of array of size T

⇒ can compute $C^{(j)}$ from $C^{(j/2)}$ in $O(T\log T)$ time

⇒ total time $\boxed{O(T\log^2 T)}$

$= \boxed{\tilde{O}(T)}$

But doesn't work for original problem...

Koiliaris & Xu '17: $\tilde{O}(T\sqrt{n})$ deterministic

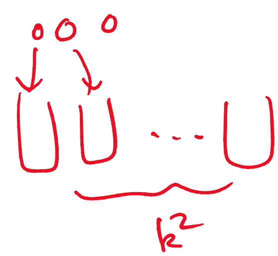
Bringmann '17: $\tilde{O}(T)$ randomized.

Bringmann's Alg'm:

Lemma Suppose $S \subseteq [U]$ & sol'n uses $\le k$ elems.
Then there is rand. alg'm with $\tilde{O}(k^2 U)$ time.

Pf:

Fact Put k balls in k^2 bins randomly.
With prob $\geq \frac{1}{2}$, every bin contains ≤ 1 ball.



Pf: $\Pr(\exists \text{ 2 balls in same bin})$

$$\le \binom{k}{2}\cdot\frac{1}{k^2} = \frac{1}{2}. \quad \square$$

idea - Partition S into k^2 subsets $S_1,\dots,S_{k^2}$ randomly

Given $S_1,\dots,S_\ell$,

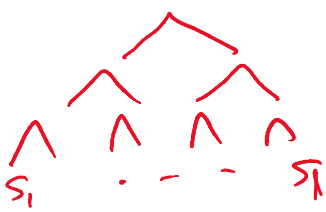
define $C_{\ldots}[i]$ = true iff

define $C_{S_1,\dots,S_\ell}[i]$ = true iff $\exists$ subset with ≤ 1 elem from each of $S_1,\dots,S_\ell$ summing to i $\quad (i = 0,\dots,\ell U)$

Then

$$C_{S_1\dots S_\ell}[i] = \bigvee_{i'} \left(C_{S_1,\dots,S_{\ell/2}}[i'] \wedge C_{S_{\ell/2+1},\dots,S_\ell}[i-i'] \right)$$

Convolution of array of size $O(\ell U)$.



(Monte Carlo err prob $\le \frac{1}{2}$
can be lowered by repeating $\log n$ times
$\rightarrow$ err prob $\le \frac{1}{n}$).

$$\Rightarrow \quad T(\ell) = 2T(\ell/2) + O(\ell U \log(\ell U))$$

$$\Rightarrow \quad T(\ell) = O((\ell \log \ell) \cdot U \log(\ell U)).$$

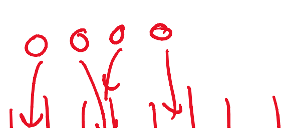
$$= \tilde{O}(\ell U).$$

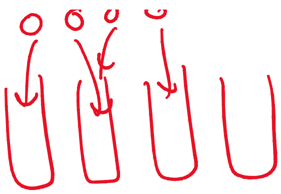
Plug in $\ell = k^2 \Rightarrow \tilde{O}(k^2 U)$.

$\square$

Lemma 2 In Lem 1, time can be improved to $\tilde{O}(kU)$.

Pf: **Fact** Put k balls in k bins randomly.
With prob $\ge 1 - O(\frac{1}{n})$,
every bin contains $\le \log n$ balls. $\quad [O(\frac{\log n}{\log\log n})]$





every bin contains $\lesssim \log n$ balls.

Pf: By Chernoff bound. $\square$

Partition S into $S_1, \ldots, S_k$ randomly,

Define $C_{S_1 \ldots S_\ell}[i] =$ true iff $\exists$ subset with $\lesssim \log n$ elems from each of $S_1 \ldots S_\ell$ summing to i

$(i = 0, \ldots, \ell U \log n)$

Same formula

$$T(\ell) = 2T(\ell/2) + O(\ell U \log n \log(\ell U))$$

base case

$$T(1) = \tilde{O}(U) \text{ by Lem 1 (with } k = \log n).$$

$$\Rightarrow \quad T(\ell) = \tilde{O}(\ell U)$$ like before but with more logs

plug in $\ell = k$: $\Rightarrow \tilde{O}(kU)$. $\square$

Overall Algm:

for each $u = 1, 2, 4, \ldots U$ do

apply Lem 2 to $\{a_i \in S : a_i \in [u, 2u)\}$ with $k = \frac{T}{u}$

$$\Rightarrow \text{time } \tilde{O}\left(\frac{T}{u} \cdot 2u\right) = \tilde{O}(T)$$

Combine by $O(\log U)$ convolutions

$\Rightarrow$ time $O(\frac{T}{u} \cdot 2u) = O(T)$

Combine by $O(\log U)$ convolutions

$\Rightarrow$ time $O((\log U) \cdot T \log T) = \tilde{O}(T)$

$\Rightarrow$ total time $\boxed{\tilde{O}(T).}$.

Jin & Wu's Alg'm ('19) (Sketch)

idea: polynomials

Suffice to compute $\prod_{a \in S} (1 + x^a) \mod x^{T+1}$

& check coeff of x^T

e.g. $(1+x^3)(1+x^5)(1+x^{11})$

$S = \{3, 5, 11\}$

How? $\exp\left(\sum_{a \in S} \ln(1 + x^a) \right) \mod x^{T+1}$

use formal power series!

$$\ln(1+x) = \sum_{i=1}^{\infty} \frac{(-1)^i}{i} x^i$$

$$\ln(1+x^a) = \sum_{i=1}^{\lfloor T/a \rfloor} \frac{(-1)^i}{i} x^{ai} \mod x^{T+1}$$

$\nearrow$ $O(T/a)$ terms

total time $O\left(T \sum_{a \in S} \frac{1}{a}\right)$

$= O(T \log T)$.

polynomial exp. known to be reducible to polynomial multiplication i.e. convolution

need to work in finite field $\mathbb{Z}_p$

pick random p

$\Rightarrow$ $\boxed{O(T \log^2 T)}$ rand. time.

Open: deterministic?

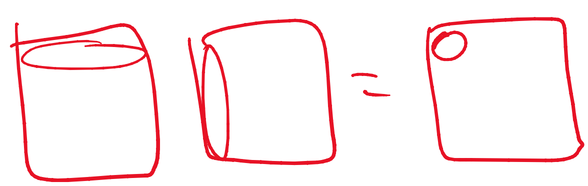
Cond. Lower Bds? later...

Matrix Multiplication

Problem Given $n \times n$ matrices $A = (a_{ij})$, $B = (b_{ij})$

compute $C = AB$

where $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$



trivial: $\boxed{O(n^3)}$ time

naive lower bd: $\Omega(n^2)$.

Strassen's Alg'm ('69)

Warm-up: $n=2$

to compute

$c_{11} = a_{11} b_{11} + a_{12} b_{21}$

$c_{21} = a_{21} b_{11} + a_{22} b_{21}$

$c_{12} = a_{11} b_{12} + a_{12} b_{22}$

$c_{22} = a_{21} b_{12} + a_{22} b_{22}$

naively: 8 mults.

HW1 available