[Some edits from what we discussed in class. Noted in purple]

Lecture 10

Feb 9, 2022

Join the course page on SAFE if you intend to appear in the exams.

Single source shortest paths on graphs with negative edge weights

Input: Directed graph, source s, weighted.
- negative edge weights allowed
- negative cycles also allowed

Output: Len of shortest path from s to v for all vertices $v \in V$

OR

Declare that there is a negative weight cycle in the graph

Bellman-Ford's Algorithm

(Based on dynamic programming)

Q1. What is the optimal substructure?

Q2. What are the subproblems?

$(w, v) \in \underline{E}$

$$P = \underset{(s \to w)}{\tilde{P}} + (w, v)$$

Claim: P is the shortest (s,v) path $\Rightarrow$ $\check{P}$ must be a shortest (s,w) path.

If green (s,w) path is shorter than $\check{P}$ then green path + (w,v) is a shorter (s,v) path than P.

Could happen:

$$\mathrm{Len}(\check{P}) > \mathrm{Len}(P)$$

Len(Path) = sum of weights of all edges in the path.

$\check{P}$ is simpler than P in the sense that

$$\boxed{\#\text{edges in } \check{P} \leq \#\text{edges in } P - 1} \checkmark$$

Subproblems

$\forall v \in V$, $L_{i,v}$:= len of the shortest $s \rightsquigarrow v$ path that uses at most i edges, and is allowed to contain cycles.

$$\left.\begin{array}{l} i \in \{0, 1, \ldots, n-1, \ldots\} \\ v \in V \end{array}\right\}$$

Lemma: If P is the shortest $s \to v$ path with $\leq i$ edges. Then, one of the following is true:

1) $\exists w \in V$ s.t. $P \equiv \underbrace{s \rightsquigarrow w}_{\tilde{P}} \to v$

$\tilde{P} \equiv$ shortest $s \rightsquigarrow w$ path and has $\leq i-1$ edges

✓ 2) P has at most $(i-1)$ edges.

Recurrence for $L_{i,v}$ Base cases — Fill in.

$$L_{i,v} = \min \begin{cases} L_{i-1,v} \\ L_{i-1,w} + \ell_{wv} \end{cases}$$ ✓

① How large does i go till we stop?

② What are the guarantees when we stop.

→ How do we detect negative weight cycles?

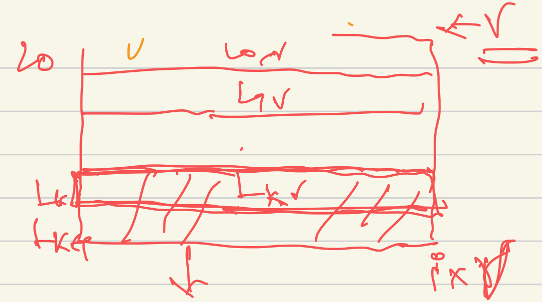
Lemma 2

If for some $k \geq 0$,

$\forall v \in V, \quad L_{k+1,v} = L_{k,v}$

Then,

1. $L_{r,v} = L_{k,v} \quad \forall v \in V, \ \forall r \geq k$
2. If G has no negative weight cycle, then for every $v \in V$, $L_{k,v}$ is the correct length of shortest path from $s \to v$ in the graph G.



Notice the change from what we discussed in class: No negative wt cycle condition is only needed for ②.

Pf: Part 1:

If $L_{k,v} = L_{k+1,v}$ $\forall v$, then the input to the recurrence for $\{L_{k+2,v} : v \in V\}$ is the same as the input to the recurrence for $\{L_{k+1,v} : v \in V\}$.

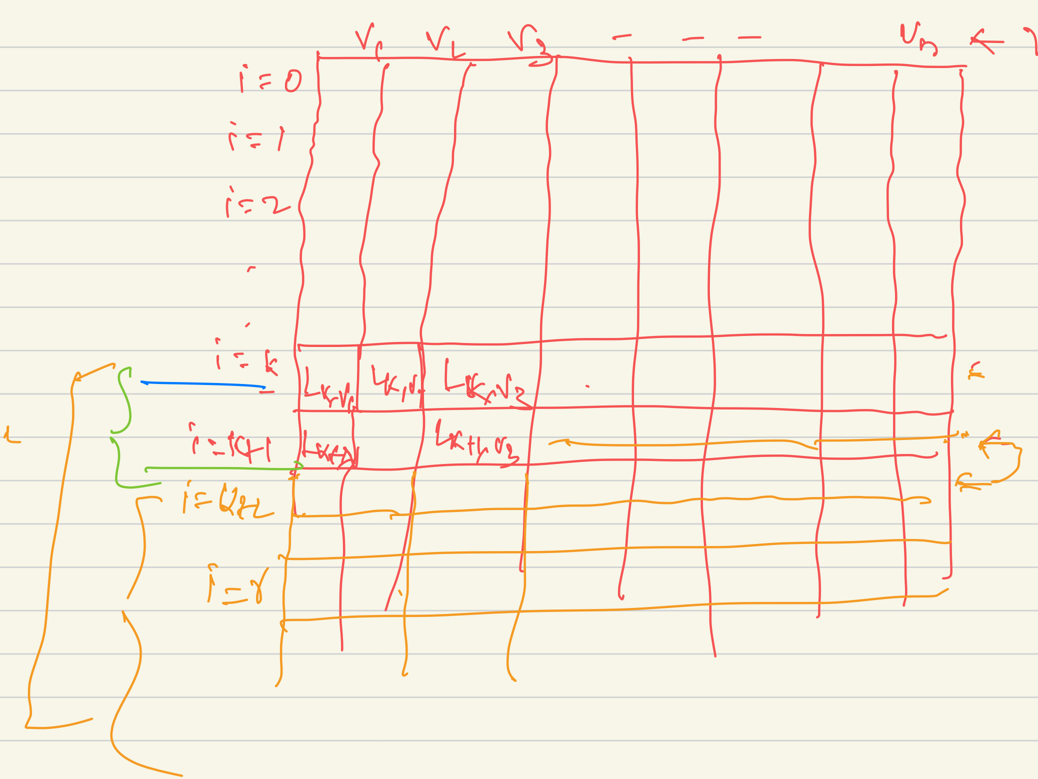
So, the outputs remain the same. + Induction.

$L_{i,v}$

$i \in \mathbb{N},\ v \in V$

This part is true for all graphs — possibly even those that have negative wt cycles

— All rows are equal.



Part 2:

[Let r be the number of edges in the shortest s-r path in the graph. Δ_v denote the length of the shortest path.

① $r \le k$

$$\Delta_v \le L_{k,v} \le L_{r,v} \le \Delta_v$$

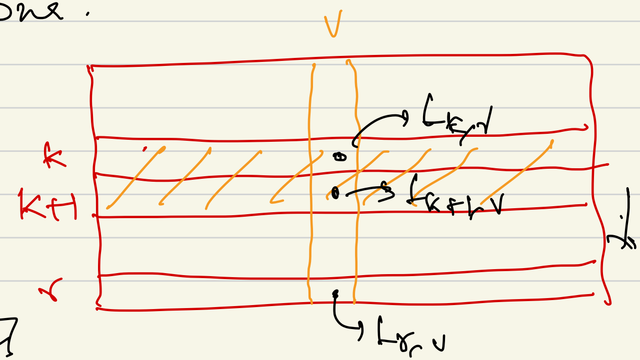
$$L_{k,v} = L_{r,v} = \Delta_v$$]

uses the fact that there is no negative wt cycle

② $r > k$.

a) $L_{r,v} = L_{k,v} = \Delta_v \rightarrow$ done.

b) $\Delta_v = L_{r,v} < L_{k,v}$



contradicts the Part A of the lemma. $\boxtimes$

Lemma 3

G : directed graph, no negative weight cycles
n : vertices.

then, for every $v \in V$ reachable from s,
there is $\boxed{a}$ shortest $s \to v$ path with at most $(n-1)$ edges.

Pf:

$> n-1$ edges in an n vertex gph
$\Rightarrow \exists$ cycles in the path.

$\Rightarrow$ There is a min wt path with fewer edges.

+ the length does not increase.

- Repeat till no cycles (# edges $\leq n-1$)

Two immediate corollaries of the lemmas:

Corollary 1:

If graph G does not have a negative weight cycle then

$$\forall v \in V, \quad L_{n-1,v} = L_{n,v}$$

Pf: From Lemma 3 it follows that $L_{n-1,v}$ is equal to the length of shortest path from $s \to v$ in G.

This + no neg cycle $\Rightarrow L_{n-1,v} = L_{n,v}$. (why?)

$\boxtimes$

Corollary 2

If G has a negative weight cycle, then $\exists\, v \in V$ s.t

$$L_{n-1,v} \neq L_{n,v}.$$

Pf: Suppose not, then $\forall v \; L_{n-1,v} = L_{n,v}$.

By part ① of Lemma 2, this implies $L_{k,v} = L_{n-1,v}$

$\forall\, v \in V$ and all $k \geq n$.

But then, if there is a negative weight cycle, then we can always find a vertex s.t $L_{k,v}$ can be made arbitrarily small by making k large enough and going around the negative weight cycle multiple times.

So, $L_{k,v}$ cannot stabilize $\forall v$ for any k.

$\boxtimes$

Pseudocode for Bellman-Ford

(slightly edited for clarity)

Base case:

L : $n \times n$ matrix

$L(0, s) := 0$

$\forall\, v \neq s$,

$L(0, v) := +\infty$

Recurrence:

for $i := 1$ to n

for $v \in V$

$$L(i, v) := \min \begin{cases} L(i-1, v) \\ \min\limits_{\substack{w \in V \\ \text{s.t. } (w,v) \in E}} L(i-1, w) + \ell_{w,v} \end{cases}$$

$\boxed{\text{Checking negative wt cycle condition}}$

If $\forall v$, $L_{n-1,v} \neq L_{n,v}$

Declare that the graph has a negative weight cycle.

Else. Return L.