MP1933026

Problem 1

Weighted min-cut algorithm

input: multi-graph **G(V,E)**,where every edge $e \in E$ is associated with a positive real weight w_e

while **|V|>2** do

choose an edge $uv \in E$ with probability proportional to the weight of uv ;

G = Contract(G,uv) ;

return **C = E**

Proof.

if $\sum_l w_l \in C$ is a min-cut in G,**then** $\sum_m w_m \in E \geq \frac{|V| \sum_l w_l \in C}{2}$

so $p_i = 1 - \frac{\sum_l w_l \in C}{\sum_m w_m \in E_i}$

$\geq 1 - \frac{2}{|V_i|}$

$= 1 - \frac{2}{n-i+1}$

p_{corret} = Pr[a weighted minimum cut is returned by weighted algorithm]

$\geq$ Pr[C is returned by weighted algorithm]

= Pr[$e_i \notin C$ for all i =1,2,...,n-2]

$= \prod_{i=1}^{n-2} \Pr\left[e_i \notin C | \forall j < i, e_j \notin C\right]$

$\geq \prod_{i=1}^{n-2} \left(1 - \frac{2}{n-i+1}\right)$

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

Problem 2

(1)

Consider the probability of two vertex not in same subset S or T:

$Pr[X_u, X_v \text{not in the same subset}] = \frac{2 * C_{n-2}^{\frac{n}{2}-1}}{C_n^{\frac{n}{2}}} = \frac{n}{2(n-1)}$where u,v $\in E$

So $E[|E(S,T)|] = \sum_{uv \in E} Pr[X_u, X_v \text{not in the same subset}] = \frac{nm}{2(n-1)}$

(2)

Generate random subset $S \in F$

input: collection of subset F,p=$\frac{1}{2}$

while|F|>1 do:

$X = R(p)$

if X=0 delete the first half F else delete the last half F

return S=F

With p=$\frac{1}{2}$,such that $Pr[X = 1] = Pr[X = 0] = \frac{1}{2}$,the probability of pick a specific S from F is $\frac{1}{2^{log|F|}}$,the total probability is $|F| * \frac{1}{2^{log|F|}} = 1$,so the algorithm generates random subset $S \in F$ uniformly at random. The number of times that the random sources is called is log|F|.

Problem 3

Randomized algorithm for testing isomorphism

input: two rooted trees T_1, T_2 ,a finite set S

for each vertex v in T_1orT_2:

if v is leaf :set $P_v = x_0$

else: with the height h of v and his children,set $P_v = \prod_{i=1}^{k}(x_h - P_{v_k})$

pick $r_1, r_2, ...r_h \in S$ uniformly and independently at random

if $P_{T_1}(r_1, r_2, ...r_h) = P_{T_2}(r_1, r_2, ...r_h)$ then return True else return False

By the above method,it is clear that the degree of $P_{T_1}(\vec{x})$ or $P_{T_2}(\vec{x})$ is at most n(n is the vertex of the T_1orT_2),then by Schwartz-Zippel Theorem ,$Pr[P_{T_1}(\vec{r}) = 0] \leq \frac{n}{|S|}$,so the probability of being incorrect $\leq \frac{n}{|S|}$.

Problem 4

Fingerprint for testing sequence

input: two sequence$A = \{a_1, a2, ...a_n\}, B = \{b_1, b_2, ...b_n\}$

construct $f_A(x) = \prod_{i=1}^{n}(x - a_i)$,$f_B(x) = \prod_{i=1}^{n}(x - b_i)$,

in the meantime get $t_a = max(A), t_b = max(B)$

if $t_a \neq t_b$,return False

if $t \leq n$:set $p \in [(nlogn)^2, 2(nlogn)^2]$ else :set $p \in [n^2 log(n)log(t), 2n^2 log(n)log(t)]$

where p is a uniform random prime chosen from list and set the finite field $Z_p = [p]$

choose a element **r** uniformly from[p]

if $f_A(r) \bmod p = f_B(r) \bmod p$ then return True else return False

if $t \leq n$,then by Theorem(Lipton 1989),the probability of $A \neq B$ is $Pr[FING(A) = FING(B)] = O(\frac{1}{n})$

if $t \geq n$:

$Pr[FING(A) = FING(B)] = Pr[f_A(r) = f_B(r)|f_A \neq f_B]Pr[f_A \neq f_B]$

$+Pr[f_A(r) = f_B(r)|f_A = f_B]Pr[f_A = f_B]$

$\leq Pr[f_A(r) = f_B(r)|f_A \neq f_B] + Pr[f_A = f_B]$

note that the degree of f_A, f_B are at most n and r is chosen uniformly from [p],

By the Schwartz-Zippel theorem for univariate polynomials

$Pr[f_A(r) = f_B(r)|f_A \neq f_B] \leq \frac{n}{p} = O(\frac{1}{nlogt})$

with non-zero coefficient $c \leq t^n$ .The event $f_A = f_B$ occurs only if c mod p =0,

which means:

$Pr[f_A = f_B] \leq Pr[c \text{ mod } p = 0]$

$= \frac{\text{number of prime factors of c}}{\text{number of primes in}[n^2log(n)log(t), 2n^2log(n)log(t)]}$

$\leq \frac{nlog(t)}{\pi(2n^2log(n)log(t)) - \pi(n^2log(n)log(t))}$

$= O(\frac{nlog(t)}{n^2log(t)})$

$= O(\frac{1}{n})$

Combing everything together ,we have the error probability is $O(\frac{1}{n})$,time complexity is $O(n)$

Problem 5

$lnZ - \epsilon \leq ln\widehat{Z} \leq lnZ + \epsilon \longrightarrow e^{-\epsilon Z} \leq \widehat{Z} \leq e^{\epsilon}Z$

$E[X_i] = p_i$

$E[\hat{p_i}] = E[\frac{1}{s}\sum_{j=1}^{s} X_i^{(j)}] = \frac{1}{s} * s * E[X_i] = p_i$

$E[ln\widehat{Z}] = E[ln\prod_{i=1}^{n} \hat{p_i}] = \sum_{i=1}^{n} p_i = lnZ$

It is not hard to know that $Var[X_i] = p_i * (1 - p_i)$ and because $p_i \geq \frac{1}{2}$

$Var[ln\widehat{Z}] = Var[\sum_{i=1}^{n} \frac{1}{s}\sum_{j=1}^{s} X_i^{(j)}] = \frac{1}{s}\sum_{i=1}^{n} p_i * (1 - p_i) \leq \frac{n}{4s}$

$Pr[e^{\epsilon}Z \leq \widehat{Z} \leq e^{\epsilon}Z]$

$= Pr[lnZ - \epsilon \leq ln\widehat{Z} \leq lnZ + \epsilon]$

$= Pr[|ln\widehat{Z}| \leq \epsilon]$

$Pr[|ln\widehat{Z}| > \epsilon]$

$\leq \frac{1}{\epsilon^2}Var[ln\widehat{Z}]$

$\leq \frac{n}{4s\epsilon^2} \leq \delta$

When $s \geq \lceil \frac{n}{4\delta\epsilon^2} \rceil$,this probability is at most δ ,so does the probability above

Problem 6

(1) the last two-choice half will change the bad situation,and othe affet is constant,so the maximum load is $O(loglogn)$

(2) the first two-choice half will not affect last random half ,and other affect is constant,so the maximum load is $O(\frac{logn}{loglogn})$

(3) the maximum load is $O(loglogn)$