Lecture 14

Last time: Shortest path in graphs

unweighted: BFS

weighted:
- Dijkstra's Shp. Alg.: if all the weights are positive.
- negative edge weights: Bellman-Ford.
- if there are negative weight cycles then SHP problem is ill defined and we cannot solve it.

Single source shortest path problem.

run Bellman-Ford one more time after finding the shp: if for some node the shp. value changes then $\exists$ cycle (NWC). STOP.

All pairs shortest path problem: FORD-FULKERSON ALG.

Dijkstra's SHP.: BFS with a cost function on the links

Dijkstra$(G, \cancel{u}, w: E(G) \to \mathbb{R})$

initialization,
for all $v \in V$
  $cost(v) = \infty$
  $cost(u) = 0$

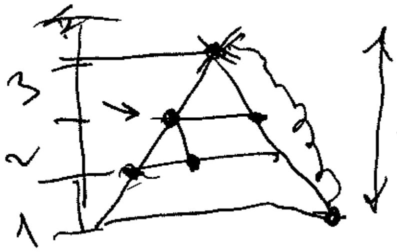
for a vertex u
  for all neighbours x of u
    If $cost(u) + w(u,x) < cost(x)$
      $cost(x) = cost(u) + w(u,x)$
      $prev(x) = u$

Priority Queue.
1. create
2. extract min
3. update cost

} operation

$\Rightarrow$ prioritize nodes based on edge weights
$\Rightarrow$ extract every vertex only once from P.Q.

Binary Heap.



$O(n \underbrace{\log n}_{h})$ cost of building a binary heap.

height	# of nodes
3	$2^0 \to 2^{3-3}$
2	$2^1 \to 2^{3-2}$
⋮	

Total cost $= \sum_{h=0}^{\lfloor \log_2 n \rfloor} h \cdot 2^{\lfloor \log_2 n \rfloor - h}$

$= \sum_{h=0}^{\lfloor \log n \rfloor} \frac{h \cdot 2^{\lfloor \log n \rfloor}}{2^h} = 2^{\lfloor \log n \rfloor} \cdot \sum_{h=0}^{\lfloor \log n \rfloor} \frac{h}{2^h}$

Recall our geometric series $|r|<1$

$$\boxed{\sum_{h=0}^{\infty} h \cdot r^h = \frac{r}{(1-r)^2}}$$

$\Downarrow$

Total Cost $= \underbrace{n}_{2^{\log n}} \sum_{h=0}^{\log n} \frac{h}{2^h} \leq n \sum_{h=0}^{\infty} h \underbrace{\left(\frac{1}{2}\right)^h}_{r = 1/2}$

$\leq \underbrace{n \cdot \frac{\frac{1}{2}}{\left(\frac{1}{2}\right)^2}}_{2n} \Rightarrow$ total cost $= O(n)$

$n = |V|$

Bellman - Ford single source SHP. Alg

<u>Given:</u> a weighted, directed graph $G=(V,E)$
and a weight function $w: E \to \mathbb{R}$.
and a source node s.

<u>Return:</u> YES or NO answer for indicating whether or not there is a negative-weight cycle reachable from source node s

we will use 2 subroutines.

Initialization

$\boxed{\begin{array}{l} \text{Init}(G,s) \\ \text{for each vertex } v \in V(G) \\ \quad \text{do } d(v) \leftarrow \infty \\ \qquad \boxed{\begin{array}{l} \text{parent}[v] \leftarrow NIL \\ \pi[v] \leftarrow nil. \end{array}} \\ d(s) \leftarrow 0 \end{array}}$

Relaxation for an <u>edge</u> $(u,v) \in E(G)$

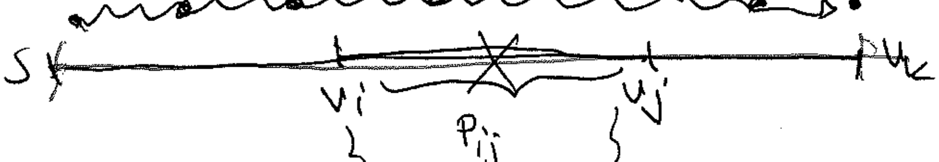
$\boxed{\begin{array}{l} \text{Relax}(u,v,w) \\ \text{If } \underbrace{d(v)}_{\text{distance from } s \text{ to } v} > \underbrace{d(u)}_{\text{distance from } s \text{ to } u} + w(u,v) \\ \quad \text{then } d(v) \leftarrow d[u] + w(u,v) \\ \qquad \pi(v) \leftarrow u \end{array}}$

$d(v) = \left.\begin{array}{l}\text{length} \\ \text{weight} \\ \text{cost}\end{array}\right\}$ distance.
of the path from $s \rightsquigarrow v$

<u>Lemma:</u> Subpaths of shortest paths are also shortest paths
(<u>optimal</u> substructure property)

$s \rightsquigarrow v_k$ path is shp.

proof sketch!



proof by contradiction

assume: P'_{ij} s.t. $\text{cost / length}(P'_{ij}) < \text{Cost}(P_{ij})$

<u>CUT and PASTE</u> operation. what happens to cost of (P_{sk})?

$\to$ Replace path P_{ij} with P'_{ij}

B-F (G, w, s)

$O(|V|)$ 1. Init (G, s)

$O(n)$ 2. for $i: 1$ to $V(G) - 1$

3. do for each edge $(u,v) \in E(G)$

$O(|E|)$ 4. do Relax(u, v, w)

$O(m \cdot n)$

5. for each $(u,v) \in E(G)$

6. do if $d(v) > d(u) + w(u,v)$

$\underbrace{O(|E|)}_{O(m)}$ 7. then Return FALSE

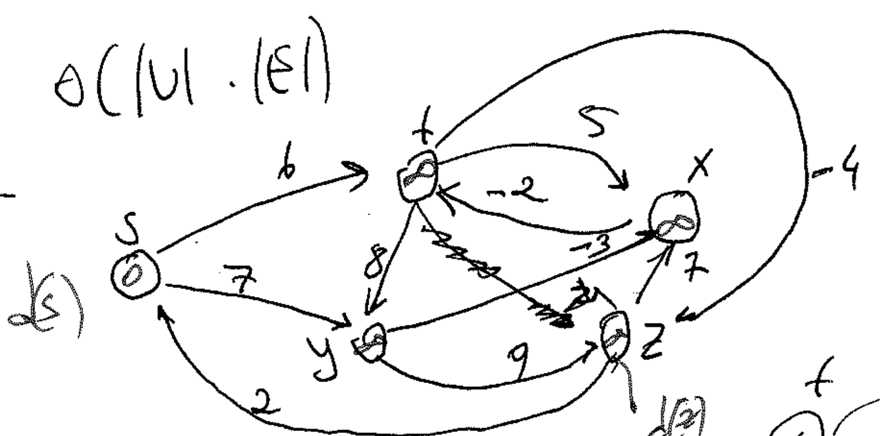
8. RETURN TRUE $\Rightarrow \exists$ negative edge cycle

$\Rightarrow O(|V| \cdot |E|)$

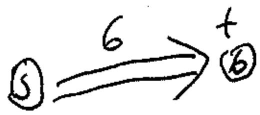
planer graph

example:

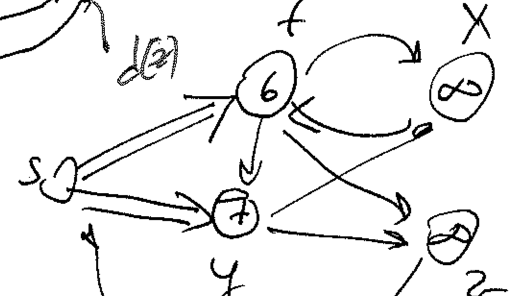
Init



1st

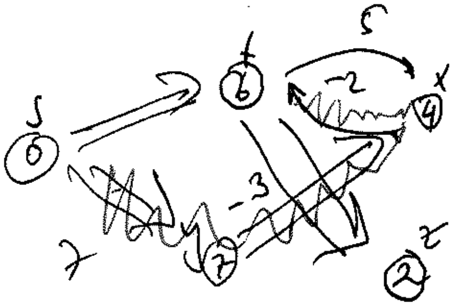


$\Rightarrow$

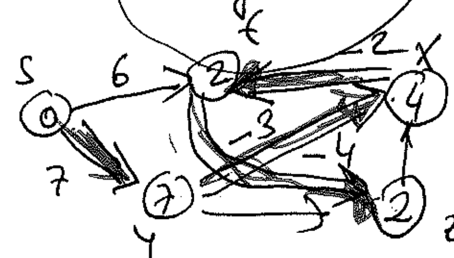


2nd

3rd



$\Rightarrow$



d[t] = 2

end of B.F

d[z] = ? −2

Run one more time $y \xrightarrow{-3} x \xrightarrow{-2} t \xrightarrow{-4} z \xrightarrow{7} x$

$\Rightarrow$ BF returns TRUE