Lecture 15: Shortest Paths II: Bellman-Ford

Lecture Overview

- Review: Notation
- Generic S.P. Algorithm
- Bellman Ford Algorithm
  - Analysis
  - Correctness

Recall:

$$\begin{aligned} \text{path } p &= \ < v_0, v_1, \ \ldots, \ v_k > \\ & \quad (v_1, v_{i+1}) \ \epsilon E \quad 0 \leq i < k \\ w(p) &= \sum_{i-0}^{k-1} w(v_i, v_{i+1}) \end{aligned}$$

Shortest path weight from u to v is $\delta(u,v)$. $\delta(u,v)$ is ∞ if v is unreachable from u, undefined if there is a negative cycle on some path from u to v.

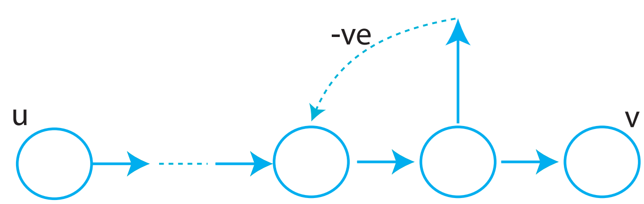

Figure 1: Negative Cycle.

Generic S.P. Algorithm

Initialize: for $v \ \epsilon \ V$: $d[v] \leftarrow \infty$
$\Pi[v] \leftarrow$ NIL
$d[S] \leftarrow 0$

Main: repeat
select edge (u,v) [somehow]

"Relax" edge (u,v)
if $d[v] > d[u] + w(u,v)$:
$d[v] \leftarrow d[u] + w(u,v)$
$\pi[v] \leftarrow u$

until you can't relax any more edges or you're tired or ...

Complexity:

Termination: Algorithm will continually relax edges when there are negative cycles present.

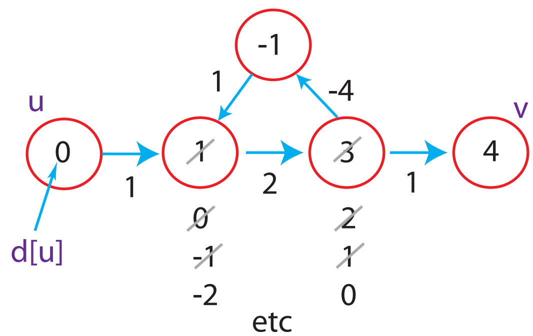

Figure 2: Algorithm may not terminate due to negative cycles.

Complexity could be exponential time with poor choice of edges.

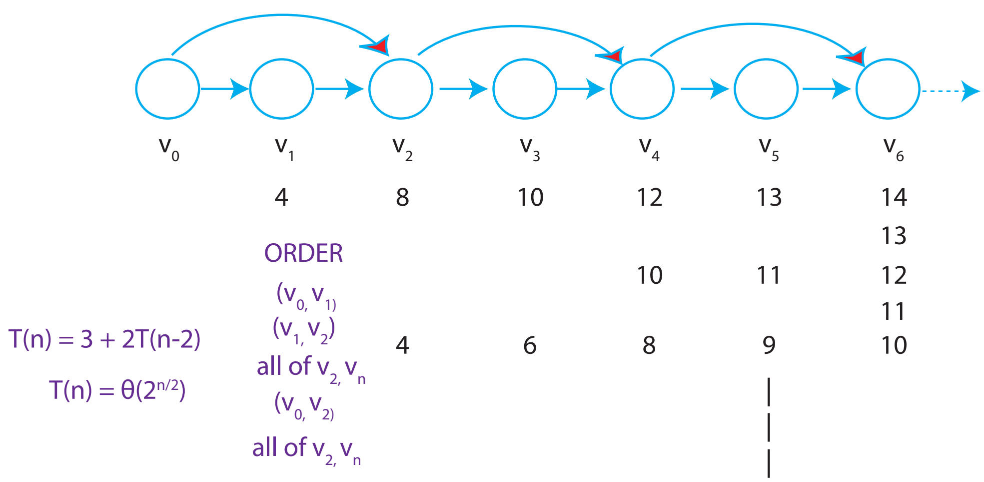

Figure 3: Algorithm could take exponential time. The outgoing edges from v_0 and v_1 have weight 4, the outgoing edges from v_2 and v_3 have weight 2, the outgoing edges from v_4 and v_5 have weight 1.

5-Minute 6.006

Here's what I want you to remember from 6.006 five years after you graduate

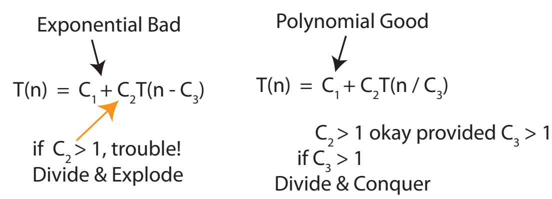

Figure 4: Exponential vs. Polynomial.

Bellman-Ford(G,W,S)

Initialize ()
for $i = 1$ to $\mid v \mid -1$
    for each edge $(u, v) \epsilon E$:
        Relax(u, v)
for each edge $(u, v) \epsilon E$
    do if $d[v] > d[u] + w(u, v)$
        then report a negative-weight cycle exists

At the end, $d[v] = \delta(s, v)$, if no negative-weight cycles

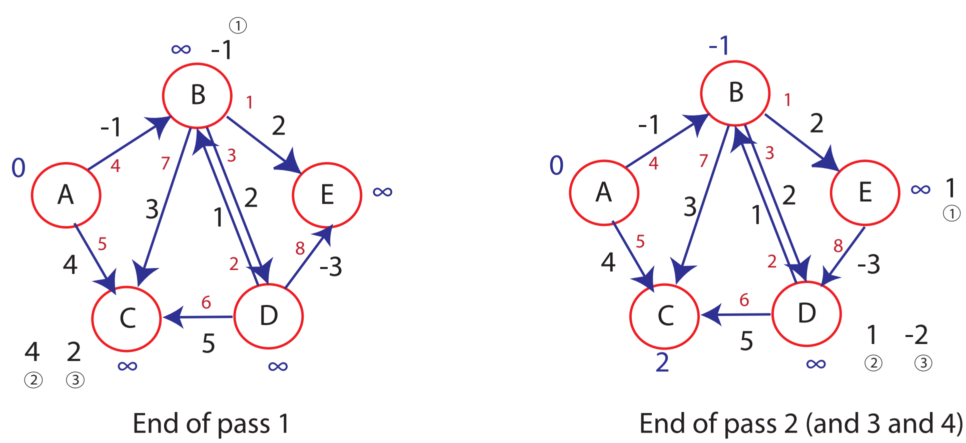

Figure 5: The numbers in circles indicate the order in which the δ values are computed. Error: Edge from D to E on left graph should be from E to D as in the right graph.

Theorem:
If $G = (V, E)$ contains no negative weight cycles, then after Bellman-Ford executes $d[v] = \delta(u, v)$ for all $v \epsilon V$.
Proof:
$v \epsilon V$ be any vertex. Consider path p from s to v that is a shortest path with minimum number of edges.

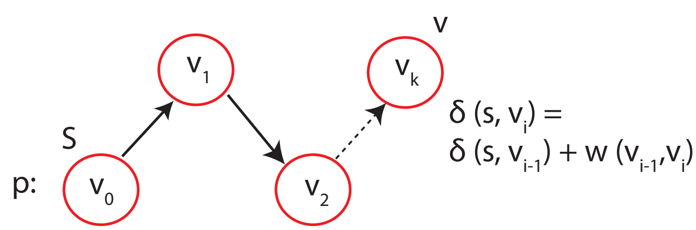

Figure 6: Illustration for proof.

Initially $d[v_0] = 0 = \delta(s, v_0)$ and is unchanged since no negative cycles.
After 1 pass through E, we have $d[v_1] = \delta(s, v_1)$
After 2 passes through E, we have $d[v_2] = \delta(s, v_2)$
After k passes through E, we have $d[v_k] = \delta(s, v_k)$
No negative weight cycles $\implies p$ is simple $\implies p$ has $\leq \mid V \mid -1$ edges

Corollary

If a value $d[v]$ fails to converge after $\mid V \mid -1$ passes, there exists a negative-weight cycle reachable from s.