of array of size $T$

$\Rightarrow$ Can compute $C^{(i)}$ from $C^{(U^2)}$ in $O(T \log T)$ time

$\Rightarrow$ Total time $O(T \log^2 T)$

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

doesn't work for
But original problem...

Koiliaris & Xu '17: $\tilde{O}(TVn)$ deterministic

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

Bringmann's Alg'm:

**Lemma:** Suppose $S \subseteq [U]$ & soln uses $\leq k$ 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 $\leq 1$ ball.

**Pf:**

$$\Pr \left( \exists \text{ 2 balls in same bin} \right)$$

$$\leq \binom{k}{2} \cdot \frac{1}{k^2} = \frac{1}{2}.$$

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

Given $S_1, \ldots, S_{k^2}$,

Define $C_{i,j} \iff i \in S_j$
define \( C_{s_1\ldots s_k}[i] = \text{true iff} \)
\[ \exists \text{ subset with } \leq 1 \text{ elem from each of } s_1, \ldots, s_k \]
summing to \( i \) \((i = 0, \ldots, 2^{LU})\)

Then
\[ C_{s_1\ldots s_k}[i] = \bigvee_{i'} \left( C_{s_1\ldots s_{k-2}}[i'] \land C_{s_{k-1}\ldots s_k}[i-i'] \right) \]

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

(Monte Carlo rev prob \( \leq \frac{1}{2} \)
Can be lowered by repeating \( \log \text{times} \)
\( \rightarrow \) error prob \( \leq \frac{1}{n} \)).

\[ T(k) = 2T(k/2) + O(U \log(kU)) \]
\[ \Rightarrow T(k) = O(k \log k) \cdot U \log(kU)) \]
\[ = \tilde{O}(kU). \]

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

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

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

\[ \text{Pf: By Chernoff bound.} \]

Partition \( S \) into \( S_1, \ldots, S_k \) randomly,

Define \( C_{s_1, \ldots, s_n}(i) = \text{true} \) iff \( \exists \) subset with \( \leq \log n \) elements

summing to \( i \)

\[ (i = 0, \ldots, kU \log n) \]

Same formula

\[ T(l) = 2T(l/2) + O(kU \log n \log(n)) \]

base case

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

\( \Rightarrow \)

\[ T(l) = \tilde{O}(kU) \text{ like before but with more logs} \]

Plug in \( k = k^* \):

\[ \Rightarrow \tilde{O}(kU). \]

Overall Alg:

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

apply Lem 2 to \( \{ a \in S : a \in [u, 2u) \} \)

with \( k = \frac{1}{u} \)

\[ \Rightarrow \text{time } \tilde{O}(\frac{1}{u} \cdot 2u) = \tilde{O}(T) \]

Combine by \( \tilde{O}(\log(n)) \) convolutions
Combine by $O(\log U)$ convolutions

$\Rightarrow$ time $O((\log U) \cdot T \log T)$

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

---

Jin & Wu's Algm ('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. $(1 + x^3)(1 + x^5)(1 + x^7)$

$S = \{3, 5, 7\}$

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

Use formal power series!

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

$\ln (1 + x^a) = \sum_{i=1}^{\infty} \frac{(-1)^i}{i} x^{ai} \mod x^{T+1}$
\[ \text{total time} \quad O\left( T \sum_{a \in S} \frac{1}{a} \right) \]

\[ = O(T \log T) . \]

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

Need to work in finite field \( \mathbb{Z}_p \).

Pick random \( p \)

\[ \Rightarrow O(T \log^2 T) \text{ rand. time.} \]

Open: deterministic?

Cont. Lower Bds? Later...

Matrix Multiplication

**Problem**

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

Compute \( C = AB \)

Where \( c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \)

\[ \text{[Diagram of matrix multiplication]} \]
trivial: $O(n^3)$ time

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

**Strassen's Alg'm ('69)**

**Warm-up: $n=2$**

- to compute
  
  \[ c_{11} = a_{11} b_{11} + a_{12} b_{21}, \]
  
  \[ c_{21} = a_{21} b_{11} + a_{22} b_{21}. \]

  naively: 8 mults.

\[ \text{HW1 available} \]