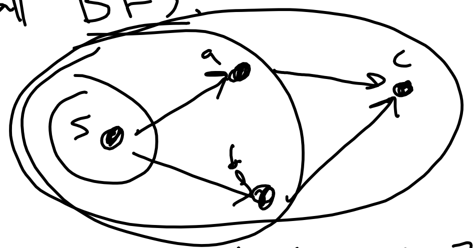
Graphs 4 (Shortest Paths)

Thursday, April 2, 2020 4:20 PM

Shortest Paths

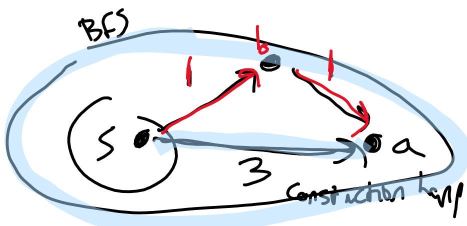
Recall BFS:



Queue: [~~s~~ ~~a~~ ~~b~~ ~~c~~ ~~c~~]

— BFS finds shortest paths (from Node s) in unweighted.

Today: weighted graphs.



$s \overset{2}{\rightsquigarrow} a$

construction happening?

Starting point:

recursive description of problem.

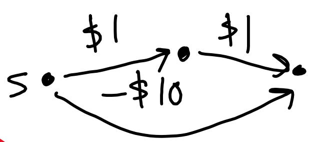
dist(u) be distance of shortest path from s to u.

$$\text{dist}(u) = \begin{cases} 0 & \text{if } u = s \\ \min\limits_{\substack{v \xrightarrow{w} u \\ v \in in(u)}} w + \underline{\text{dist}(v)} \end{cases}$$

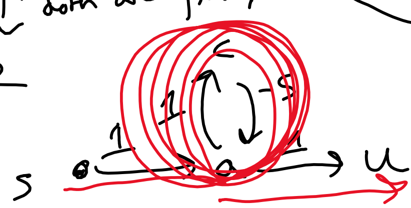
in(v): nodes with edge to v

recursive but not a recurrence (in terms of smaller problems).

— Negative weight? ↕ both are a problem!



— cycles



SP: $s \rightsquigarrow u$?

no shortest path! $-\infty$

— Special case: DAG.

Original recursive definition is a recurrence relation.

dist(u) depends on dist(v) for each $v \in in(u)$.

DAGs have an topo. ordering $\prec$.

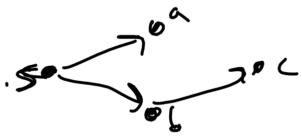
$v \to u \implies v \prec u$ Analyze: $O(|V|+|E|)$

---

s ⇝ u sp from s to u

SSSP = single source shortest path

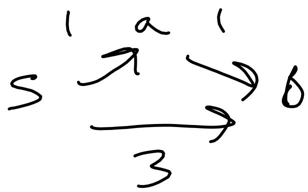
from s to every node



Later All pairs

- Bellman-Ford: "hop" counter

dist(i, u): distance of sp from s to u, taking at most i hops (∞ is unreachable in i edges)



$dist(0,s) = 0$

$dist(0,a) = \infty$

$dist(1,b) = 3$

$dist(2,b) = 2$

dist(i, v):

$$\begin{cases} \text{if } i=0: & 0 \text{ if } v=s,\ \infty \text{ otherwise} \\ \text{else}: & \min\left(\begin{array}{l} \min\limits_{u \in in(v)} w(u \to v) + dist(i-1,u) \\ dist(i-1,v) \end{array} \right) \end{cases}$$