# Shortest Paths : Lecture #4

APSP : all pairs shortest paths
SSSP : single source shortest paths : } usually directed graphs.

may be negative edge weights

often assume weights are reals on which we can do add / subtract but not much else

sometimes assume they are small ~~numbers~~ integers

---

## Algo 1: Dijkstra's Algorithm   SSSP, directed, non-negative.

Maintain "estimates" of s.p. distance from s. initially $d(s)=0$

$d(x) = \begin{cases} \ell_{sx} & \text{if } x \in N(s) \\ \infty & \text{o/w.} \end{cases}$

then pick unfixed vertex $u$ with smallest estimate, fix it,

and relax all its out-edges : $\forall v$ st $u \to v$

$$d(v) = \min\{d(v), d(u) + \ell_{uv}\}.$$

$O(n^2)$ dumb. $O(m \log n)$ binary heaps, $O(m + n \log n)$ Fibonacci Heaps.

[HW: sps edge lengths integers are at most $C$. simple algo to get $m + n\sqrt{C}$ algo.]

---

Negative edge lengths (but no negative cycles, of course)

## Bellman Ford (Moore)

for $i = 1$ to $n-1$
  for each edge $(xy)$
    relax $(xy)$   $d(y) = \min\{d(y), d(x) + \ell_{xy}\}$
    (parent$_y$ updated accordingly)

Thm: computes SSSP correctly if no neg. cycle. in $O(mn)$ time

Fact: do one more round and edge lengths change $\iff$ negative cycle.

Better: [Goldberg] if edge weights $\leq C$ then (even with negcycles)
$O(\sqrt{n}\, m \lg C)$ time

---

~~BTW~~ APSP: non-neg weights : $n$ Dijkstras $O(mn + n^2 \lg n)$.
neg wts : $n$ BF $O(mn^2)$ ← bad.

Here's a simple fix: [Johnson's algorithm].

Suppose $\exists$ a vertex $s \in V$ st can reach everyone from $s$. let $D_x$ = shortest path distance of $s \to x$ (using say Bellman-Ford). ← no neg cycles, of course.

~~Fact~~ Def: Reduced cost of an edge $xy$

$$\Rightarrow \ell'_{xy} := D_x + \ell_{xy} - D_y.$$

($\geq 0$ by fact that shortest paths sat $D_y \leq D_x + \ell_{xy}$ $\forall xy \in E$)

Simple fact: $\forall$ $a \to b$ paths $P$

$$\text{length}_{\ell'}(P) = \text{length}_{\ell}(P) + D_a - D_b.$$

$\Rightarrow$ shortest $ab$ paths the same for both length functions

$\Rightarrow$ Can compute shortest paths in this non-neg wts graph.

$\Rightarrow$ APSP in time $1 \times$ BF $+ (n-1)$ Dijkstras $= O(mn + n^2 \lg n)$.
[Pettie $O(mn + n^2 \lg\lg n)$]

---

In fact could have used any function $\Phi_x$ $\forall x \in V$ st

$$\Phi_x + \ell_{xy} - \Phi_y \geq 0.$$

← this is called a "feasible potential" ~~or a "valid labeling scheme"~~.

Eg ~~BTW~~: if $\ell \geq 0$ then $\Phi = 0$ is feasible potential. Adding a constant to all pts does not change feasibility.

- moreover suppose we set $\Phi_s = 0$ for some node $s \in V$ then $\Phi_x$ is ~~an upper~~ a lower bound on distance from $s \to x$.

$\Rightarrow$ feasible potentials are underestimates of $s \to x$ distance.

so suppose we say $\max \sum_x \Phi_x$

st $\Phi_s = 0$.

$\Phi_y \leq \Phi_x + d_{xy} \quad \forall xy \in E$

this is an LP.

and it is the dual of the standard shortest path LP. (Exercise).

---

APSP in earnest now: init $d_{ij} = l_{ij}$ if $ij \in E$, $\infty$ otherwise

Floyd-Warshall: $\begin{cases} \forall k = 1..n \\ \quad \forall i = 1..n \\ \quad\quad \forall j = 1..n \\ \quad\quad\quad d(ij) = \min\{d_{ij}, d_{ik} + d_{kj}\}. \end{cases}$

$O(n^3)$ time.

---

Could use "matrix mult". Define the min-sum product

$$(A \circ B)_{ij} = \min_k \{A_{ik} + B_{kj}\}$$

$$A_{ij} = \begin{cases} 0 & \text{if } i=j \\ l_{ij} & \text{if } (ij) \in E \\ \infty & \text{otherwise} \end{cases}$$

then $A \circ A$ = shortest path using 2 hops.

$$A^n = A \circ A \circ A \cdots \circ A$$

this matrix product is associative so can repeatedly square; get $A^n$ in $\lg n$ mults.

$\Rightarrow$ $APSP(n) \leq O(\log n \times \mathrm{MinSumProd}(n))$.

---

Q1) How fast can we do MSP?

Q2) Can we get $APSP(n) = \Theta(\mathrm{MinSumProd}(n))$ $\longleftarrow$ Yes! [Maybe] [HW]

$\to$ Big open question:

the problem is that $(\mathbb{R}, \min, +)$ is not a ring so cannot use fast mat-mult.

it is a semiring $\longleftarrow$ does not have inverses. under min.

so currently, $MSP \overset{?}{=} O(n^{3-\epsilon})$ for $\epsilon > 0$ is open.

But next lecture we'll see some progress.

A different tack: what if edges have small lengths. Can we do something? ~~Yes~~

~~[Yes for undirected graphs]~~

~~Simple cases~~ Undirected $[0..M]$ : $Mn^{\omega}$ [Zwick, Alon-Galil-Margalit, Seidel]

Directed: $M^{0.68} n^{2.58}$ [Zwick, AGM, Takaoka, etc].

---

Today: Seidel's Algorithm for undirected, unweighted graphs in time

$O(n^{\omega} \cdot \log n)$.

[Recall: we don't know how to do min sum product even in this case] in a generic way.

[use undirected crucially!]

---

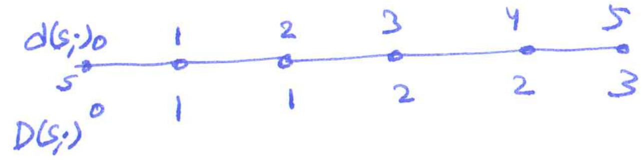
· Suppose $A$ is the adjacency matrix $A_{ij}=1 \Leftrightarrow ij$ is an edge.

$B$ is the $\{1,2\}$-hop matrix : $B_{ij}=1$ if $A_{ij}=1$ or $(A_{ik}, A_{kj})$ is a path.

Note: $B = A +_2 A^2$ ← Boolean matrix mult.

let $D$ be ~~dist~~ distances in $B$.

$d$ ~~A~~ be distances in $A$.




Fact 1: $D(x,y) = \left\lceil \frac{d_{xy}}{2} \right\rceil$.

Pf: $d_{xy} \le 2 D_{xy}$ ⇒ $D_{xy} \ge \left\lceil \frac{d_{xy}}{2} \right\rceil$. And $\exists$ a path of length $\left\lceil \frac{d_{xy}}{2} \right\rceil$ in $B$.

↑ every edge in $B$ contributes $\le 2$ in $A$

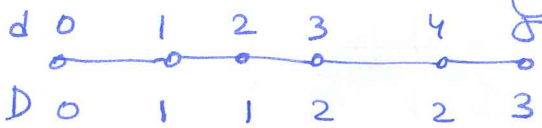
∎

Good: so having computed $D$ recursively, need to figure out

whether $d_{xy} = 2D_{xy}$ or $2D_{xy} - 1$..

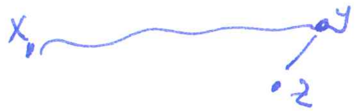
How to do this fast?

Fact 2: Sps $d_{xy} = 2D_{xy}$




then $D_{xz} \geq D_{xy}$ $\forall z$ neighbor of y in A.

Pf: sps not: then $D_{xz} < D_{xy}$




$$d_{xz} \leq 2D_{xz} \leq 2D_{xy} \leq 2D_{xy}-2$$

$$d_{xy} \leq d_{xz}+1 \leq 2D_{xy}-1.$$

contradicts $d_{xy} = 2D_{xy}$. ∎

$\Rightarrow$ for all nbrs, $D_{xz} \geq D_{xy}$. $\Rightarrow$ $\sum'_{z \text{ nbr } y} D_{xz} \geq D_{xy} \cdot \deg(y)$.

---

Fact 3: if $d_{xy} = 2D_{xy} - 1$

then $D_{xz} \leq D_{xy}$ $\forall$ nbrs of y in A

and $\exists z$ st $D_{xz} \lneq D_{xy}$.

Pf: $d_{xz} \leq 2D_{xy} - 1 + 1 = 2D_{xy} \Rightarrow D_{xz} = \lceil \frac{2D_{xy}}{2} \rceil = D_{xy}$.

also look at the shortest path x, y and z be predecessor.

then $d_{xz} = 2D_{xy} - 2 \Rightarrow D_{xz} = \lceil \frac{2D_{xy}-2}{2} \rceil = D_{xy} - 1$.

---

$\Rightarrow$ for x, y:

$$d_{xy} = \begin{cases} 2D_{xy} & \text{if } \sum'_{z \in N(y)} D_{xz} \geq D_{xy} \cdot \deg(y) \\ 2D_{xy} - 1 & \text{if } \sum'_{z \in N(y)} D_{xz} < D_{xy} \cdot \deg(y). \end{cases}$$

$(D \cdot A)_{xy}$

shortest path matrix

can compute ~~up front~~ in $O(n^2)$ time.

$\Rightarrow$ Algo: Seidel( $A'$ ) $\leftarrow$ A+I)

{

if $A' = J$ then return $(J-I)$.

else

$B = A^2$ $\leftarrow$ Boolean Matrix Mult. $(\{0,1\}, OR, AND)$

$D \leftarrow$ Seidel$(B)$.

$d_{ij} \leftarrow 2D_{ij} - \mathbb{1}\left((DA)_{ij} < D_{ij} \cdot dg_A(j)\right)$

$\boxed{\text{regular matrix mult}}$

return$(d)$

}.

---

Runtime: $O(n^{\omega} \cdot \lg n)$.

Q: Can we get the paths? related : doesn't it possibly take $n^3$ time to write the paths down?

Want a "successor" matrix $S$: st $S_{ij} = k$ if $k$ is next on the $ij$ shortest path.

Can represent in $O(n^2 \lg n)$ bits. So no information theory lower bound.

How to compute successor matrix?

Boolean Product Matrix witness Problem: given $A, A'$ boolean matrices

out: $W_{ij} = \begin{cases} k & \text{if } A_{ik}, A'_{kj} = 1 \\ 0 & \text{ow.} \end{cases}$

Q1: How to solve this problem? [HW2] in $O(n^{\omega} \text{ polylog} n)$ time whp.

Q2: And if we can solve ~~BMW~~ BPMW?

Thm: Can solve successor problem using constant # of BPMW calls. $+ O(n^2)$ time

Proof: next page.

Fix distance ~~D~~ $d^*$. (for now). Compute $d_{ij} \leftarrow$ distances in $G$

Sps $ij$ are distance $d^*$. Want to find $k$ st.

(1) $A_{ik} = 1$ and

(2) $d_{kj} = d^* - 1$.

So define $A'_{kj} = \begin{cases} 1 & \text{if } d_{kj} = d^* - 1. \\ 0 & \text{otherwise} \end{cases}$ build $A'$ in $n^2$ time from $d$.

and find MPWitness for $A, A'$.

will give successors for all pairs at distance $d^*$.

OK: Naively need to do $n^{-1}$ of these calls, one for each $d^* \in [1 \cdots n-1]$

But! Consider things modulo 3. Since unweighted, ~~distances are~~ ~~if not~~.

so they don't interfere with each other.

$A_{ik} = 1$ iff $ik \in E$

$A^{(c)}_{kj} = 1$ iff $(d_{kj}+1) \bmod 3 = c$ for $c \in \{0, 1, 2\}$.

find the MPWitnesses. Gives the answer! ☒

[i.e. for $ij$, look up the witness in matrix $W^{(d_{ij} \bmod 3)}$.].