6.006 Lecture ~~#~~ 16: Bellman-Ford (Shortest ~~Pat~~ Paths II)

- Review of the general SSSP algorithm
- Bellman Ford + Analysis
- CLRS 24.1

General Structure of SSSP algorithms

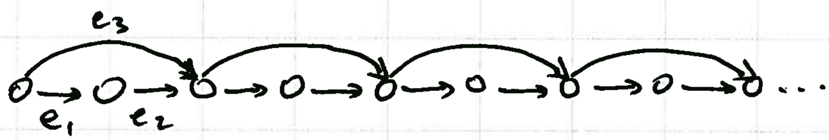
for v in V:
    d[v] = ∞   distance estimates
    π[v] = None   predecessor pointers

d[s] = 0   distance from s to s is zero

while d[v] > d[u] + w(u,v) for some v: edge weight

$$\left.\begin{aligned} & d[v] = d[u] + w(u,v) \\ & \pi[v] = u \end{aligned}\right\} \text{relax}$$

The algorithm may run for an exponential number of steps



relax e_1

relax e_2 $\qquad$ $m = |E|$

relax $\{e_4 \cdots e_m\}$ recursively to convergence

relax e_3

relax $\{e_4 \cdots e_m\}$ to convergence again

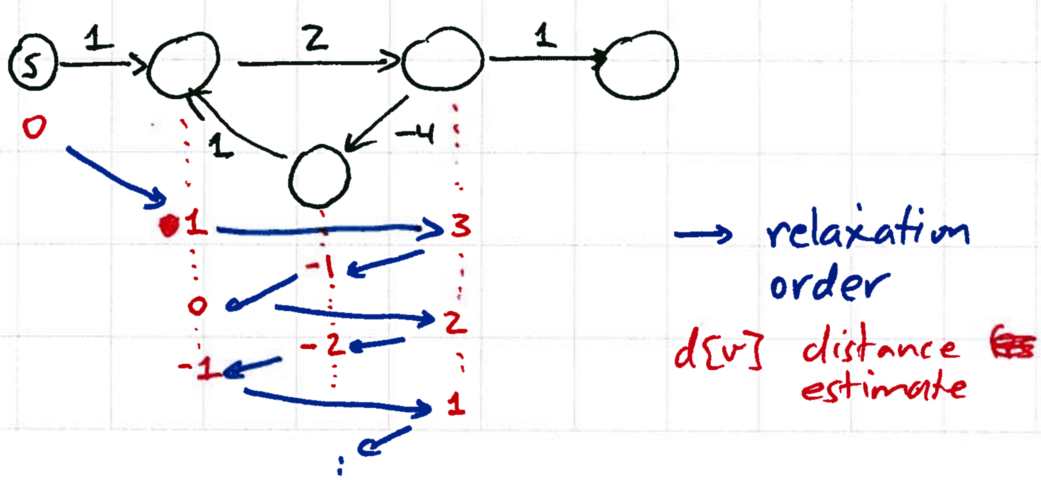
Number of relaxations, $\Theta(1)$ each:

$T(3) = 3$ $\qquad$ $n = |V|$

$$T(n) = 3 + 2T(n-2) = 3 + 6 + 4T(n-4) =$$

$$= 3 + 6 + 12 + 8T(n-6) = \Theta(2^{n/2})$$

And it may fail to terminate:

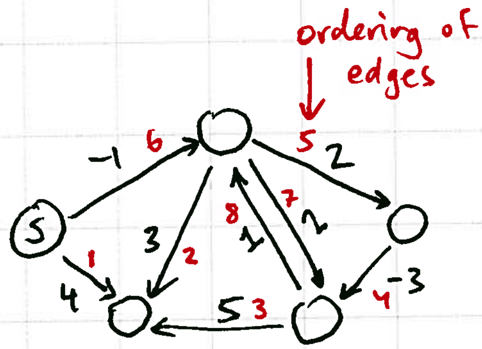


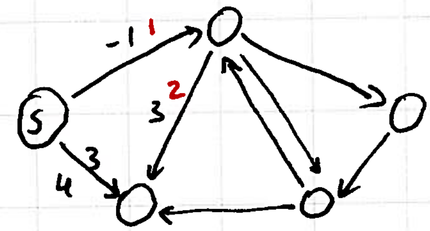
we must order relaxations for efficiency and
we must add negative-cycle detection (or disallow negative cycles)

Bellman-Ford

```
for v in V:
    d[v] = ∞
    π[v] = None
d[s] = 0
do n-1 times:
    for every edge (u,v) in E:
        if d[v] > d[u] + w(u,v):
            d[v] = d[u] + w(u,v)
            π(v) = u
for every edge (u,v) in E:
    if d[v] > d[u] + w(u,v):
        report a negative cycle and return
report that there are no negative cycles.
```

Example:





iteration					
	0	∞	∞	∞	∞
1		4	-1	1	
2		2			1
3				-2	
4	(nothing changes)				

0	∞	∞	∞	∞
	2	-1		

edge ordering influences convergence!
we did not specify an edge ordering.

Running time of Bellman-Ford:

Initialization $\Theta(V)$

Main loop: $|V|-1$ iterations over $|E|$ edges, $\Theta(1)$ operations per edge $\Rightarrow$ $\Theta(VE)$ total

Negative-cycle detection $\Theta(E)$

Total # operations is $\Theta(VE)$

Two Structural Properties

Theorem: subpaths of shortest paths are also shortest paths.

Proof: Let $p = \langle v_0, v_1, \cdots v_i, v_{i+1}, \cdots v_j, \cdots v_k \rangle$

$$p_{ij} = \langle v_i, v_{i+1}, \cdots, v_j \rangle$$

$$\begin{tikzcd} p = v_0 \arrow[r, squiggly, "p_{0i}"] & v_i \arrow[r, squiggly, "p_{ij}"] \arrow[r, bend right=60, "p'_{ij}"'] & v_j \arrow[r, squiggly, "p_{jk}"] & v_k \end{tikzcd}$$

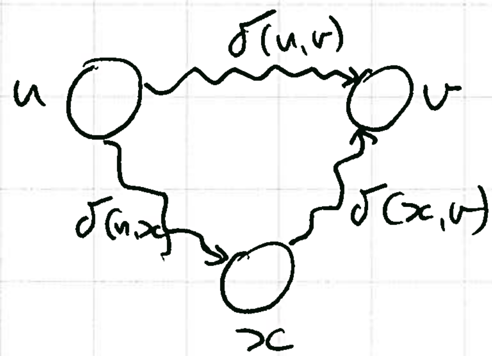
If $w(p'_{ij}) < w(p_{ij})$ then p is not shortest.

we can replace p_{ij} with p'_{ij} and get

$v_0 \overset{p'}{\rightsquigarrow} v_k$ with $w(p') < w(p)$.

Theorem: triangle inequality, for all $u, v, x \in V$ we have $\delta(u,v) \leq \delta(u,x) + \delta(x,v)$

Proof:



Analysis of Bellman-Ford

Theorem: if $G=(V,E)$ contains no negative cycles, then at the end of the "do n-1 times" loop we have $d[v]=\delta(s,v)$

length of SP from s to v.

Proof: Let $p=\langle s,v_1,v_2,\cdots v_k\rangle$ be a shortest path from s to v_k.

By the subpath theorem, $\langle s,v_1,\cdots v_{k-1}\rangle$ is also a SP, so $\delta(s,v_k)=\delta(s,v_{k-1})+w(v_{k-1},v_k)$, and similarly $\delta(s,v_i)=\delta(s,v_{i-1})+w(v_{i-1},v_i)$.

After 1 iteration of the loop, $d[v_1]=\delta(s,v_1)$

2 $\quad d[v_2]=\delta(s,v_2)$

$\vdots$

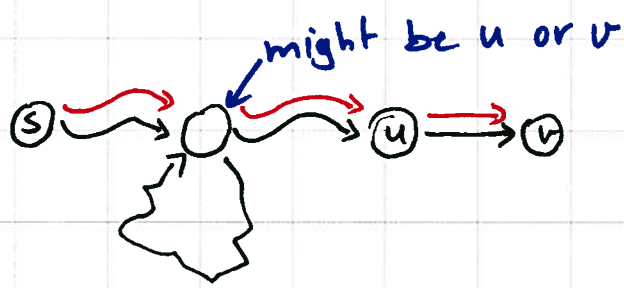
k $\quad d[v_k]=\delta(s,v_k)$

Since no negative cycles, $k\le n-1$. Done.

Theorem: Bellman-Ford correctly reports negative cycles.

Proof: After $n-1$ iterations, $d[v]$ is the length of the shortest path with $n-1$ or fewer edges from s to v.

If (u,v) can still be relaxed, there is a shorter path with n edges; it must contain a cycle.



Black path shorter than red path, so the cycle must have negative weight.

On the other hand, if there is a negative cycle, there will always be an edge that can be relaxed.