed = true
    while changed do
        changed = false
        for (x,y) in E do
            if dist[y] > dist[x] + c(x,y) then
                dist[y] = dist[x] + c(x, y)
                prev[y] = x
                changed = true
            end if
        end for
    end while
```

Proof of correctness

The proof is in three parts:

- at each stage, *dist* and *prev* correspond to existing walks (this comes immediately from how the relaxation operation works;
- the algorithm finishes;
- when the algorithm finishes, *dist[x] = d(s,x)* for all vertices *x*.

For the last two parts, we notice that, at iteration *k*, we have that *dist[x]* $\leq w_{k,x}$ (see the Bellman's dynamic programming algorithm). This makes the Bellman-Ford finish in at most *n-1* iterations and end with the correct distances.

Radu-Lucian LUPŞA
2016-03-15