Single Source Shortest Paths Problem

Motivation

$G(V, E, W)$ V: vertices E: edges W: weights

Two algorithms:
- Dijkstra, none negative edges, $O(V\lg V + E)$
  $E \in O(V^2)$
- Bellmen – Ford: pos/neg edges, $O(VE)$
  $E \in O(V^2)$

Note: cost of above algorithms is not a func. of weight!

Problem

Path $P = \langle v_0, v_1, \cdots v_k \rangle$

$\langle v_i, v_{i+1} \rangle \in E$ for $0 \le i < k$

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

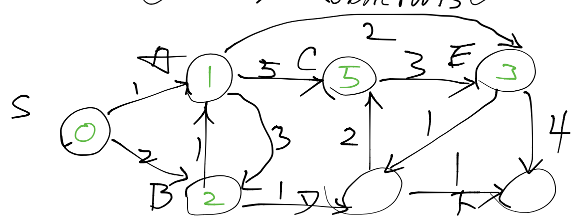
Find p with minimium weight w

Weight Graphs

- $v_0 \xrightarrow{P} v_K$ (v_0) is a path from v_0 to v_0 of weight 0
- shortest path weight from u to v as

$$\delta(u, v) = \begin{cases} \min\{ w(P) : u \xrightarrow{P} v \} & \exists \text{ any such path} \\ \infty, & \text{otherwise} \end{cases}$$

e.g.



$d(v)$: (value in the circle) current weight

$\pi[v]$: predecessor on best path to v

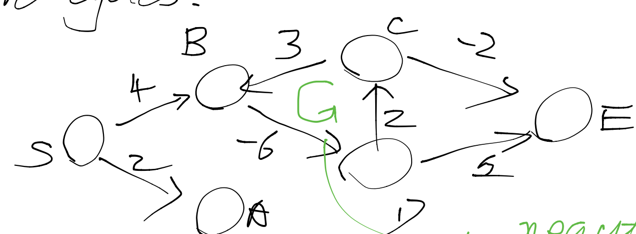
$\pi[s] = NIL$

when $d(v) == \delta(u,v) \Rightarrow$ done!

- *Negative weights*

Why?: reverse toll, social networks, ...

Negative cycles:



negative cycles make shortest weighted path problem indeterminant

This is what makes Bellman-Ford algorithm more complex

General Structure (for no neg. cycles)

- Initialize for $u \in V$, $d[v] \leftarrow \infty$ $\pi[u] \leftarrow NIL$
- Repeat | select edge (u,v)

  if $d[v] > d[u] + w(u,v)$

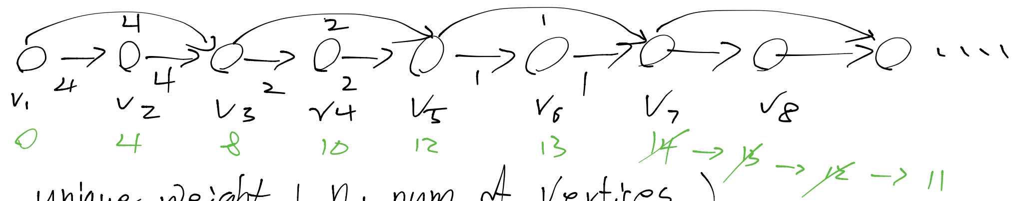
  ("Relax edge (u,v)")

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

  $\pi[v] \leftarrow u$

until all edges have $d[v] \leq d[u] + w(u,v)$
(can't relax any edges any more)

Brutal force, not necessarially fast, e.g.



$2^{n/2}$ unique weight (n: num of vertices)

May need to relax the edge $2^{n/2}$ times in order to get the shortest weight path!

Q: How do we select edges properly

□ Optimal substructure Property

- Subpath of a shortest path are also shortest paths