Announcements

Hw1 solutions posted on Canvas

Hw2 is available on Gradescope (one coding question and 2 written question) **Due Friday Feb 6 only one late day.**

Sections **attendance mandatory, will include a 10 min quiz** about previous hw.

Prelim 1: Thursday, Feb 12. fill out this form, if you have a conflict.

Covers hw1-2, sections week 1-2, lectures through this week. Section next week is review. this week dynamic programming practice

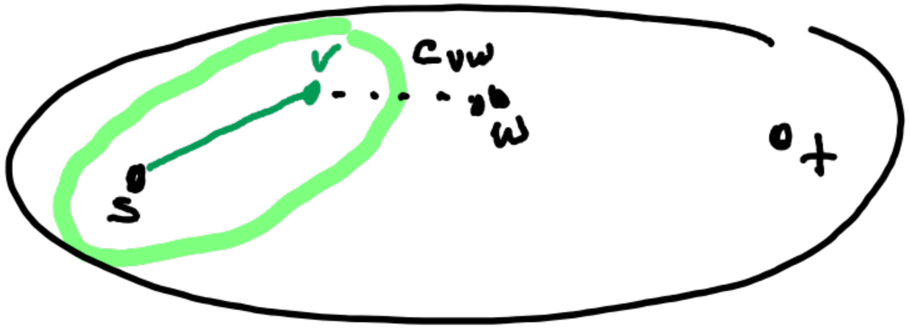
Other prelim info and practice questions will be posted shortly

Min-cost path: recall Dijkstra (sec 4.4)

costs are ≥ 0 also in CS 2110/12

Graph (V, E) cost c_{vw} for $(v,w) \in E$

Problem find min-cost $s - t$ path



similar to Prim

select min cost out of s

(s,v) = min cost path to v

Computes dist $s \to v$ all other $v \in V$

Dijstra

dist$(s)=0$ $S=\{s\}$ nodes reached

while $t \notin S$

select $w \notin S$ min

$\min_{v,w}$ dist$(v) + c_{vw}$

set dist(w) = dist$(v) + c_{vw}$

add w to S

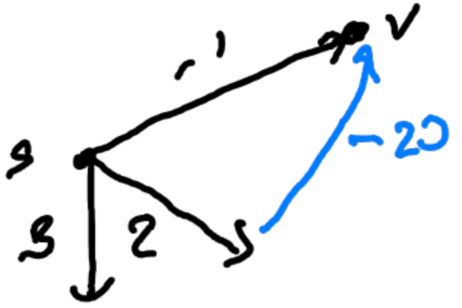
endwhile

Min cost path with negative costs (sec 6.8)
assume no negative cycles

Does Dijkstra work OK?

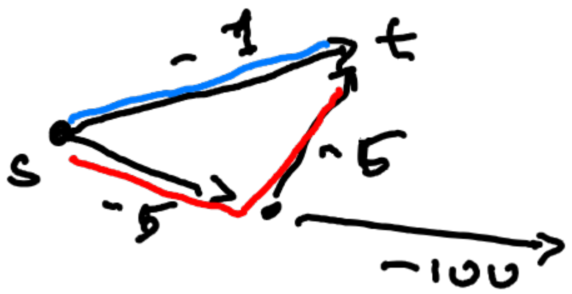
min cost s → v path
(s, v)

1



not working

2 natural idea: add M to all costs, run Dijkstra $c_{vw}+M$



min cost (s,t) cost -1

with $c_{vw}+M$ red wins

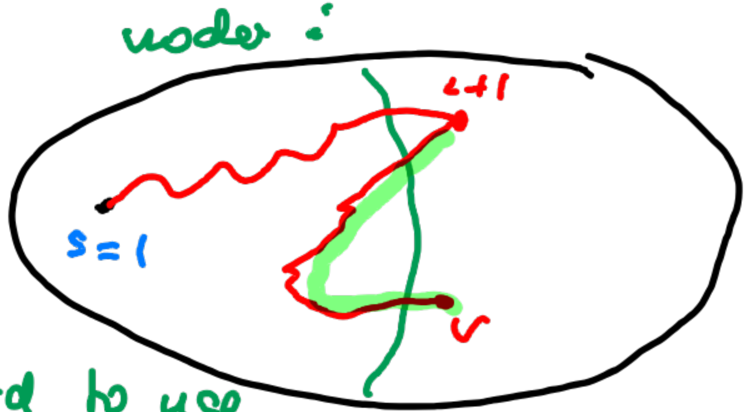
We will compute min cost s → v for all nodes v

Dynamic Programming III: Min-cost path
assume no negative cycles

promise

Subproblems?

version 1: min cost using only nodes $\{1,\dots,i\}$ in path



version 2: min cost using $\leq i$ edges

this is hard to use

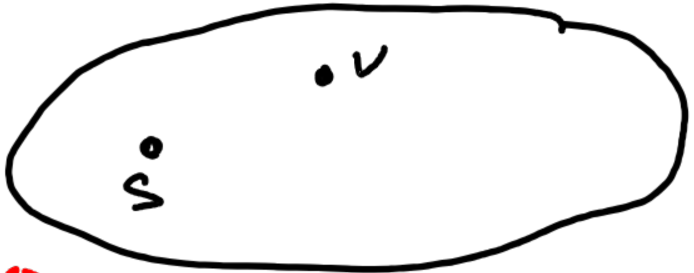
today try version 2.

$Opt(i,v)$ = min cost of $s \to v$ path using $\leq i$ edges

base $i=0$ or 1

$$Opt(1,v) = \begin{cases} c_{sv} & (s,v) \in E \ \& \ v \neq s \\ +\infty & (s,v) \notin E \ \& \ v \neq s \\ 0 & v = s \end{cases}$$

due to promise of no neg. cycles



The dynamic program

Bellman-Ford algorithm

$$Opt(0,v) = \begin{cases} +\infty & v \neq s \\ 0 & v = s \end{cases}$$

For $i = 1, 2, \ldots, |V|-1$

For all $v \in V$

$$Opt(i,v) = \min\Big(Opt(i-1,v),\ \min_{w:(w,v)\in E}\big(Opt(i-1,w) + c_{wv}\big)\Big)$$

using $< i$ edges

using i edges

end for

end for

return $Opt(|V|-1, t)$

$Opt(i,v)$ = min cost path $s \to v$ using $\leq i$ edges

$Opt(i-1,v)$ computed

last edge

How many edges can be in a path $s \to t$

$\leq |V|-1$

Join by Web **PollEv.com/evatardos772**

What is the running time of the Bellman-Ford Algorithm on a graph with n nodes and m edges?

A. $O(n + m)$

B. $O(n^2)$

C. $O(nm)$

D. $O(m^2)$

E. $O(n^3)$

F. *Slower than any of these bounds*

inner loop $\sum_v O(\deg(v)) = O(m)$

updating v $O(\deg(v))$

- $\boxed{\text{outside for loop } u}$
- inside for loop u

computing Opt

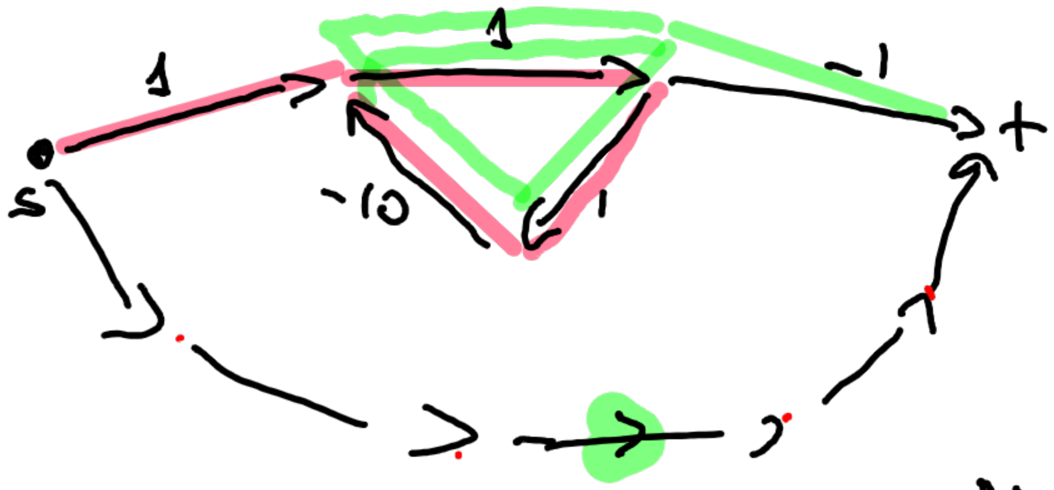
max n-1 if v has many incoming edges

$Opt(0,v) = \begin{cases} 0 & v = s \\ \infty & \text{else} \end{cases}$

- For $i = 1, \ldots, n-1$
  - For $v \in V$
    - $Opt(i,v) = \min\left(Opt(i-1,v), \min_{w:(w,v)\in E} Opt(i-1,w) + c_{wv}\right)$

What happens if G has negative cycle?

see section this week: trust but verify



Alg finds

path using ≤ 8 edges

so min cost = − ∞

Note: min cost simple path = no repeat nodes

computationally hard