CSE 101 – Nov 27, 2019 (Week 9)

Notes provided by Ben Sihota bsihota@ucsc.edu

SSP in weighted graph

- **Dijkstra's algorithm** (no negative edge weight)
- **Bellman-Ford algorithm**

Remarks

- Both actually find shortest (i.e. minimum weight) walks
- Minimum weight walks and minimum weight paths are one and the same, unless there exists a negative weight cycle reachable from the the source

Example

Note: $W(x, y, z, x) = -3$

Dijkstra: Requires no negative weight edge

Bellman-Ford: Returns a boolean which is true iff there is no negative weight cycle reachable from the source

Common infrastructure

Vertex attributes:

- $p[x]$ – Parent or predecessor of x
- $d[x]$ – Distance estimate of $\delta(s, x)$

Predecessor subgraph: $G_p = (V_p, E_p)$

$$V_p = \{v \in V \mid P[x] \neq n_1\} \cup \{s\}$$

$$E_p = \{(P[x], x) \mid P[x] \neq \text{nil}\}$$

Note: $|E_p| = |V_p| - 1$

Both algorithms cannot create cycles, G_p is a tree

See

PrintPath(G, s, x)

in Sec. 22.2, P. 601 to print out shortest path

Helper functions:

Initialize(G, s)

1. for all $x \in V$
   2. $d[x] = \infty$
   3. $p[x] = \text{nil}$
2. $d[s] = 0$

Relax(x, y) Pre: $y \in adj[x]$

1. if $d[y] > d[x] + w(x, y)$
   2. $d[y] = d[x] + w(x, y)$
   3. $p[y] = x$

Note:

- Relax: Changes at most the fields of y
- After Relax(x, y): We must have $d[y] \leq d[x] + w(x, y)$
- If Relax(x, y) changes a d-value, it goes down
- Both algorithms create a sequence of relaxations that causes all $d[x]$ to converge to $\delta(s, x)$

Lemma 1

Let $x \in V(G)$. Suppose that after Initialize(G, s), some sequence of calls to Relax$(., .)$ results in $d[x]$ being finite. Then G contains an $s \breve{} x$ path of weight $d[x]$

Proof Induction on length n of the relaxation sequence, i.e.

$$n = \text{ number of calls to Relax}(., .)$$

I. If $n = 0$, then the only vertex with finite d-value is the source s

So $x = s$ in this case. Indeed the trivial $s \breve{} s$ path has weight 0. The base case is therefore satisfied.

II. Let $n > 0$. Assume that for any vertex u *(instead of x)*, if $d[u]$ becomes finite after a sequence of fewer than n Relxations, then G contains an $s \breve{} u$ path of weight $d[u]$. We must show that if $d[x]$ becomes finite after n Relxations, then G contains an $s \breve{} x$ path of weight d[x]$.

So, suppose that $d[x]$ is finite after a sequence of n Relaxations. Then, some edge with terminus x was relaxed in that sequence.

$$u- \to -x$$

Let u be the origin of that edge. On call to Relax(u, x), $d[x]$ was set to

$$d[x] = d[u] + w(u, x)$$

Since we suppose that this # is finite, $d[u]$ must have been finite before call to Relax(u, x). Therefore, $d[u]$ was finite after a sequence of **fewer** than n Relaxations. By the induction hypothesis, G contains an $s\breve{}u$ path of weight $d[u]$. That path, followed by the edge (u, x) constitutes an $s\breve{}x$ path of weight $d[u] + w(u, x) = d[x]$