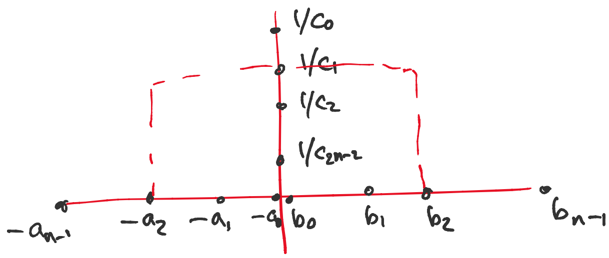

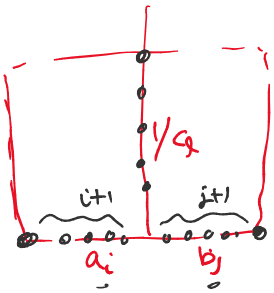

\# pts enclosed

$= i+1 + j+1$
$\quad + 2n-2-\ell+1$

$= i+j-\ell+2n+1$

$= h$

$\iff i+j=\ell.$

$$\text{area} = (a_i+b_j)\cdot\frac{1}{c_\ell} \le 1$$

$$\iff a_i+b_j \le c_\ell. \qquad \square$$

Conditional Lower Bds Based on SAT

(CNF-) SAT Problem Given CNF formula F with n vars, decide if $\exists$ satisfying assignment

e.g. $\underbrace{(x_1 \vee x_2 \vee \overset{\text{literal}}{\overline{x_3}})}_{\text{clause}} \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge \cdots$

k-SAT: version for max clause length k

Original P ≠ NP Hypothesis

No polytime alg'm for k-SAT for any const $k \geq 3$.

"Exponential-Time Hypothesis" (ETH)

No alg'm for k-SAT in $2^{o(n)}$ time for any const $k \geq 3$.

"Strong Exponential-Time Hypothesis" (SETH)

... $(2^{(1-\delta)n})$ time

"Strong Exponential-Time Hypothesis" (SETH)

No alg'm for k-SAT in $O(2^{(1-\delta)n})$ time

for a fixed const $\delta > 0$ indep of k. $\quad \equiv (2-\delta')^n$

Note: there are k-SAT alg'ms faster than 2^n for any fixed k.

[e.g. pick any clause $\alpha_1 \vee \dots \vee \alpha_k$
try all $2^k - 1$ possible settings of $\alpha_1, \dots, \alpha_k$

$$T(n) \le (2^k - 1)\, T(n-k)$$
$$\Rightarrow \quad (2^k - 1)^{n/k}$$

$$\left((2^k-1)^{\frac{1}{k}}\right)^n = \left(\left(2^k\left(1-\tfrac{1}{2^k}\right)\right)^{\frac{1}{k}}\right)^n = \left(2 - \Theta\left(\tfrac{1}{k2^k}\right)\right)^n$$

e.g. $k=3$: $7^{n/3} < 1.92^n$

$k=4$: $15^{n/4} < 1.97^n$

⋮

Schöning '99: $\left(2 - \frac{2}{k}\right)^n$]

Sparsification Lemma (Impagliazzo, Paturi, Zane '98)

k-SAT could be solved in $O(2^{(1-\delta)n})$ time for some $\delta > 0$ indep of k

$\Leftrightarrow$ "f-sparse k-SAT" could be solved in $O(2^{(1-\delta')n})$ time for some $\delta' > 0$ indep of k, f

every var occurs in $\le f$ clauses

Pf: omitted...

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

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

... SETH.

Thm (Abboud, Bringmann, Hermelin, ...

Assuming SETH,
no alg'm for subset sum for n integers & target T
in $O(T^{1-\delta} \cdot \underline{2^{o(n)}})$ 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 $F = \overbrace{(x_1 \vee \bar{x}_2)}^{C_1} \wedge \overbrace{(\bar{x}_1 \vee x_3)}^{C_2}$

$k=2$
$f=2$

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 of its $\le k$ vars that satisfies C_j, ($\le 2^k - 1$)

Create number $z(C_j, \varphi) = \overbrace{[\quad\quad\quad]}^{n}\ \overbrace{[0010\ldots0]}^{m}\ \overbrace{[0\cdots0]}^{n}$

Create number $z(C_j, \varphi) = \boxed{}\ \boxed{0010\cdots0}\ \boxed{0\cdots\cdots0}$

i^{th} digit is
$$\begin{cases} 1 & \text{if } x_i = 1 \text{ in } \varphi \\ 0 & \text{if } x_i = 0 \text{ in } \varphi \\ 0 & \text{if } x_i \text{ not in } C_j \end{cases}$$

j^{th} digit is 1

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

Create number $z(x_i, \alpha) = \boxed{}\ \boxed{0\cdots0}\ \boxed{0\cdots010\cdots}$

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 $T = \boxed{ff\cdots f}\ \boxed{1\,1\,1\,1}\ \boxed{1\,1\,1\,1\,1}$

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 $\rightarrow$ super-vars

& $\frac{m}{B}$ groups of B clauses $\rightarrow$ super-clauses

will need "average-free set" ... TO BE CON'TD ...