3SUM - Hardness

Reminder: 3SUM $\quad A \subseteq \mathbb{Z} \quad\quad a, b, c \in A$

Given $A, B, C \subseteq \mathbb{Z}$, are there $a \in A, b \in B, c \in C$ s.t.

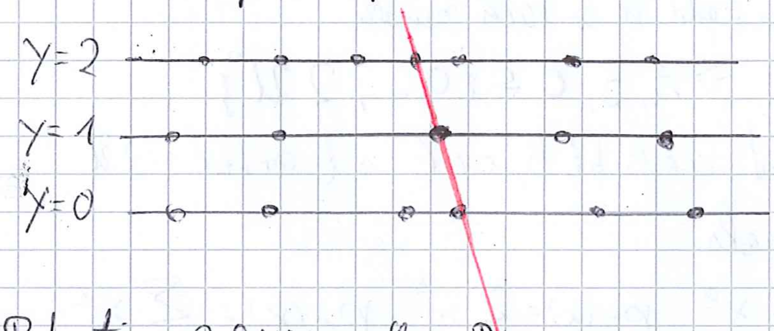
$$a + b + c = 0$$

Hardness of Geometric Problems

Def: (GeomBase)

Given: set of n points on three horizontal lines $y=0$, $y=1$, $y=2$

Task: Decide if there exists a non-horizontal line containing three of the points



Reduction: 3SUM $\Rightarrow$ GeomBase

Given instance (A, B, C) of 3SUM, construct points

$(a, 0)$ for any $a \in A$

$(b, 2)$ for any $b \in B$

$(c/2, 1)$ for any $c \in C$

3 points on a line if $c/2 - a = b - c/2 \Leftrightarrow a + b = c$

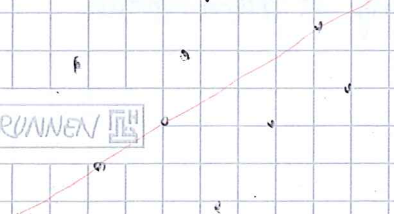
$\Rightarrow T(n)$-alg. for GeomBase implies $O(T(n))$-alg. for 3SUM

(Actually: equivalent) $\quad$ GeomBase is "3SUM-hard"

Def: (Collinear / 3-Points-on-Line)

Given: Set on n points in the plane

Task: Decide if there is a line containing at least 3 of the points



Reduction 3SUM → Collinear

Given instance A of 3SUM
construct points (a, a^3) for any $a \in A$

Observation: (a,a^3) (b,b^3) (c,c^3) collinear if and only if
$$a+b+c=0$$

3SUM for small universe

Theorem: Given 3SUM instance $A, B, C \subseteq \{-U, \dots, U\} \cap \mathbb{Z}$
We can decide if there is a 3SUM triple in time $O(n + U \operatorname{polylog} U)$

Proof: Preprocessing step: add U to each number
$\rightarrow A, B, C \subseteq \{0, \dots, 2U\}$

Goal: find $a \in A, b \in B, c \in C$ s.t. $a+b+c = 3U$

Define polynomials
$$p_A(x) = \sum_{a \in A} x^a \quad p_B(x) = \sum_{b \in B} x^b \quad p_C(x) = \sum_{c \in C} x^c$$
of degree at most $2U$

Compute $q(x) = p_A(x) \cdot p_B(x) \cdot p_C(x)$
(where $x^a \cdot x^b \cdot x^c = x^{a+b+c}$)

Look at coefficient of x^{3U} in $q(x)$
This coefficient gives the number of triples summing to $3U$

Multiplication of polynomials of degree d:
Time $O(d \operatorname{polylog} d)$ (via Fast Fourier Transform) $\square$

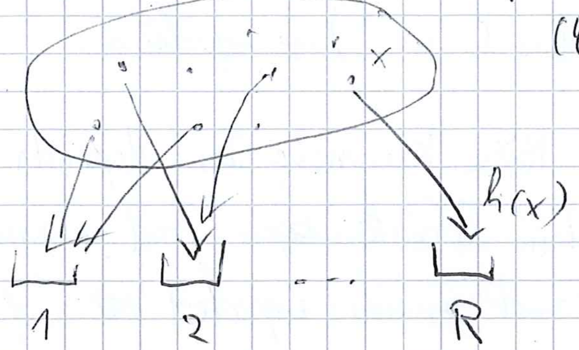
We will show: 3SUM on universe size $U = O(n^3)$ is as hard as in general

"Hashing down" the ~~uni~~ universe using a randomized reduction

Hash function

$h: [\mathcal{U}] \to [R]$

Usually: Family of hash functions (Chose one ~~one~~ from family uniformly at random)



Desired properties:

① Uniform difference:

$Pr[h(x) - h(y) = i] = 1/R$

for any $x, y \in [\mathcal{U}]$ s.t. $x \neq y$ and any $i \in [R]$

② Balanced:

$|\{x \in S : h(x) = i\}| \leq 3 \frac{n}{R}$

for any set $S = \{x_1, \dots, x_n\} \subseteq [\mathcal{U}]$ and any $i \in [R]$

③ ~~A~~ Linear:

$h(x) + h(y) = h(x+y) \bmod R$

for any $x, y \in \mathcal{U}$

Obtained properties:

①' Uniform difference

②' Almost balanced

$i \in R$ is heavy if

$|\{x \in S : h(x) = i\}| > 3 \frac{n}{R}$

Expected # element from S hashed to heavy values is $O(R)$ for any set $S = \{x_1, \dots, x_n\} \subseteq [\mathcal{U}]$ ~~and any $i \in [R]$~~

③' Almost linear:

$h(x) + h(y) \in h(x+y) + c_h + \{0,1\} \bmod R$

for any $x, y \in \mathcal{U}$ and some integer c_h depending only on h

Definition of hash function:

Let $r = k\,R$ for some $k \geq \frac{\mathcal{U}}{2}$, and $\mathcal{U}, R, r$ powers of 2

$\mathcal{H}_{\mathcal{U},R,r} = \{h_{a,b} : [\mathcal{U}] \to [R] \;\; a \in [r] \text{ odd integer and } b \in [r]\}$

with $h_{a,b}(x) := (ax + b \bmod r) \operatorname{div} (r/R)$

integer division

Thm [Dietzfelbinger '96, Baran et al. '08]:

$\mathcal{H}_{\mathcal{U},R,r}$ has properties unif. diff., almost balanced, almost linear with $c_{h_{a,b}} = (b-1 \bmod r) \operatorname{div} (r/R)$

[Pătraşcu et al. '08]

<u>Lemma</u>: If 3SUM on universe of size $O(n^3)$ is solvable in ~~expected~~ time $O(n^{2-\varepsilon})$, then 3SUM on arbitrary universe is solvable in time $O(n^{2-\varepsilon})$ in expectation.

Expected running time: ~~Random~~ Running Time depends on random choices of the algorithm. ~~Consider expec~~ Treat running time as random variable X and consider expected value of X under the random choices

<u>Proof</u>: W.l.o.g consider 3SUM variant $A, B, C \subseteq [U]$ and we want to find $a \in A, b \in B, c \in C$ s.t. $a + b = c$

Algorithm:

Repeat until output performed:

- Pick hash function $h: [1, \dots, U] \to [1 \dots 6n^3]$ uniformly at random from family $\mathcal{H}_{U,R,r}$
- Construct sets $A' = \{h(a) \mid a \in A\}$, $B' = \{h(b) \mid b \in B\}$, $C' = \{h(c) + d_h \mid c \in C\}$
  $A'' = \{h(a) \mid a \in A\}$, $B'' = \{h(b) \mid b \in B\}$, $C'' = \{h(c) + d_h + 1 \mid c \in C\}$
- Solve 3SUM instances (A', B', C') and (A'', B'', C'') using algorithm $\mathcal{A}$ for universe size $\leq 6n^3$
- If $\mathcal{A}$ reports no 3SUM witness: output 'no 3SUM'
- Consider first reported 3SUM witness x', y', z' for (A', B', C')
  If $h^{-1}(x'), h^{-1}(y'), h^{-1}(z' - d_h)$ contains 3SUM witness $\cancel{x, y, z}$ a, b, c for (A, B, C): output $\cancel{x, y, z}$ a, b, c

  set of elements hashed to x
- Consider first reported 3SUM witness x'', y'', z'' for (A'', B'', C'')
  If $h^{-1}(x''), h^{-1}(y''), h^{-1}(z'' - d_h - 1)$ contains witness $\cancel{x, y, z}$ a, b, c: output $\cancel{x, y, z}$ a, b, c

Correctness

- All triples output by algorithm are valid 3SUM witnesses
- The algorithm does not produce "false negative" output

  If $a+b=c$, then $h(x)+h(y) \in \underbrace{h(z)+d_h}+\{0,1\}$ (almost linear)

  $\quad\quad\quad\quad\quad\quad\;\; \cap \quad\quad\quad\;\; \cap$

  $\quad\quad\quad\quad A' \text{ and } A'' \quad B' \text{ and } B'' \quad \in C' \text{ or } C''$

  $\Rightarrow$ Either (A', B', C') or (A'', B'', C'') contains a 3SUM witness and thus the algorithm does not output "no 3SUM"
- Remains to show that alg. terminates

  (in particular: that expected running time is finite

__Claim:__ The algorithm performs a constant number of iterations in expectation

__Proof:__ A false ~~negative~~ positive for hash function h is a triple $a \in A, b \in B, c \in C$

s.t. $a+b \neq c$ and $h(a)+h(b)=h(c)+d_h$ or $h(a)+h(b)=h(c)+d_h+1$

If there is no false ~~negative~~ positive, then algorithm stops certainly:

~~If witness $a+b=c$ exists, then (A', B', C') or (A'', B'', C'') contains a witness~~

If x', y', z' reported for (A', B', C'), then

$x'=h(a)$, $y'=h(b)$, $z'=h(c)+d_h$ for some a,b,c and $x'+y'=z'$

$\Rightarrow a \in h^{-1}(x')$, $b \in h^{-1}(y')$ and $c \in h^{-1}(z'-d_h)$

$\Rightarrow$ witness a,b,c reported. as no false positives exist, a,b,c is correct witness

If x'', y'', z'' reported for (A'', B'', C''): similar argument

We now bound $\Pr[\exists \text{ false positive for } h] \leq \frac{1}{2}$

Fix a,b,c s.t. $a+b \neq c$

By linearity $h(a)+h(b) = h(a+b)+d_h$ or

$h(a)+h(b) = h(a+b)+d_h+1$

$\Pr[a, b, c \text{ false positive}]$

$= \Pr[h(a)+h(b) = h(c)+d_h \text{ or } h(a)+h(b) = h(c)+d_h+1]$

$\leq \Pr[h(a+b) - h(c) \in \{-1, 0, +1\}]$

$\leq 3 \cdot \frac{1}{6n^3} = \frac{1}{2n^3}$

uniform difference: $\Pr[h(a+b) - h(c) = 0] = \frac{1}{6n^3}$

union bound: $\Pr[E_1 \cup E_2 \cup E_3] \leq \Pr[E_1] + \Pr[E_2] + \Pr[E_3]$

There are at most n^3 triples a, b, c

$\Rightarrow$ overall prob. of false positive $\leq n^3 \cdot \frac{1}{2n^3} = \frac{1}{2}$

$\Rightarrow$ In expectation: 2 iteration until ~~no~~ false positive

("Waiting Time Bound": success prob $p \Rightarrow$ expected number of trials until first success $= \frac{1}{p}$) $\square$

$\Rightarrow$ # calls to 3SUM alg. $\mathcal{A}$ ~~on~~ ≤ 2 in expectation

<u>Claim:</u> In each iteration, the algorithm checks a constant number of candidate witnesses

<u>Proof:</u> Fix 3SUM witness x', y', z' of instance (A', B', C')

(similar argument for (A'', B'', C''))

Let $a^* \in h^{-1}(x')$ (must exist)

For every $a \neq a^*$: $\Pr[h(a) = h(a^*)] = \frac{1}{6n^3}$ (uniform diff.)

$\Rightarrow E[|h^{-1}(x')|] = \sum_{a \in A} \Pr[a \in h^{-1}(x')]$ (linearity of exp.)

$= \sum_{a \in A} \Pr[h(a) = h(a^*)] \leq 1 + \frac{n}{4n^3} \leq 2$

Similarly: $E[|h^{-1}(y')|] \leq 2$ and $E[|h^{-1}(z')|] \leq 2$ $\square$

Thus we showed: 3SUM ~~on~~ on arbitrary universe size can be solved in time ~~O(T...)~~ $O(T(n))$, where $T(n)$ is running time for universe size $O(n^3)$