**Shortest Paths: Bellman Ford**

**Review:** Notation

Generic s.p. algorithm

Bellman Ford algorithm
  - analysis
  - correctness

**Path:**

\[ p = (v_0, v_1, \ldots, v_k) \]

\[ (v_i, v_{i+1}) \in E \quad 0 \leq i < k \]

\[ w(p) = \sum_{i=0}^{k-1} w(v_i, v_{i+1}) \]

Shortest path weight from \( u \) to \( v \) is \( s(u,v) \)

\( s(u,v) \) is \( \infty \) if \( v \) is unreachable from \( u \)

\( s(u,v) \) is undefined if there is a negative cycle on some path from \( u \) to \( v \)
**General Structure of S.P. Algs**

Initialize: \( \text{for } v \in V : d[v] \leftarrow \infty \)
\( \Pi[v] \leftarrow \text{NIL} \)
\( d[s] \leftarrow 0 \)

Main: repeat:
- Select edge \((u,v)\) \( \text{[somehow]} \)
- 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.
Comlexity

Could be exponential time with poor choice of edges.

\[ T(n) = 3^2 T(n-2) \]

\[ T(n) = \Theta(2^{n/2}) \]

5-minute 6.006

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

Exponential bad, Polynomial good.

\[ T(n) = C_1 + C_2 T(n{-}c_3) \]

\[ T(n) = C_1 + C_2 T \left( \frac{n}{c_3} \right) \]

If \( c_2 > 1 \), trouble!

Divide & Conquer

If \( c_3 > 1 \)

Divide & Conquer
**Bellman-Ford** \((G, w, s)\)

Initialize()
for \(i = 1\) to \(|V| - 1\)
    for each edge \((u,v) \in E\):
        Relax \((u,v)\)
for each edge \((u,v) \in E\)
do if \(d[v] > d[u] + w(u,v)\)
    then report a negative-weight cycle exists
At the end, \(d[v] = d(s, v)\), if no negative-weight cycles

---

![Graph Diagram]

End of pass 1
End of pass 2 (and 3 and 4)

**Theorem**

If $G = (V, E)$ contains no negative weight cycles, then after Bellman-Ford executes $d[v] = S(s,v)$ for all $v \in V$.

**Proof:**

Let $v \in V$ be any vertex.

Consider path $p$ from $s$ to $v$ that is a shortest path with minimum number of edges.

Let $p: s \rightarrow v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_k = v$.

Then $S(s,v_i) = S(s,v_{i-1}) + \omega(t_{i-1}, v_i)$.
Proof (contd.):

\[ P: \hspace{1cm} S \xrightarrow{} v_0 \xrightarrow{} v_1 \xrightarrow{} v_2 \xrightarrow{} \cdots \xrightarrow{} v_k \xrightarrow{} v \]

Initially, \( d[v_0] = 0 = s(S, v_0) \) and is unchanged since no negative cycles.

After 1 pass through \( E \), we have:
\[ d[v_1] = s(S, v_1) \]
\[ d[v_2] = s(S, v_2) \]

" 2 passes "
\[ d[v_k] = s(S, v_k) \]

" k passes "

No negative-weight cycles \( \Rightarrow p \) is simple \( \Rightarrow p \) has \( |V|-1 \) edges.

---

**Corollary**

If a value \( d[v] \) fails to converge after \( |V|-1 \) passes, there exists a negative-weight cycle reachable from \( S \).