Computer Algorithms I

Spring 2020

Shortest paths

directed, edge-weighted graphs

sometimes edge weights are assumed to be ≥ 0

weight of a path: sum of its edge weights

$\delta(s, v)$: weight of a shortest path from s to v (distance)

single-source shortest paths: find shortest paths from a source s to all other vertices!

all-pairs shortest paths: find shortest paths between any two vertices!

Relaxation for single-source shortest paths

assume that we compute a vertex attribute $v.d$ such that $\delta(s, v) \leq v.d$

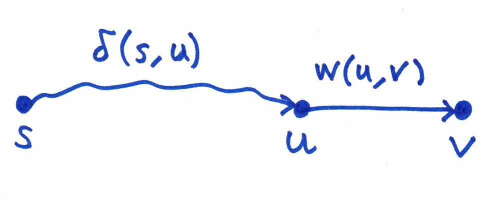
$RELAX(u, v, w)$

if $v.d > u.d + w(u, v)$ **then**

$$v.d = u.d + w(u, v)$$
$$v.\pi = u$$

Claim: after $RELAX(u, v, w)$ it still holds that $\delta(s, v) \leq v.d$

$$\delta(s, v) \leq \delta(s, u) + w(u, v) \leq u.d + w(u, v) = v.d$$

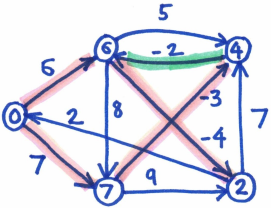


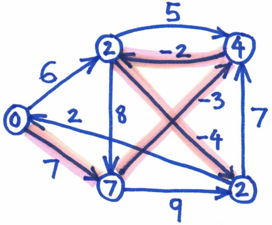
Bellman - Ford algorithm

initialization: $v.d = \infty, v.\pi = NIL$ for every vertex v, $s.d = 0$

```
BELLMAN-FORD(G, w, s)
INITIALIZE-SINGLE-SOURCE(G, s)
for i = 1 to |G.V| - 1
    for each edge (u, v) ∈ G.E
        RELAX(u, v, w)
for each edge (u, v) ∈ G.E
    if v.d > u.d + w(u, v)
        return FALSE
return TRUE
```

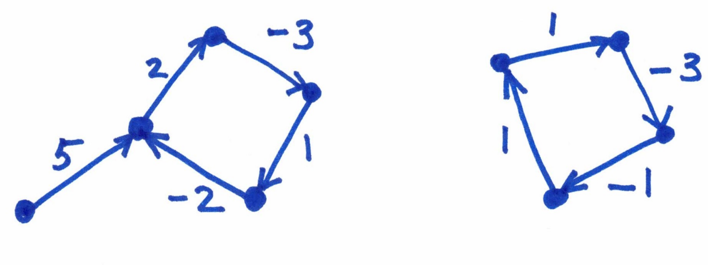
Bellman - Ford example





$6 > 4 - 2$

Negative weight cycles



shortest paths do not exist if there is a negative weight cycle reachable from s

Bellman - Ford algorithm properties

edges can have negative weights

Theorem

If there are no negative weight cycles reachable from s *then at the end* $v.d = \delta(s, v)$ *for every vertex* v*, the edges* $(v.\pi, v)$ *form a shortest-paths tree, and the algorithm returns TRUE.*
Otherwise the algorithm returns FALSE.

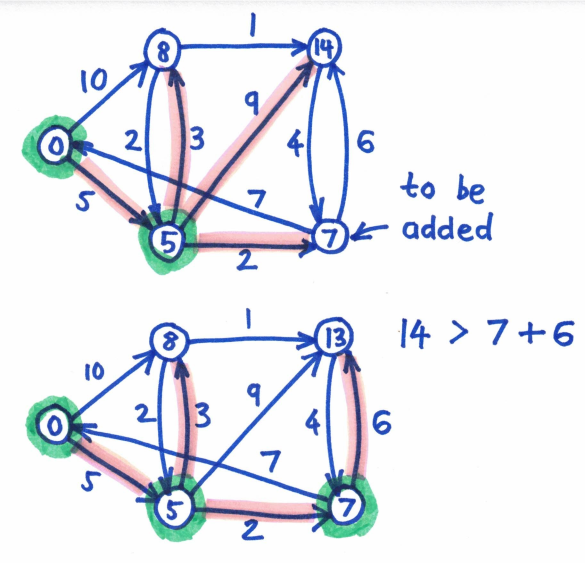
complexity is $O(|V| \cdot |E|)$

Dijkstra algorithm

```
DIJKSTRA(G, w, s)
INITIALIZE-SINGLE-SOURCE(G, s)
S = ∅
Q = G.V
while Q ≠ ∅
    u = EXTRACT-MIN(Q)
    S = S ∪ {u}
    for each vertex v ∈ G.Adj[u]
        RELAX(u, v, w)
```

compare with Prim: similarities and differences

Dijkstra example



Dijkstra algorithm properties

edge weights have to be non-negative

at the end:

- $v.d = \delta(s, v)$ for every vertex v
- the edges $(v.\pi, v)$ form a shortest-paths tree

complexity $O(|E| \log |V|)$