Thursday, March 28, 2024 10:51 AM

Shortest Paths Cont'd:

Last lec: Single Source Shortest Path (SSSP)

Given a directed $G = (V, E)$, $w(e) \geq 0$, $\forall e \in E$

& $s \in V$.

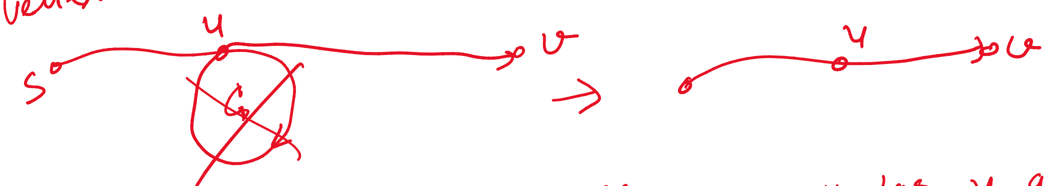
Find, $\forall v \in V$ s-v shortest path length

$\equiv$ minimum distance from s to v

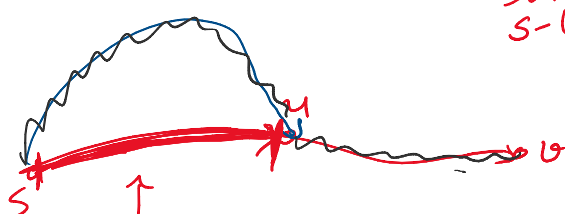
$\equiv$ mindist (s, v).

Observations:

(1) No vertex repeated



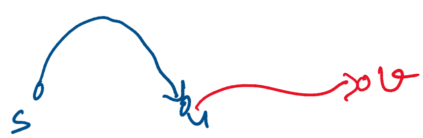
(2)



Suppose, s-v shortest path has u as an intermediate vertex.

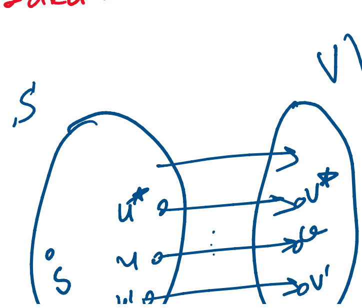
shortest s-u path

If not then we get a smaller s-v path

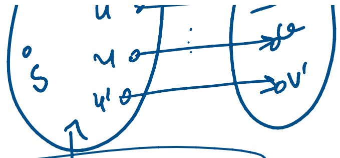


★ Dijkstra's Alg'm (SSSP):

Idea: Find next nearest vertex, starting from s itself.



$$(u^*, v^*) = \underset{(u,v) \in E}{\arg\min} \; d[u] + w(u, v)$$



$$(u^*, v^*) = \underset{\substack{(u,v)\in E\\ u\in S,\, v\notin S}}{\arg\min}\ d[u] + w(u,v)$$

$$\boxed{\begin{array}{l}\forall u \in S.\\ \text{mindist}(s,u) = d[u]\end{array}}$$

claim: $\text{mindist}(s, v^*) = d[v^*] = d[u^*] + w(u^*, v^*)$

($\le$)

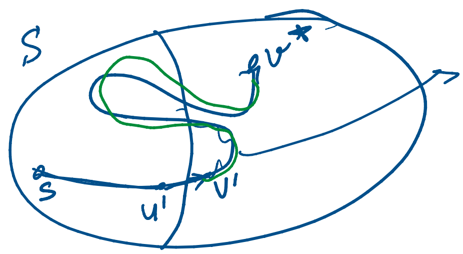
($\ge$)

pf sketch:

($\le$)

$$d[v^*] \le d[u^*] + w(u^*, v^*)$$

($\ge$)



p^*: shortest s-v^* path

$$\text{mindist}(s, v^*) = \text{length}(p^*) = d[u'] + w(u', v') + \cdots$$

$$\ge d[u'] + w(u', v') + 0$$

$$\ge d[u^*] + w(u^*, v^*)$$

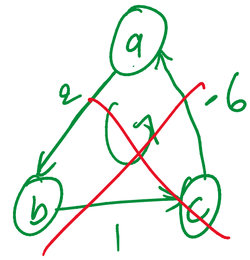
($\because$ choice of (u^*, v^*))

Implementation via Fibbonacci heap

$$\boxed{O(n \lg n + m) \text{ runtime.}}$$

★ **Negative Weight.:** $w(e) \in \mathbb{R}$, $\forall e \in E$.

e.g.



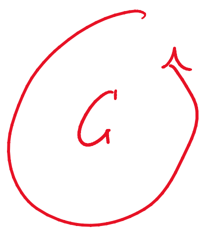
$\text{mindist}(a,b) = 2 \quad (a \to b)$

$= -1 \quad (a,b,c,a,b)$

$= -4$

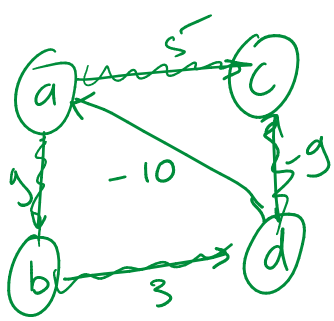
$\vdots$

Assumption: No neg. weight cycle.



$\sum_{e \in G} w(e) \geq 0$

e.g.



$s = a$

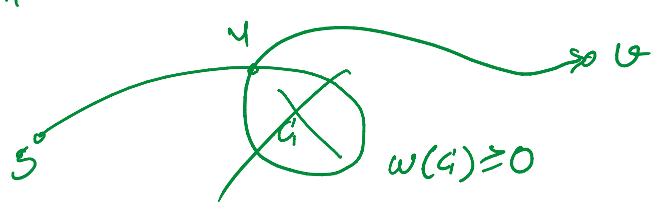
$d(c,1) = 5$

$d(c,2) = 5$

$d(c,3) = 3$

★ Bellman-Ford (1956): Dynamic Prog.

Obs: still no vertex repeats.



$w(C) \geq 0$

edge on any shortest path $\leq (n-1)$

→ Subproblem Def: $\forall u \in V,\ 0 \leq l \leq n$

from s to v

→ Subproblem Def: $\forall u \in V$, ...

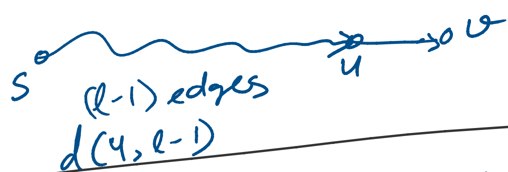
$d(v, \ell)$: min distance from s to v using "at most ℓ edges".

Ans: $d(v, n)$, $\forall v \in V$

→ Base Case: $d(s, \ell) = 0 \quad \forall \ell$

$\forall v \neq s$, $d(v, 0) = \infty$

→ Formula:



$$\boxed{d(v,\ell) = \min \begin{cases} d(v, \ell-1) \\ \displaystyle\min_{\substack{u: \\ (u,v) \in E}} w(u,v) + d(u, \ell-1) \end{cases}}$$

→ Evaluation Order: Increasing order of ℓ.

→ Running Time: For a fixed $\ell = 1, 2, \dots, n$

time / vertex v: $O(|\text{in-neighbors}(v)|)$

time for all vertices: $O\left(\sum_{v \in V} |\text{in-neigh.}(v)|\right)$

$= O(m)$

Total time: $\underset{\substack{\uparrow \\ \#\text{ diff values of } \ell}}{O(n)} \cdot O(m) = O(m \cdot n)$.

Total time = O( )

$\uparrow$

diff values of ℓ

Aside:

claim: G has no neg weight cycle

$$\iff d(v, n) = d(v, n-1) \quad \forall v \in V$$

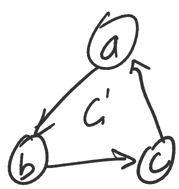
Pf sketch: ($\Rightarrow$) easy (exe)

($\Leftarrow$) $d(v,n) = d(v,n-1) \ \forall v \Rightarrow \nexists$ neg weight cycle.

$$d(v, n-1) = d(v,n) \leq d(u, n-1) + w(u,v)$$

$\because$ hypothesis

$$\Rightarrow \boxed{w(u,v) \geq d(v, n-1) - d(u, n-1)}$$



$$w(a,b) \geq d(\cancel{b, n-1}) - d(\cancel{a, n-1})$$

$$+ \ w(b,c) \geq d(\cancel{c, n-1}) - d(\cancel{b, n-1})$$

$$+ \ w(c,a) \geq d(\cancel{a, n-1}) - d(\cancel{c, n-1})$$

$$w(C') \geq 0$$

---

* All Pairs Shortest Paths:

Given directed graph $G = (V, E)$ w/ weights on edges. Find, $\forall u, v \in V$, u-v shortest path distance.

* non-neg weights: $w(e) \geq 0 \quad \forall e \in E$

Run Dijkstra starting at every vertex.

n times Dijkstra.

$$n \cdot O(n \lg n + m) = O(n^2 \lg n + mn)$$

n times Dijkstra

$$n \cdot O(n \lg n + m) = O(n^2 \lg n + mn)$$
$$\leq O(n^3)$$

★ Neg weights: $w(e) \in \mathbb{R}, \ \forall e \in E$

$V = \{1, 2, 3, \ldots, n\}$ (number the vertices)

Method 1: (BF style DP)

→ Subproblem Def: $i, j \in V$, $\ell = 0, \ldots, n$

$d(i, j, \ell)$: min distance from i to j using at most ℓ edges.

→ Ans: $\forall i, j \in V$, $d(i, j, n)$

→ Base case: $d(i, i, \ell) = 0 \quad \forall i \in V$
$d(i, j, 0) = \infty \quad \forall i \neq j \in V$

→ Formula:

$$d(i, j, \ell) = \min \begin{cases} d(i, j, \ell-1) \\ \min\limits_{\substack{k: \\ (k,j) \in E}} d(i, k, \ell-1) + w(k, j) \end{cases}$$

$k \in$ in-neigh(j)

$k \in$ in-neighbor(j)

Runtime :

$$O(\underbrace{n^3}_{\#\text{ subproblems}} \cdot n) = O(n^4)$$

→ Better Formula :

$$d(i,j,l) = \min_{\substack{k \in V \\ k \neq i,j}} d(i,k,l/2) + (k,j,l/2)$$

Suppose $n = 2^h$

then $l = 1, 2, 4, 8, 16, \ldots, 2^h$ suffices

$\#$ possible values of l = $\log n = h$.

Runtime : $O(\underbrace{n^2 \log n}_{\#\text{ sub problems}} \cdot n) = O(n^3 \cdot \log n)$

Method 2 : Floyd-Warshall (1964) : (DP).

Number the vertices $1, 2, \ldots, n$

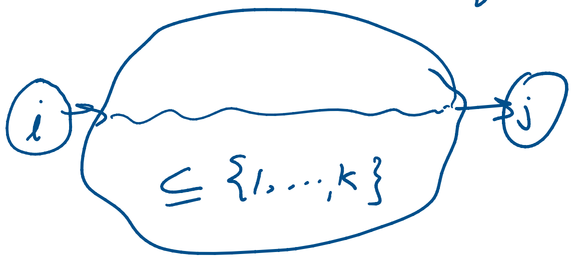
$$V = \{1, 2, 3, \ldots, n\}$$

→ Subproblem Def. : $\forall i,j \in V$; $k = 0, 1, \ldots, n$

$d(i,j,k)$: min distance from i to j

$d(i,j,k)$: min distance from i to j s.t. all intermediate vertices on this path is from $\{1,\ldots,k\}$ vertices.

($k=0$, implies empty set $\Rightarrow$ cannot use any intermediate vertex)

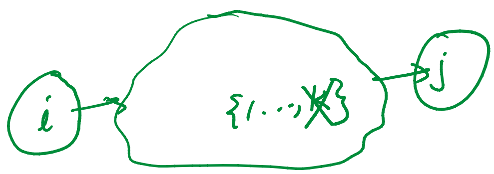


→ Ans: $d(i,j,n)$

→ Base Case: $d(i,i,k)=0 \quad \forall i \in V, \ \forall k=0,1,\ldots,n$

$$\forall i \neq j \in V, \quad d(i,j,0) = \begin{cases} w(i,j) & \text{if } (i,j) \in E \\ \infty & \text{o.w.} \end{cases}$$

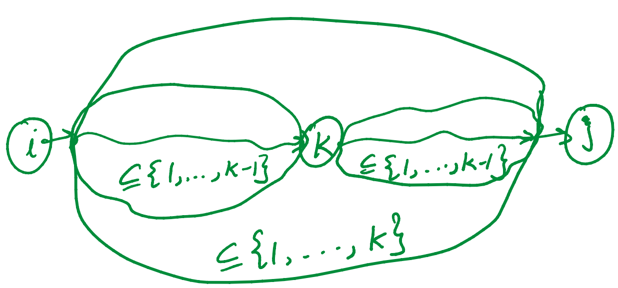
→ Formula:



do not use k   use k

$$\boxed{d(i,j,k) = \min\left\{ d(i,j,k-1),\ d(i,k,k-1) + d(k,j,k-1) \right\}}$$

use k:



k does not repeat.

→ Runtime: $O(n^3 \cdot O(1)) = O(n^3)$!

$\rightarrow$ Runtime: $O\Big(\underbrace{n^3}_{\#\text{ subprob.}} \cdot \underbrace{O(1)}_{\text{time/subprob.}}\Big) = O(n^3)$!!

$$\boxed{O(n^{2.9999})\text{ ? OPEN!}}$$

★ History:

$O(n^3)$ Floyd-Warshall '64

$O(n^3/\log^{1/3} n)$ Friedman '76

$\vdots$

$O(n^3/\log n)$ Chan '05

$\vdots$

$O(n^3/c^{\sqrt{\log n}})$ Williams '14 (Randomized)

Chan-Williams '16 (Deterministic)