Lecture 24 – Hardness

[[We'll talk about a bunch of hard problems...]]
[[Well, we don't know how to prove anything is hard, so it's always...]]

Hardness: Assumption + Reductions
$=$ algorithms [[all hardness research is arguably algs research]]

e.g.: $\boxed{3\text{Sat} \notin P}$ (NP ≠ P) cook-levin $\xrightarrow{*\text{ alg } R *}$ Indep-Set $\notin P$

$\exists$ 3CNF φ ⟶ (G, k)

thm: • φ sat $\Rightarrow$ G has I.S. of size $\geq k$

• φ unsat $\Rightarrow$ G doesn't have " " " " "

[[Yes, there's a wonderful thy. of NP-completeness based on $\boxed{}$ But insuffic...]]

☹ "worst-case hardness": [[for crypto...]] better if $\exists$ eff. rand. alg for generating hard insts

☹ how hard?? $\in TIME(n^{\log n})$?
$\in TIME(2^{\sqrt{n}})$? $O(1.3^n)$? [[1.31^n ✓ (Hertli '11)]]

[[But remember, can't prove anything $\therefore$ need more assumps. Try to be economical.]]

last time: Approx-CVP (worst case) $\xrightarrow[\text{(quantum)}]{\text{poly-time}}$ LWE (avg case) ⟶ Cryptomania

LWE: earning with errors

Worst-to-avg: Rare!

$\approx$ solving noisy lin. eqns mod $q = \text{poly}(n)$

LPN: the mod 2 version Params n, $0 < \varepsilon < 1/2$

"secret" $s \sim \mathbb{F}_2^n$ chosen unif.

- Alg. can ask for (noisy) $\mathbb{F}_2$-lin eqn:
  Gets "$a_1 s_1 + \cdots + a_n s_n = b$",
  where $a \sim \mathbb{F}_2^n$ unif. rand,
  b = correct val. except w/ prob. ε
- Task: output s.

Assumption: $\forall$ const $\varepsilon > 0$ not solvable in poly-time [or worse?]

$\Rightarrow$ OWFs, very efficient SKE ["minicrypt"]

$\overset{?}{\Rightarrow}$ PKE [Ale'03] $\Rightarrow$ PKE if take $\varepsilon = \Theta(1/\sqrt{n})$ [questionable?]

Fastest alg: [BKW'03] (CMU) "Clever Gaussian Elim" [Quite beautiful]

- Time, samples $2^{O(n/\log n)}$. [even for $\varepsilon = \frac{1}{2} - \frac{1}{2}e^{\dots}$, $2^{-n^{.99}}$]

Rem: Fastest alg using only poly(n) eq's: $2^{O(n/\log\log n)}$. [Lyub.'05, $n^{1+o(1)}$ samps]

[Rem: [Katz-Shin-Smith-10]: search $\equiv$ decision $\Leftarrow$ dist. from random.]

[Fei'02 Ale'03]

Sparse-LPN: Fix $k \geq 3$. Eq^n's "$a_1 s_1 + \cdots + a_n s_n = b$" have only k nonzero a_i's

Say alg gets m examples, [poly time].
If $m \gg n^k$, can get all eqs. many times $\to$ kill noise.
$\exists$ poly-time algs even for $m = \tilde{\Theta}(n^{k/2})$ [FO'06, OW'13]

e.g. Assump: for $k=3$, $m = O(n)$ no poly-time alg [or even $2^{o(n/\log n)}$?]

... can even find $\tilde{s}$ with $\frac{1}{2} + \frac{1}{2}\varepsilon$ agreement w/ s.

Thm: Grigoriev '01 [Schoenebeck '08, Tuls. 09, Chan '13]
Lasserre/Parrilo/SOS hierarchy doesn't solve until deg. $\Omega(n)$
(time $2^{\Omega(n)}$).

[more on which later]

Thm: [Hås '99] $(\frac{1}{2}+\varepsilon, 1-\varepsilon)$-approx, 3LIN(mod 2) is NP-hard

3SAT $\xrightarrow{\text{PCP thm}}$ (.999, 1)-approx Max-3Sat. $\xrightarrow{\text{Parallel Rep}}$ "Label-Cover"

[Moshkovitz-Raz '10]

Fourier analysis gadget.

$\forall \varepsilon > 0$, Given $O(n)$ eqns over n vbls
$x_i + x_j + x_k = 0/1 \mod 2$
NP-hard to tell if $\geq (1-\varepsilon)$-frac. of eq's sat'ble
$\leq (\frac{1}{2}+\varepsilon)$- " . . . . . .

☺ Sparse-LPN assump. gives natural, effic. "avg-case" way to generate hard-seeming insts.
[Stronger assump than P≠NP, of course, but get more]

Great basis for further reductions
⟶ best known Max-Cut, 2Sat inapprox, other CSPs [Hås, TSSW]

⟶ TSP inapprox [PV'06]

⟶ Dense-k-Subgraph, Non-unif. Sparsest Cut, GISo hardness...

Yields poly-time gen'ble (the arguably "unnatural") hard insts for

[Let's talk about 3SAT, much better studied. ∞ # papers, heuristics algs for it]

Generating "hard" satisfiable 3sat instances trickier.

[planting? ⟶ if $s_i = 0$, more likely to see $\bar{s}_i$]

Q: Suppose choose a unif rand. m-clause, n-vbl 3SAT inst. Is it sat'ble?

A: Empirically/stat phys.: $\exists\, c \approx 4.2$ s.t.

$m > cn \Rightarrow$ unsat whp — Thm: true for $c \approx 4.5$

$m < cn \Rightarrow$ sat whp. — Thm: true for $c \approx 3.5$, & $\exists$ effic. alg.

~~thm:~~

¿Assump? [[? not very useful?]]: For $c = 4.2$, rand 3sat reqs. $2^{\Omega(n)}$ time.

[[Questionable in thy. b/c if you get c wrong by even a millimetre, it's false.]]

better: Feige's R3Sat assump.: For all suff. large const. c:

"≈" [ALe '03] Sparse 3-Lin

$\not\exists$ a poly alg. which refutes almost all rand 3sat insts w/ $m = \lfloor cn \rfloor$

[[As w/ 3Lin, known to be broken if $c = \tilde{\Theta}(n^{.5})$]]

[[outputs a proof of unsat; alternatively, weaker, just have to say "typical" w.h.p., but never say "typical" if sat'ble]]

CNF-Sat in "worst-case": Trivial alg: $2^n\,\mathrm{poly}(m)$ time.

3-Sat: $\tilde{O}(1.31^n)$ best known.

easy $\tilde{O}((4/3)^n)$ alg: Walk-Sat [[Schöning '99]]

① pick rand asgn

② take any unsat clause, flip rand vbl in it

Repeat ② $O(n)$ times [[to try to find sat. asgn]]

Repeat ①&② $\tilde{O}((\frac{4}{3})^n)$ times.

$2^{n/\log n}$ time? $2^{\sqrt{n}}$ time?? $n^{\log\log n}$ time?? [huge distinctions]

ETH: $\exists \delta > 0$ s.t. 3-Sat reqs time $2^{\delta n}$ time. [$\forall$ suff large n]
& inf. o. for n

[Imp, Paturi '99]

[Once you have such a hypoth, reduction size/time/other params key...]

[Suppose, e.g. you prove Subset-Sum is NP-hard.]

[ancient Garey Johnson Tarjan '76]

thm: 3Sat (input size n) $\xrightarrow{R: O(n^3) \text{ time}}$ Planar-Ham-Path (input size $O(n^2)$) [vtcs & edges]

Assuming ETH, implies Planar-Ham-Path on size-N inputs reqs time $2^{\Omega(N^{1/2})}$

rem: Solvable in $2^{O(\sqrt{N})}$ time

[Strong, but not awesome.]

$\therefore$ Alg. effic. of reductions is important!

rem: Some NP-complete probs are solvable in, eg, $2^{n^{.01}}$ time:
eg: Given graph $\exists$? indep set of size $\geq n^{.005}$?

rem: "most" classic NPC reductions have output size $O(n)$.
eg: ETH $\Rightarrow$ 3-Col, Ham-Path, Indep-Set req. $2^{\Omega(n)}$ time.

[IPZ, inc '01] SETH: $\forall \delta > 0$ CNF-Sat not solvable in $2^{(1-\delta)n}$ poly(m) time.
(strong)

facts: • $\exists$ $1.3^k \cdot$ poly(n) alg for "Does G have V.C. of size k?"
• But ETH $\Rightarrow$ no $2^{o(k)}$ poly(n) alg.

• $\exists$ $n^{O(k)}$ alg for k-Clique. ETH $\Rightarrow$ no $f(k)\, n^{o(k)}$ alg

• Max-Cut solvable in time $2^{O(\text{treewidth}(G))}$ poly(n).
• ETH $\Rightarrow$ not in $2^{o(tw(G))}$ poly(n).

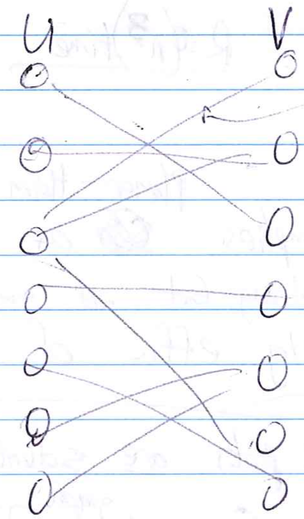
[Patrascu-Williams] : SETH $\Rightarrow$ "k-Sum" on n integers
($\exists$ some k adding to 0?) reqs $n^{\Omega(k)}$ time
[[For $k = n^{.99}$; handling the famed 3-Sum would be cool.]]

Hardness of approx:

[[LC : inapprox :: 3Sat : NPC]]

Mother of all probs: Label Cover(q):

2CSP. Input:
(biregular)



each edge (u,v) has a fcn
$\pi_{u,v} : [q] \to [q]$ written on it.

Goal: . assgn. $A : U \cup V \to [q]$

. (u,v) constr sat iff
$A(v) = \pi_{uv}(A(u))$

"Projection constrs".

thm [Raz '94] $\forall \delta > 0 \ \exists q = \text{poly}(1/\delta)$ s.t.
$(\delta, 1)$-approx'ing Label-Cover(q) is NP-hard.

$\Rightarrow$ Many inapprox results; e.g. $(\frac{1}{2}+\varepsilon, 1-\varepsilon)$-inapprox for 3-Lin
$(\frac{7}{8}+\varepsilon, 1)$- " " 3-Sat
$(n^{\varepsilon}, n^{1-\varepsilon})$- " " Max-Indep-set

☹ Raz: 3Sat size n $\longrightarrow$ LC(q) size $n^{O(\log 1/\delta)}$.

$\therefore$ ETH $\Rightarrow$ $(.51, .99)$-approx 3-Lin reqs $2^{n^{.000\cdots01}}$ time.

7

☺ [Mosh-Raz '10] 3Sat $\longrightarrow$ LC($2^{poly(1/\delta)}$) ☹

n size $n^{1+o(1)}$ even for $\delta = o(1)$

[$\frac{1}{\log\log n}$?]

∴ ETH ⇒ (.51, .99)-approx 3-Lin eqs $2^{n^{1-o(1)}}$ time ☺

[Khot '02] <u>UGC</u>: UG = Label-Cover where π_{uv}'s are bijs

↳ "$\forall \delta > 0\ \exists q$ s.t. $(\delta, 1-\delta)$-approx. UG is NP-hard."

Maybe should have said "not in P".

⇒ many sharp inapprox results: whatever

e.g. [Ragh '09] $\forall_k$ CSPs, deg-k Lasserre/Parrilo best (α, β)-approximates,

UGC ⇒ getting $(\alpha+\varepsilon, \beta-\varepsilon)$ is NP-hard $\forall \varepsilon > 0$.

[Basically, assuming UGC closes NP-hardness of all CSPs].

But.... [ABS'10] $\forall \delta > 0$, $(\delta, 1-\delta)$-approx'ing UGC doable in $2^{n^{O(\delta^{1/3})}}$ time.

Assuming ETH, ~~[illegible]~~ this is as <u>easy</u> as an NPC problem can be.

UGC De facto easy??

- no known "avg-case hard" instances
- Lasserre/Parrilo deg 4 solves all known "tricky instances"