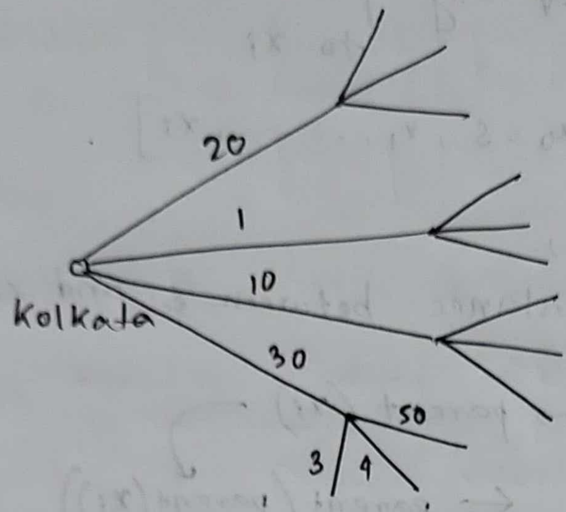


Weighted graphs but +ve weights single source shortest path.

⇒ Dijkstra's Algorithm (G, s, l)

Input: $G = (V, E)$

positive edge lengths → $l(u,v)$ $\forall (u,v) \in E$

source vertex: $s \in V$, $|V| = n+1$

Output: For all vertices u reachable from s,

dist[u] = distance of u from S in G.
(shortest distance).

Data Structure: Priority Queue implemented using heaps.

Algo for all $v \in V$:
  dist [V] = ∞
  parent [V] = nil
dist [S] = 0
R = { } # known region

while $R \neq V$:
  pick the vertex $v \notin R$, with the smallest dist().
  Add v to R
  for all edges $(v, z) \in E$:

if dist(z) > dist(v) + $l[v,z]$;

dist[z] = dist[v] + $l[v,z]$

parent[z] = v.

$\boxed{\text{Termination}}$:→

The while loop runs for atmost $n+1$ iterations.

why? Each time in the while loop, one vertex $v \in V \setminus R$ is added to R and is stopped when $R = V$.

Lemma [Monotonicity Lemma]

Let the order in which vertices are added to R be $v_0, v_1, \ldots, v_n$ where $v_0 = s$, then

$$\text{dist}[v_0] \leq \text{dist}[v_1] \leq \ldots \leq \text{dist}[v_n].$$

where dist[v] is the distance value at the end of the algorithm.

Proof by contradiction :→ Let v_{i+1} be the first vertex for which this is violated,

$$\text{dist}[v_{i+1}] < \text{dist}[v_i]$$

By our defn, v_i was added to R first.

when v_i was added; $\text{dist}[v_i] \leq \text{dist}[v_{i+1}]$ —(1)

Case-1 Say $(v_i, v_{i+1}) \notin E$

That is there doesn't exist an edge between v_i, v_{i+1}

So, from v_i, at next iteration dist[v_{i+1}] is not updated.

∴ $\text{dist}[v_i] \leq \text{dist}[v_{i+1}]$ (will hold).

[From (1)]

$\boxed{\text{Case-2}}$ Let $(v_i, v_{i+1}) \in E$,

$$dist[v_{i+1}] = dist[v_i] + l[v_{i+1}, v_i]$$

$$\Rightarrow dist[v_{i+1}] \geq dist[v_i] \quad [\because \geq 0]$$

- $dist[u]$ is the distance computed when u is added to R.

"Using next Lemma this Lemma can be proved".

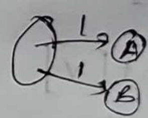
$\boxed{\text{Lemma}}$:→ Once a vertex enters R, its distance label and parent is unchanged.

Fix some vertex $x \in R$

u

Let u be the first vertex added to R after 'x' which can potentially change $dist[x]$.

This means that $(u, x) \in E$.



If $dist_i[x]$ is changed then,

$$dist_i[x] > dist_j[u] + \underbrace{l[u, x]}_{\geq 0}$$

This is contradiction.

- x was added before u to R
- When x was added;

$$dist[x] \leq dist[u]$$

$\boxed{\text{Corollary}}$:→ $x \neq s$

"Parent at the end of algorithm"

[Not parent at each step].

$$dist[x] = dist[parent[x]] + l[parent[x], x]$$

Lemma-3 (Shortest Path Lemma) :→

Let $P = (x_0 = s, x_1, \ldots, x_k)$ be any path,

Let p_i be the subpath $(x_0, \ldots, x_i)$

$$dist[x_i] \leq l[P_i]$$

Proof by contradiction :-

Lets take a path p and x_i be the first vertex in P, for which this is violated;

$$dist[x_i] > l[P_i] \quad -(1) \quad ; \quad dist[x_{i-1}] \leq l[P_{i-1}]$$

$$dist[x_{i-1}] \leq l[P_{i-1}] = l[P_i] - l[x_{i-1}, x_i] \leq l[P_i] < dist[x_i]$$

[from (1)]

$\Rightarrow$ $dist[x_{i-1}] < dist[x_i]$

$\Rightarrow$ x_{i-1} went into R first

$$dist[x_i] = \min \{ dist[x_i], \underbrace{dist[x_{i-1}] + l[x_{i-1}, x_i]} \}$$

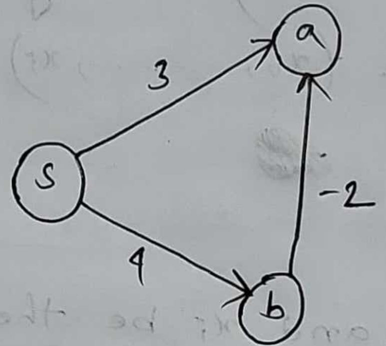
$$\leq l[P_{i-1}] + l[x_{i-1}, x_i]$$

$$= l[P_i]$$

$\therefore$ $dist[x_i] \leq l[P_i]$

$\therefore$ Contradiction.

Bellman Ford ⇒



update(z)

$$dist(z) = min\{dist(z), dist[u] + l[u,z]\}$$

(i) dist(z) is shortest ~~path~~ to s, if u is the last node in the shortest path to z and dist[u] is correctly set.

(ii) More updates don't harm you.

Bellman - ford (G, l, s)

for all $v \in V$

$dist(v) = \infty$

$parent(v) = nil$

$dist(s) = 0$

repeat $|V| - 1$ times :

for all $(u,v) \in E$:

$$dist[v] = min\{dist[v], dist[u] + l[u,v]\}$$

for -ve weight cycles

for all $(u,v) \in E$:

if $dist[v] > dist[u] + l[u,v]$

return "Negative cycle".

Proof of correctness

(i) Bellman-Ford detects negative cycles reachable from s.

(ii) If there are no negative cycles, dist at the end of the algo is the true shortest distance.

$dist_i[u] \rightarrow$ distance estimated for u from s at the end of i^{th} iteration

Proof for (i) $\exists$ neg cycle, $\exists v$ s.t.

$$dist_n[v] > dist_n[u] + l[u,v]$$

for some $(u,v) \in E$

Neg cycle C

$$v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_k \rightarrow v_0 \Rightarrow \sum_{i=0}^{k-1} l[v_i, v_{i+1}] < 0$$

By contradiction Assume that $\forall v$

$$dist_n[v] \leq dist_n[u] + l[u,v]$$

$$\Rightarrow \underbrace{\sum_{i=1}^{k} dist_n[v]}_{\text{Both are same}} \leq \underbrace{\sum_{i=1}^{k} dist_n[v_{i-1}]}_{\text{all } v_{i-1} \text{ are connected to } v_i} + \underbrace{\sum_{i=1}^{k} l[v_{i-1}, v_i]}_{\rightarrow \geq 0}$$

$$0 \leq \sum_{i=1}^{k} l[v_{i-1}, v_i]$$

(Contradiction).

Proof for (ii)

$dist_k[v]$ = Shortest dist. from s to v using $\leq k$ edges

atmost $n-1$ edges

shortest path

v_0 v_n

No, neg cycles

Base-case $k = 0$

$dist_0[s] = 0$ $dist_0[v] = \infty$

$\because$ All other vertices are not connected at $k=0$. Trivially True.

Induction hypothesis

$dist[k-1] = l[Q]$

at the k^{th} iteration.

$$dist_k[V] = \min \left\{ \underset{\geq l[P]}{dist_{k-1}[V]}, \underbrace{dist_{k-1}[U] + l[U,V]}_{l(P)} \right\}$$

$dist_k[V] \leq l[P]$

Also study

DAG,

Topological Sort