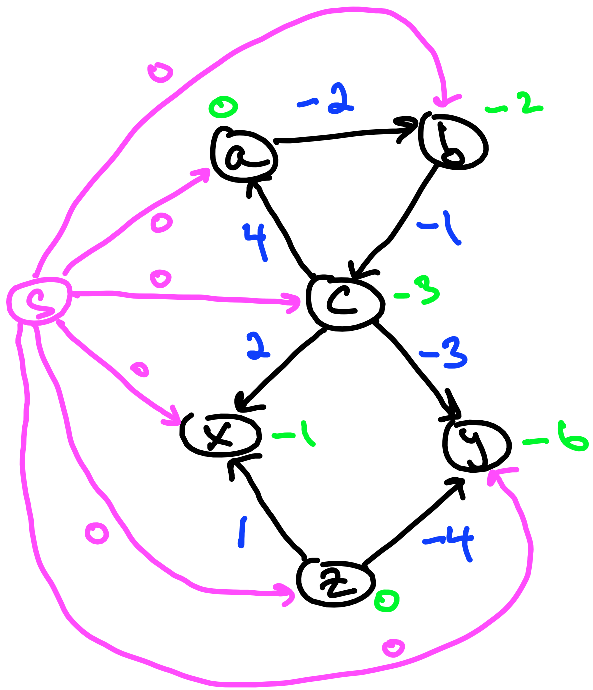
Algorithms: Design and Analysis, Part II

All-Pairs Shortest Paths (APSP)

Johnson's Algorithm

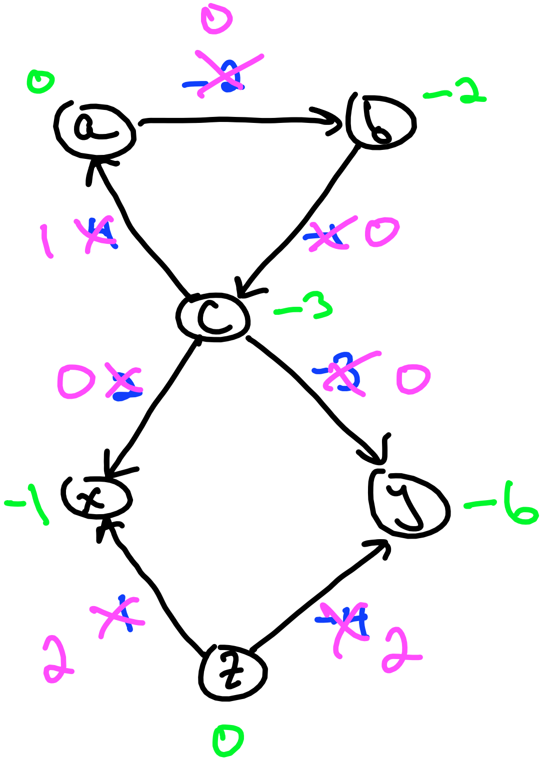
Example



Note: adding s does not add any new u-v paths for any $u,v \in G$.

Key insight: Define vertex weight $p_v :=$ length of a shortest s-v path.

Example (con'd)



Recall: For each edge $e=(u,v)$, define $c'_e = c_e + p_u - p_v$.

Note: after reweighting, all edge lengths nonnegative!

$\Rightarrow$ Can compute all (reweighted) shortest paths via n Dijkstra computations!

[no need for Bellman-Ford]

Johnson's Algorithm

Input: directed graph $G=(V,E)$, general edge lengths c_e.

① Form G' by adding a new vertex s and a new edge (s,v) with length 0 for each $v \in G$.

② Run Bellman-Ford on G' with source vertex s.

[If B-F detects a negative-cost cycle in G' (which must lie in G), halt + report this.]

③ For each $v \in G$, define p_v = length of a shortest $s \to v$ path in G'.

For each edge $e=(u,v) \in G$, define $c'_e = c_e + p_u - p_v$.

④ For each vertex u of G:

Run Dijkstra's algorithm in G, with edge lengths $\{c'_e\}$, with source vertex u, to compute the shortest-path distance $d'(u,v)$ for each $v \in G$.

⑤ For each pair $u,v \in G$, return the shortest-path distance

$$d(u,v) := d'(u,v) - p_u + p_v.$$

Analysis of Johnson's Algorithm

Running time: $\underbrace{O(n)}_{\text{step (1), form } G'} + \underbrace{O(mn)}_{\text{step (2), run BF}} + \underbrace{O(m)}_{\text{step (3), form } c'} + \underbrace{O(nm\log n)}_{\text{step (4), } n \cdot \text{Dijkstra}} + \underbrace{O(n^2)}_{\text{step (5), } O(1) \text{ work per } u\text{-}v \text{ pair}}$

$= O(mn \log n)$. [much better than Floyd-Warshall for sparse graphs!]

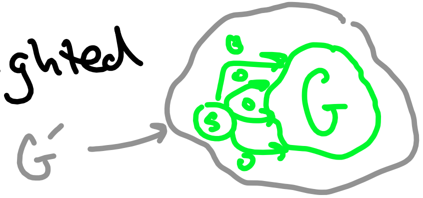
see next slide for proof

Correctness: assuming $c'_e \geq 0$ for all edges e, correctness follows from last video's quiz.

[reweighting doesn't change the shortest u-v path, it just adds $(p_u - p_v)$ to its length]

Correctness of Johnson's Algorithm

Claim: for every edge $e=(u,v)$ of G, the reweighted length $c'_e = c_e + p_u - p_v$ is nonnegative.

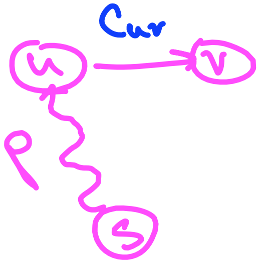


Proof: Fix an edge (u,v). By construction,

p_u = length of a shortest s-u path in G'

p_v = length of a shortest s-v path in G'

exists, by construction of G'



Let P = a shortest s-u path in G' (with length p_u).

$\Rightarrow$ $P + (u,v)$ = an s-v path with length $p_u + c_{uv}$

$\Rightarrow$ Shortest s-v path only shorter, so $p_v \le p_u + c_{uv}$

$\Rightarrow$ $c'_{uv} = c_{uv} + p_u - p_v \ge 0$.

QED!