LECTURE 3

15/10/2020

QUICKSORT

INPUT : set S of distinct numbers

ALGORITHM

if $|S| = 1$ then OUTPUT S and STOP

choose uniformly at random $p \in S$ ... pivot

split S : $S^- = \{a \in S \mid a < p\}$

$S^+ = \{a \in S \mid a > p\}$

OUTPUT QUICKSORT(S^-), p, QUICKSORT(S^+)

ANALYSIS

- worst case .... $\Theta(n^2)$ $\qquad n = |S|$
- average case : what is the expected number of comparisons ?

fix an input $x_1, x_2, \dots, x_n$

let $y_1 < y_2 < \dots < y_n$ be the same values in an increasing order

define an indicator variable

$$X_{ij} = \begin{cases} 1 & \text{if } y_i \text{ and } y_j \text{ are compared at any time by the algorithm} \\ 0 & \text{otherwise} \end{cases}$$

!! $\forall i, j$: y_i and y_j are compared at most once

$\Rightarrow$ total number of comparisons $X = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}$

$$\mathbb{E}[X] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \mathbb{E}[X_{ij}] \quad ? \text{ by linearity of expectation}$$

The main QUESTION: for fixed i, j, what is the probability that y_i, y_j compared?

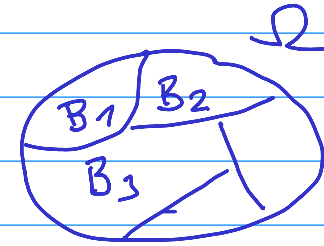
→ Fix i and j, assume $y_i < y_j$

for $\ell = 1, 2, \dots$

let B_ℓ be the event that on reccursion level ℓ, a pivot $p \in \{y_i, \dots y_j\}$ is chosen for the first time

(i.e., ℓ is the last level of recursion in which y_i and y_j are together)

2 $\quad B_\ell \cap B_{\ell'} = 0 \qquad \ell \neq \ell'$



3 $$\bigcup_{\ell=1}^{n} B_\ell = \Omega$$

Assume B_ℓ occurres

4 $$\Pr[y_i \text{ and } y_j \text{ are compared} \mid B_\ell] = \frac{2}{j-i+1}$$

$$\underbrace{\{y_i, y_{i+1}, \dots, y_j\}}_{j-i+1}$$

By the theorem of COMPLETE PROBABILITY

$$\Pr[y_i \text{ and } y_j \text{ are compared}] = \sum_{\ell=1}^{n} \Pr[y_i, y_j \text{ compared} \mid B_\ell] \cdot \Pr[B_\ell] =$$

$$= \sum_{\ell=1}^{n} \frac{2}{j-i+1} \cdot \Pr[B_\ell] = \frac{2}{j-i+1} \Rightarrow E[X_{ij}] = \frac{2}{j-i+1}$$

$$E[X] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{2}{j-i+1} = 2 \sum_{i=1}^{n-1} \underbrace{\left(\frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n-i+1}\right)}_{\leq H_n} \leq$$

$$\leq 2n \cdot H_n$$

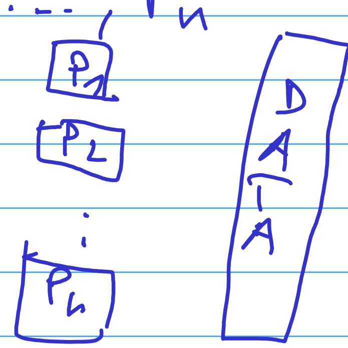
THM: For any input, the expected number of comparisons made by Quicksort is $\leq 2n H_n$.

LAS VEGAS type algorithm:
- always correct solution
- time depends on the random choices

FACT: $\frac{1}{4} \leq \left(1-\frac{1}{n}\right)^n \leq \frac{1}{e} \leq \left(1-\frac{1}{n}\right)^{n-1} \leq \frac{1}{2}$ for $n \geq 2$

CONTENTION RESOLUTION IN A DISTRIBUTED SYSTEM

PROBLEM: n identical processes $P_1, P_2, \ldots, P_n$
single shared database



- processes operate synchronously in discrete rounds
- no direct communication between the processes
- database can be accessed by at most one process in every round

GOAL: device a protocol that ensures access to the database for all processes "fast".

ALGORITHM ACCESS
- in each round: with probability p try to access the database

$p = \frac{1}{n}$

event $A_{i,t}$ - process P_i succeeds in round t

$$\Pr[A_{i,t}] = p \cdot (1-p)^{n-1} = \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1} \geq \frac{1}{e \cdot n}$$

event $F_{i,t}$ – the failure event that P_i does not succeed in any of the first t rounds

$$\Pr[F_{i,t}] = \prod_{r=1}^{t}\left(1 - \Pr[A_{i,r}]\right) \le \left(1 - \frac{1}{e\cdot n}\right)^t =$$

$$= \left[\left(1 - \frac{1}{en}\right)^{en}\right]^{\frac{t}{en}} \le \left(\frac{1}{e}\right)^{\frac{t}{en}}$$

for

$t = 2\cdot n\cdot e\cdot \ln n$

$$= \left(\frac{1}{e}\right)^{2\ln n} = \frac{1}{n^2}$$

$\Rightarrow$ Probability that some P_i will not succeed

is $\le n\cdot \frac{1}{n^2} = \frac{1}{n}$

THM: With probability $\ge 1 - \frac{1}{n}$, all processes succeed in accessing the database at least once within $t = 2\cdot e\cdot n\cdot \ln n$ rounds, for $p = \frac{1}{n}$.

GLOBAL MINIMUM CUT

INPUT: multigraph $G=(V,E)$

OUTPUT: a cut, i.e., $\emptyset \neq S \subsetneq V$

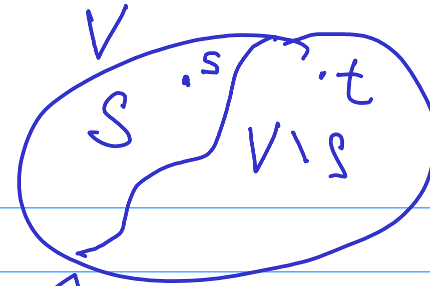
GOAL: minimize the size of the cut, i.e., # of edges between S and $V \setminus S$

IDEA: consider all $\binom{n}{2}$ pairs of vertices $s,t \in V$

for each pair, run a min s-t cut algorithm

OUTPUT the min cut you found

time: $\binom{n}{2} \cdot O(\text{time for min } s\text{-}t \text{ cut})$



Global min. cut

a better algorithm?

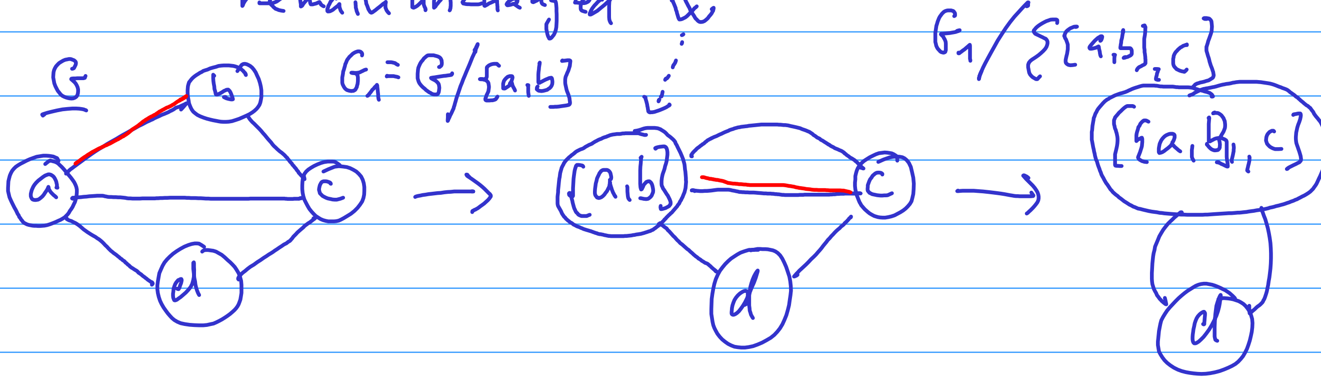
HW: $O(n \cdot \text{time for min } s\text{-}t \text{ cut})$

Given a $G=(V,E)$ and $\{u,v\} \in E$
multigraph

a contraction of $\{u,v\}$ $\qquad$ G/e

- replace u and v by a new vertex w
- for each $z \notin \{u,v\}$
  replace any edge $\{z,u\}$ by $\{z,w\}$
  $\qquad$ " $\qquad$ $\{z,v\}$ by $\{z,w\}$
- all other vertices and edges not adjacent to u or v remain unchanged

E.g.



Note: when only 2 vertices are left, we have a global cut - $S=\{a,b,c\}$ in the example