CMSC 39600 - Lec #2 (Sep 27)

Today
- PCP Theorem - & its variants
- Hardness of approx clique
- Coding Theory - Prelims
-

PCPs - definition

Verifier:

$(r, q, m, t)_{\Sigma}$-restricted verifier
- randomized TM with oracle access to proof π over alphabet Σ

On input x (of length n)
- tosses $r(n)$ coins
- probes $q(n)$ locations in a proof of size $\leq m(n)$
- runs in time $t(n)$

ACC/REJ verdict.

$V^{\pi}[x; R]$

PCP class

If L has a $(r, q, m, t)_{\Sigma}$-rest verifier s.t

C: $\forall x \in L$, $\Pr_R[V^{\pi}[x; R] = \text{ACC}] \geq c(n)$

S: $\forall x \notin L$, $\Pr_R[V^{\pi}[x; R] = \text{ACC}] \leq s(n)$

$L \in \text{PCP}_{c,s}\left[\begin{matrix} r \\ q \\ m \\ t \end{matrix}\right]_{\Sigma}$

$m \approx 2^{r+q}$

If ~~$r = poly$~~, $\Sigma = \{0,1\}$, $t = poly$, omit n.

Remark: • Verifier - adaptive or non-adaptive
a PCP , na PCP

• $m(n) \le q \cdot 2^r \le t \cdot 2^r$ (- non-adaptive
$\in 2^{r+q}$ (- adaptive)

• $q(n) \le t(n)$.

• $PCP_{c,s}[r, q] \subseteq NTIME(2^{r+q})$

Clearly,

$NP = PCP_{1,0}[0, poly(n)]$, $BPP = PCP_{2/3,1/3}[poly(n), 0]$

PCP Theorem [AS, ALMSS]

$\exists q$, $NP = \bigcup_{c>0} PCP_{1,1/2}[c\log n, q]$

Earlier PCP results:

• [BFL] + [FRS]: $NEXP = PCP_{1,1/2}(poly, poly)$

[BFLS] : $NP \subseteq PCP_{1,1/2}\left[\begin{array}{l} q = t = polylog\, n \\ m = n \cdot polylog\, n \end{array}\right]$

[FGLSS] $NP \subseteq PCP[polylog\, n, polylog\, n]$

[AS] $NP = PCP(\log n, \sqrt{\log n})$

[ALMSS] $NP = PCP(\log n, k)$

Strengthening of PCP Theorem

1. $\forall \epsilon > 0$, $\exists$ alphabet Σ, $|\Sigma| = \text{poly}(1/\epsilon)$

$$NP \subseteq PCP_{1,\epsilon}^{\Sigma}[O(\log n), 2] \qquad \text{[Raz]}$$

2. $\forall \epsilon > 0$,

$$NP \subseteq PCP_{1-\epsilon, \frac{1}{2}+\epsilon}[O(\log n), 3] \qquad \text{[Håstad]}$$

— Furthermore, verifiers actions

$$\pi_{i_1} \oplus \pi_{i_2} \oplus \pi_{i_3} = b. \ ?$$

—

Cor: $\forall \epsilon > 0$ NP-hard to approx MAX-3SAT to within $\frac{8}{7} - \epsilon$

(Exercise).

—

3. $\forall \epsilon > 0$

$$NP \subseteq PCP_{1, 1/2}\left[\begin{matrix} O(\log n), & q \\ n \cdot \text{poly} \log n & \end{matrix}\right] \qquad \text{[BS+Dinur]}$$

—

Hardness of approximating Clique

CLIQUE

gap-Clique$_\alpha$ $\langle G, k \rangle$

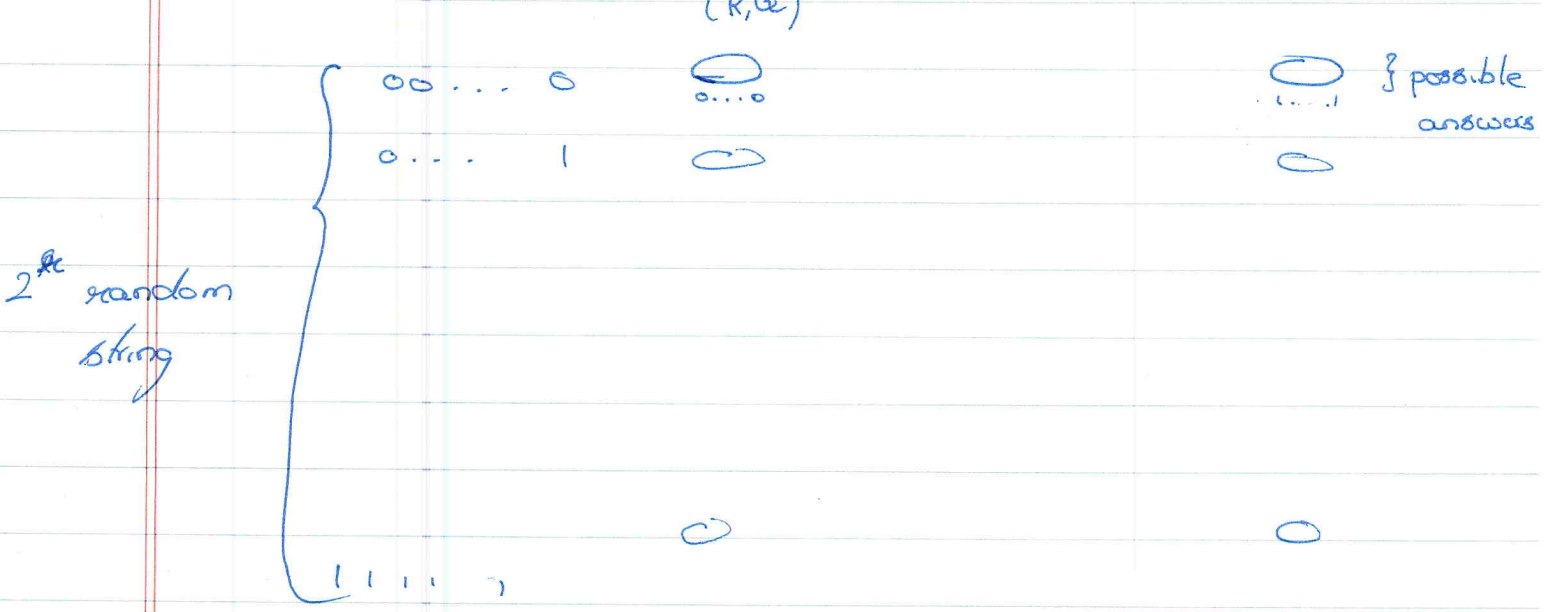
YES = $\{\langle G, k \rangle \mid \exists$ clique of size $\geq k\}$

NO = $\{\langle G, k \rangle \mid \forall$ cliques, size $\leq \alpha k\}$

Thm: $\forall \alpha > 0$, gap-Clique$_\alpha$ is NP-hard.

Cor: Approximating Clique ~~to Gotler α At~~

Pf: Encode Verifiers action by graph G



$|V| = 2^{r+q} = R \cdot Q$

Edges: $(R,Q) \sim (R',Q')$ if they do not contradict each other.

$x \in L$

Comp: $\exists \pi, \Pr_R [V^\pi(x;R) = 1] \geq c$

$C_\pi = \{(R, Q_{\pi,R}) \mid V^\pi(x;R) = 1\}$

$Q_{\pi,R}$ – bit sequence seen by V on random coins R & proof π.

Clearly, C_π – clique.

$|C_\pi| = 2^{q_r} \cdot \Pr_R [V^\pi(x;R) = 1] \geq c \cdot 2^{q_r}$

Thus, $x \in L \Rightarrow$ MAX-CLIQUE$(G) \geq c \cdot 2^{q_r}$

Soundness: $x \notin L$

$\forall \pi, \Pr_R [V^\pi(x;R) = 1] \leq s \Rightarrow$ MAX-CLIQUE$(G) \leq s \cdot 2^{q_r}$

Clique corresponds to proof.

Corollary:

$NP \subseteq PCP_{c,s}[\log, q] \Rightarrow$ gap-CLIQUE$_{s/c}$ is NP hard (time of reduction 2^{2r+q})

Sequential repetition

$NP \subseteq PCP_{c^k, s^k}[k \log q, kq] \Rightarrow$ gap-CLIQUE$_{s^k/c^k}$ is NP hard

Can k - super constant ?

Reduction becomes super polynomial time.

Recycle Random Bits:

$r + 2k$ - random bits suff.

$$PCP_{c,s}[r, q] \subseteq PCP_{c^{k/2}, 2s^k}[r+2k, kq]$$

Hastads 3-query PCP

$$NP \subseteq PCP_{1-\varepsilon, 1/2+\varepsilon}[\log, 3] \subseteq PCP_{1/2-\delta, \frac{1}{2^{k-1}}+\delta}[\log + 2k, kq]$$

$$\text{Set } k = \log \quad s/c \approx (2-\delta')^k \qquad 2^{r+q} \simeq N \cdot 2^{5k}$$

$$k = \log n/p \qquad s/c \approx c$$

approx $N^{1/5}$ gap-Clique$_{N^{1/5}}$ is NP-hard.

Recycle Queries:

gap-Clique$_{\frac{1}{N^{1/2-\varepsilon}}}$ is NP-hard

gap-Clique$_{\frac{1}{N^{1-\varepsilon}}}$ is NP-hard under randomized reductions