8 November 2021

SSSP = single source shortest path

→ spec. for problem

<u>Given</u> a weighted graph (V, E, w) and a source vertex $s \in V$,

<u>Find</u> the distance (= length of the SP) to all other vertices.

<u>observation</u>: small adaptation to algo allows us to return the paths as well!

→ the sol'n stores two variables for each $v \in V$

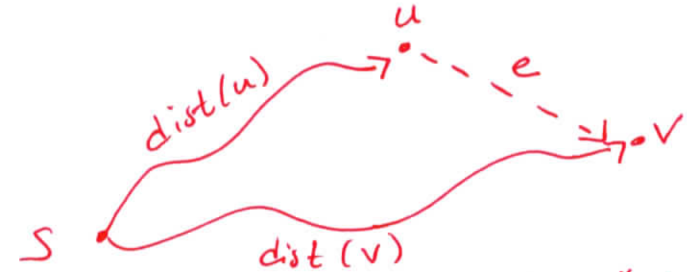
① dist(v) = dist to s (via best known path so far)

② pred(v) = parent in the SSSP tree (via best known path so far)

$G = (V, E, w)$

$|V| =: n$

$|E| =: m$



e is "~~[illegible]~~ "tense" if $\text{dist}(u) + w(e) < \text{dist}(v)$

The generic SSSP algo

```
FORDSSSP(s):
  INITSSSP(s)
  while there is at least one tense edge
      RELAX any tense edge
```

how to pick which edge next?

What is the runtime?

1. Unweighted graph, BFS — $\Theta(n+m)$
2. DAG, use DFS — $\Theta(n+m)$
3. Dijkstra, "best first" — $\Theta(m \log n)$
4. Bellman-Ford, relax them all! — $\Theta(nm)$

What is the LI of SSSP? First, recall LI's:

Q = post condition (what the loop should do)

P = pre condition (what is true going in the 1st time / "base case")

G = the loop guard

~G = not (the loop guard)

L = the loop inv.

Must prove ① Initialization "the base case"

$$P \Rightarrow L$$

② Maint

$L \wedge G \Rightarrow$ after going through the loop, L is true again

$$L_i \wedge G \Rightarrow L_{i+1}$$

③ End

$$L \wedge \sim G \Rightarrow Q$$

LI of SSSP

$\forall v \in V$,

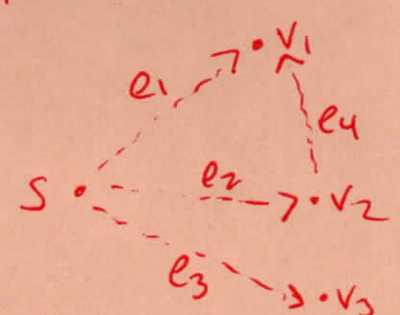
I = dist(v) is the distance from s to v in $G=(V,E)$

P = {dist(s) = 0, pred(s) = NULL} and $\forall v \in V$ st $v \neq s$: dist(v) = ∞, pred(v) = NULL

G = there exists a tense edge.

$L_{①}$ = {dist(v) is the distance from s to v in G', where $G' = (V, E')$ with E' all edges that were relaxed so far}

$L_{②}$ = {dist(v) is the distance from s to v in G'', where $G'' = (V, E'')$ with $E'' = E \setminus$ {tense edges}}



$L_{③}$ = {dist(v) is either ∞ or the length of a path from s to v}

③ $L \wedge \sim G \Rightarrow Q$

⇕

$L_{②} \wedge (\nexists \text{ tense edge}) \Rightarrow$ dist(v) is the distance from s to v

By $L_{②}$, $\mathcal{G}'' = (V,E) \setminus \{\text{tense edges}\}$.
Since $\nexists$ tense edge, we know that $\mathcal{G}'' = (V,E)$. By $L_{②}$, dist(v) stores the distance from s to v in $\mathcal{G}'' = (V,E)$, as was to be shown $\square$

$L_3 \wedge \sim G \Rightarrow Q$

Must show
① If $\sim G$ and dist(v) $= \infty$, then s and v are in different conn. components. (Proof by contra.)
② Suppose dist(v) $< \infty$. Then, $\sim G \Rightarrow$ dist(v) is the length of <u>the</u> shortest path from s to v.
(Proof by contradiction)

All Pairs Shortest Paths (APSP)

Given: (V, E, ω)

Want: ~~the distance~~

$\forall\, v, w \in V$, the distance from v to w $=$ length of the SP

~~Obvious APSP~~

~~dist [ ][ ] ← 2d array~~

~~for s ∈ V~~

~~dist [s][·] ← SSSP (s)~~

$$\boxed{\begin{array}{l}\text{Obvious APSP } (V, E, \omega)\\ \quad \text{dist [ ][ ]} \leftarrow \text{2d array}\\ \quad \text{for } s \in V\\ \quad \quad \text{dist [s][ ]} \leftarrow \text{SSSP}((V, E, \omega), s)\\ \quad \text{end for}\\ \quad \text{return dist}\end{array}}$$

Runtimes: $n \times$ runtime of SSSP.