October 14, 2022 10:46 AM

Reducing (f-sparse) k-SAT → Subset Sum

Thm (Abboud, Bringmann, Hermelin, Shabtay '19)

Assuming SETH, no alg'm for subset sum for n integers & target T in $O(T^{1-\delta} \cdot 2^{o(n)})$ time, for any fixed const $\delta > 0$.

Pf: first idea: follow textbook NP-completeness pf for subset sum

Suppose subset sum has $O(T^{1-\delta} \cdot 2^{o(n)})$ alg'm

Given f-sparse k-CNF formula F with vars $x_1, \dots, x_n$ & clauses $C_1, \dots, C_m$ $(m \le fn)$

Ex $F = \underbrace{(x_1 \vee \bar{x}_2)}_{C_1} \wedge \underbrace{(\bar{x}_1 \vee x_3)}_{C_2}$

k=2, f=2

a satisfying assignment: $x_1=1, x_2=0, x_3=1$

let f_i = frequency of x_i

	x_1	x_2	x_3	C_1	C_2	x_1	x_2	x_3
C_1	0	0	0	1	0	0	0	0
	1	0	0	1	0	0	0	0
	1	1	0	1	0	0	0	0
C_2	0	0	0	0	1	0	0	0
	0	0	1	0	1	0	0	0
	1	0	1	0	1	0	0	0
x_1 (f)	2	0	0	0	0	1	0	0
($f-f_i$)	0	0	0	0	0	1	0	0
x_2	0	2	0	0	0	0	1	0
	0	1	0	0	0	0	1	0
x_3	0	0	2	0	0	0	0	1
	0	0	1	0	0	0	0	1
T	2	2	2	1	1	1	1	1

(base ≥ 4)

1. For each clause C_j & each assignment φ ($\le 2^k - 1$) of its $\le k$ vars that satisfies C_j,

   Create number $z(C_j, \varphi) = [\underbrace{\quad\quad\quad}_{n}][\underbrace{0010\cdots0}_{m}][\underbrace{0\cdots\cdots0}_{n}]$

   i^{th} digit is
   - 1 if $x_i = 1$ in φ
   - 0 if $x_i = 0$ in φ
   - 0 if x_i not in C_j

   j^{th} digit is 1

2. For each var x_i & $\alpha \in \{0,1\}$,

2. For each var x_i & $\alpha \in \{0,1\}$,

create number $z(x_i, \alpha) =$ [ ] [0...0] [0..010..]

i^{th} digit is
$\begin{cases} f & \text{if } \alpha = 0 \\ f - f_i & \text{if } \alpha = 1 \end{cases}$
rest are 0's

i^{th} digit is 1

3. let T = [ff ... f] [1111] [1111]

with base $\geq \max\{2^k, f\}$

But numbers too big
(even if base = 2,
$T \gg 2^{2n+m} \gg 2^n$)

next/new idea - reduce # vars & # clauses

divide into $\frac{n}{B}$ groups of B vars → super-vars
& $\frac{m}{B}$ groups of B clauses → super-clauses

Lemma (from additive combinatorics, Behrend '46)
("average-free set")   obv. bd. - $O(S^N)$

For any N, S,
$\exists$ set of N numbers $\ell(1), \ldots, \ell(N) \in [S^{O(1/\varepsilon)} N^{1+\varepsilon}]$
which is S-average-free

i.e. $\frac{\ell(i_1) + \ldots + \ell(i_s)}{s} \neq \ell(i)$ for any $s \leq S$

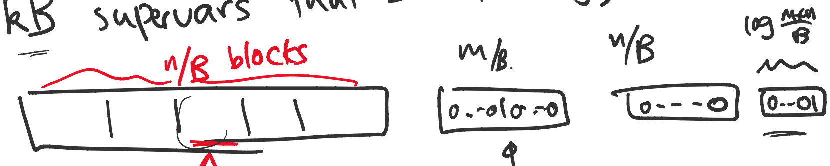
unless $i_1 = \ldots = i_s = i$.

Apply Lemma with $N = 2^B$, $S = fB$

$\Rightarrow$ numbers bdd by $R = (fB)^{O(1/\varepsilon)} (2^B)^{1+\varepsilon}$

1. For each super-clause C_j & each assignment φ of its $\leq kB$ supervars that satisfies C_j,

create $z(C_j, \varphi) =$



in base 2

i^{th} block is $\ell(\alpha)$ if supervar $x_i = \alpha$ in φ

0 if supervar x_i is not in C_j

j^{th} bit is 1

2. For each super-var x_i & each $\alpha \in [2^B]$,

create $z(x_i, \alpha) =$

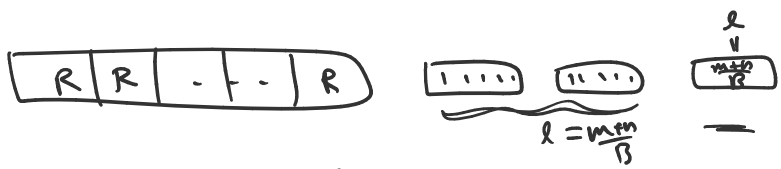
(f_i = frequency of x_i)

i^{th} block is $R - f_i \cdot \ell(\alpha)$

all other blocks 0

i^{th} bit is 1.

3. let $T =$



Correctness:

only 1 way to express $\overbrace{1 \cdots 1}^{\ell}$ as sums of ℓ powers of 2

want $\ell(\alpha_1) + \dots + \ell(\alpha_{f_i}) + \cancel{R} - f_i \ell(\alpha) = \cancel{R}$

$$\Leftrightarrow \frac{\ell(\alpha_1) + \dots + \ell(\alpha_{f_i})}{f_i} = \ell(\alpha)$$

$$\Leftrightarrow \alpha_1 = \dots = \alpha_{f_i} = \alpha$$

ensure consistency! $\qquad f_i \leq fB$

Runtime Analysis:

each number has # bits

$$\approx \frac{n}{B} \log R + \frac{m}{B} + \frac{n}{B} + O(\log n)$$

$$\approx \frac{n}{B}\left(\log\left[(fB)^{O(1/\varepsilon)} (2^B)^{1+\varepsilon} \right] \right) + \frac{m+n}{B}$$

$m \le fn$

$$\approx \frac{n}{B}\left(O\left(\tfrac{1}{\varepsilon}\right) \log(fB) + (1+\varepsilon) B \right) + O\left(\frac{fn}{B}\right)$$

$$\approx (1+\varepsilon) n + O\left(\frac{1}{\varepsilon} \frac{n}{B} \log(fB) \right) + O\left(\frac{fn}{B}\right)$$

set $B = \frac{1}{\varepsilon^2} f \log f$

$$\approx (1 + O(\varepsilon)) n < \left(1 + \frac{\delta}{2}\right) n$$

Set $\varepsilon = \frac{\delta}{1000}$

$$\Rightarrow \quad T \lesssim 2^{(1+\delta/2) n}$$

$$\# \text{ numbers} \le O\left(\frac{m}{B} \cdot (2^B)^{kB} + \frac{n}{B} \cdot 2^B \right)$$

$$= O(n) \quad \text{if } f, k, \varepsilon \text{ const.}$$

$$\text{total time} \le T^{1-\delta} \cdot 2^{o(n)}$$

$$\lesssim \left(2^{(1+\delta/2) n} \right)^{1-\delta} \cdot 2^{o(n)}$$

$$\lesssim 2^{(1-\frac{\delta}{2}) n + o(n)}$$

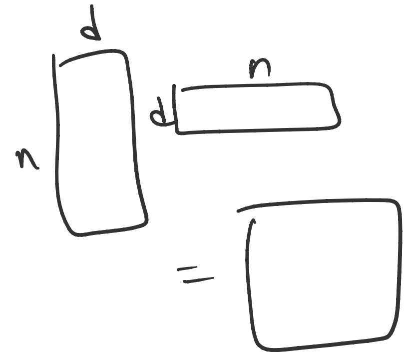
$(1+\frac{\delta}{2})(1-\delta) < 1 - \frac{\delta}{2}$

Contradicts SETH! $\square$

Problem: Boolean Orthogonal Vectors (OV)

Given sets A, B of n vectors in $\{0,1\}^d$, decide $\exists\, a \in A, b \in B$ s.t. $a \cdot b = 0$

$$\sum_{i=1}^{d} a[i]\cdot b[i] = 0.$$



naive alg'm: $O(dn^2)$

or $O(M(n,d,n))$

$\le O(d^{\omega-2}n^2)$

Can we beat n^2?

$O(2^d n)$

or $O(n+4^d)$ ← good only when $d = o(\log n)$

OPEN: $O(n^{2-\delta})$ for $d \gg \log n$??

<u>OV Conjecture</u> No alg'm for OV in $O(d^{O(1)} n^{2-\delta})$ time