? ⟶ ||

written

CS473 Algorithms: Lecture 17 (2022-03-24)

logistics: - pset 6 due F17
- pset 7 out F17

last lecture: randomized algo - complexity of randomized algo - expectation [primarily]
- distribution [whole thing]
- tails [more realistic]

- tail bounds - Markov
- quickselect restart [restart if takes too long]
- CLT
- tail bounds - Chernoff
- polling
- balls-in-bins

today: randomized algo [recap]

Q: when does randomness help algorithm design?

A: almost always [$\Pr \overset{?}{=} 1$]
- breaking symmetry - contention resolution
- divide and conquer - [randomly] divide ↳ quicksort
- dynamic programming - [all pairs] shortest paths (unweighted) [didn't discuss] [won't see]
- data structures - hashing - algorithms that use data structures - closest pair [most algos] - Chernoff
- approximation counting
- flows - bipartite matching [won't see] - global min cut [today]

def. $G=(V,E)$ undirected. A cut is a partition $V = A \sqcup B$, $A, B \neq \emptyset$

The capacity is $|C| = |\{e : e=(u,v)\ u \in A, v \in B\}|$ [unweighted]

The global min-cut problem is to compute $\underset{C}{\arg\min} |C|$

Q: global min cut vs (s,t)-min-cut? [directedness] [directed vs undirected] [global vs (s,t)-cut]

prop. $G=(V,E)$ undirected. $G'=(V,E')$ associated directed graph w/ $E' = \{e, -e : e \in E\}$ [reversal] and capacities $c_e = 1$.

G: u — v G': $u \rightleftarrows v$

Then for any cut C with $V = A \sqcup B$, $s \in A$, $t \in B$, $|C| = |C|$ (undirected global cut; directed (s,t)-cut)

pf. $|C| = \sum_{e: u \to v,\ u \in A,\ v \in B} \underbrace{c_e}_{=1}$ [oriented $A \to B$] directed

$= |\{e=(u,v) : u \in A, v \in B\}|$

$= \quad e = \{u,v\}$ [in G]

$= |C|$ undirected

Prop: global min cut in deterministic $O(n^2 m)$ time

pf: algo: construct G' from G
$s = v_1$ [arbitrary choice]
for $t = v_2, \dots, v_n$
compute min (s,t)-cut in G'
return min of ↗

min st cut = st min cut

complexity - construction of G' is $O(m)$ time

n max-flow calls, each $O(nm)$ time [Orlin]

correctness - clm: output is always valid cut in G [but need to show is min]

clm: $C = A \sqcup B$ global min-cut in G. either

$v_1 \in A$, $v_j \in B$ some $j \neq i$ => C is (v_1, v_j)-cut => C is found [indeed, C also min s,t-cut]

$v_1 \in B$, $v_j \in A$, some $j \neq i$

↳ $C = A \sqcup B$ is (v_j, v_1)-cut

=> $C = B \sqcup A$ is (v_1, v_j)-cut [[illegible]] [partition has no direction] [& equivalently ... still holds]

=> C is found

Q: global min cut vs (s,t)-min cut [global seems harder] [& undirected seems easier] [& undirected]

thm: global min-cut in $O(n^2 \lg^2 n)$ time with high probability [we'll see]

rmk: in dense graphs $m = \Theta(n^2)$

=> $O(n^2 \lg^2 n) = O(m \lg^2 n)$ ← nearly linear time [optimal]

[today] thm: global min-cut in $O(n^4 \lg n)$ time, whp [& faster than flow, barring recent breakthroughs]

rmk: $n^4 \lg n$ worse than $n^2 m$ [flow-based]

↳ even for dense $m = \Theta(n^2)$

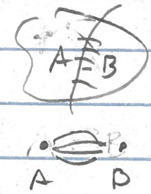
algorithm is simple

→ yields better thm w/ further optimizations

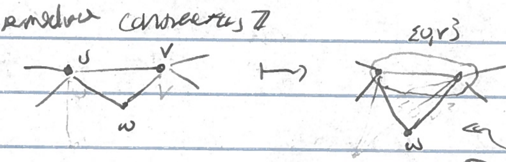
thm: global min cut in $O(n^4 \lg n)$ time whp

idea: contract a random edge

G:



[intermediate connectedness]



[set] direction

parallel edges [multigraph]

algo: global $G = (V, E)$

random contract(H)

H multigraph - vertices $S_1, \dots, S_k$ — $S_i \subseteq V$, $V = S_1 \sqcup \dots \sqcup S_k$ [partition]

- edges between S_i, S_j are $\{e = (u,v) \mid u \in S_i, v \in S_j\}$

- no self loops [not part of a cut]

if $k = 2$, return $S_1 \sqcup S_2$

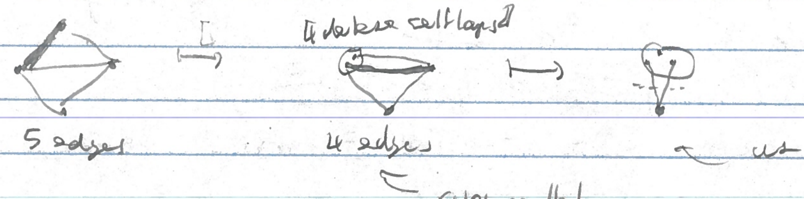
randomly choose edge e in H between some S_i, S_j $i \neq j$ — achieve all cut by contraction

merge S_i, S_j to $S_i \cup S_j$

delete self loops in $S_i \cup S_j$

return random-contract(H) — iterate

eg



fact: random-contract can be implemented in $O(n^2)$ steps [[clearly can be done in poly time]] [[not the randomized part]]

sketch: data structures

Q: success probability?

prop: random-contract returns a global min-cut w.p. $\geq 1/\binom{n}{2}$

rmk: there are $\underbrace{2^n}_{\text{all cuts}} - \underbrace{2}_{\emptyset \cup V,\ V \cup \emptyset}$ possible cuts $\Rightarrow$ also for all min cuts — mod the swap

Q: how can a min cut $\boxed{\text{not}}$ be output?

prop: cut C as $A \sqcup B$ is output iff no edge from A to B is selected for contraction

pf: $\neg \Leftarrow \neg$: if edge $e = (u, v)$, $u \in A$, $v \in B$ is contracted,

it is deleted from hypergraph

the final cut is all edges between S_1, S_2

$\Rightarrow$ if some $A \leftrightarrow B$ edge deleted then $S_1, S_2 \neq A, B$ — better?

$\Leftarrow$: by induction

initially $A = \{v : v \in A\} = T_1 \sqcup \dots \sqcup T_\ell$ [[partition into sets of merged vertices]]

$B = \{v : v \in B\} = T_1' \sqcup \dots \sqcup T'_{\ell'}$

now contract along e — e between T_i, T_j [[within A]]

— e between T_i', T_j' [[within B]]

$\Rightarrow$ $A \sqcup B$ still valid cut in contracted multigraph

[[induction]] $\Rightarrow$ at end of algo $A \sqcup B$ is valid cut in $V = S_1 \sqcup S_2 \Rightarrow S_1, S_2 = A, B$ ☒

Q: $\Pr[$no edge of C is contracted$]$?

prop: $\Pr[$no edge of C contracted, in $\boxed{\text{first}}$ round$] = 1 - \frac{|C|}{|E|}$ [[select randomly edge]]

Q: $|C|/|E| = ?$

prop: multigraph $H = (V, E)$ w/ global min cut size r, has $\geq rn/2$ edges

pf: any $v \in V$ $\{v\} \sqcup (V \setminus \{v\})$ is cut $\Rightarrow$ value $\geq r$

$\Rightarrow \deg(v) \geq r$

$\Rightarrow \quad 2|E| = \sum_v \deg(v) \ge rn$ [double count]

$\therefore \quad |E| \ge rn/2$

cor: $\Pr[\text{no edge of } C \text{ contracted in } \boxed{\text{first}} \text{ round}] \ge 1 - \frac{|C|}{\frac{|C|\cdot n}{2}} \ge 1 - \frac{2}{n}$ [# edges] □

[more rounds]

pf: $\mathcal{E}_j$ = event no edge in C is contracted in round j

$\Pr[\mathcal{E}_1] \ge 1 - 2/n$ [I saw]

$\Pr[\mathcal{E}_{j+1} \mid \mathcal{E}_1, \ldots, \mathcal{E}_j] \ge 1 - \frac{2}{(n-j)}$

pf: no edge of C is contracted in first j rounds

$\Rightarrow$ multigraph H — represents partition A, B, with $A = T_1 \cup \ldots \cup T_\ell$, $B = T_1' \cup \ldots \cup T_{\ell'}'$

— has $n-j$ vertices [I lose 1 vertex per contraction]

$\Rightarrow$ A, B valid cut in H w/ value $|C|$

clm: $\boxed{\text{any}}$ cut in H represents cut in G

sketch: undo merges

cor: H has $\boxed{\text{min}}$ cut C

$\Rightarrow$ H has $\ge |C|(n-j)/2$ edges [I as before]

$\Rightarrow$ random contracted edge in C w/p $\frac{|C|}{|E(H)|} \le \frac{2}{(n-j)}$

cor: $\Pr[\text{no edge of } C \text{ ever contracted}] \ge 1/\binom{n}{2}$

pf: $= \Pr[\mathcal{E}_1 \cap \ldots \cap \mathcal{E}_{n-2}]$ [n-2 rounds to get 2 vertices]

[conditional]

$$= \underbrace{\Pr[\mathcal{E}_1]}_{\ge 1 - \frac{2}{n}} \underbrace{\Pr[\mathcal{E}_2 \mid \mathcal{E}_1]}_{\ge 1 - \frac{2}{n-1}} \underbrace{\Pr[\mathcal{E}_3 \mid \mathcal{E}_2, \mathcal{E}_1]}_{\ge 1 - \frac{2}{n-2}} \cdots \underbrace{\Pr[\mathcal{E}_{n-2} \mid \mathcal{E}_1, \ldots, \mathcal{E}_{n-3}]}_{\ge 1 - \frac{2}{\underbrace{n-(n-3)}_{=3}}}$$

$$= \frac{\cancel{n-2}}{n} \cdot \frac{\cancel{n-3}}{n-1} \cdot \frac{\cancel{n-4}}{\cancel{n-2}} \cdots \frac{2}{\cancel{4}} \cdot \frac{1}{\cancel{3}}$$

$$= \frac{2}{n(n-1)} = 1/\binom{n}{2}$$

cor: random contract produces a given min cut C w/p $\ge 1/\binom{n}{2}$

Q: isn't this terrible? [we want $\Pr \approx 1$]

idea: $\boxed{\text{amplify}}$ probability of success by repeating algorithm

algo: randomly contract G, k times [indep]

output best cut

cor: C some min cut in G. k random contractions will observe C, except w/p $\le \left(1 - \frac{1}{\binom{n}{2}}\right)^k$ [I k indep failures]

cor: $O(n^2 \lg n)$ random contractions of G will produce a min cut whp

pf: take $k = \binom{n}{2} \ln n$. $\left(1 - 1/\binom{n}{2}\right)^{\binom{n}{2}\ln n} \le e^{-\ln n} = 1/n$ ← $1+z \le e^z$ [I again]

thm: global min cut in $O(n^4 \lg n)$ time whp

sketch: $O(n^2 \lg n)$ random contractions of G, output best cut seen

each $O(n^2)$ time; failure prob $1/n$

today: randomized algo — recap — global mincut in undirected G — vs directed (s,t)-cut — random contraction [I w/ repeat] [I skip ...]

next lecture: linear programming

logistics: pset 6 due F[?] pset 7 out F[?]

clean up

rmk: whp runtime [I didn't need coloring, but could have applied it]

runtime of algo is $\boxed{\text{fixed}}$

algo may be incorrect [I no way to know whether found min cut]

[I prev algo always correct, runtime may vary]