L8 10.4.17

First Topic Lower bounds

Dynamic Partial Sums

Model : Cell Probe Model

- computation for free
- pay for memory access

This is stronger than RAM / pointer machine

Goal

- Maintain array $A[0, \dots, n-1]$
- Update(i,x) $A[i] = x$
- Query(i) $\sum_{k=0}^{i} A[k]$

Arbitrary group op fine

use a tree $\rightarrow O(\log n)$

Lower bound : $\Omega(\log n)$ / OP worst case & amortized | det/rand

either update/query

Assume $A = [0, \dots, 0]$, $|A| = n$ $\quad n = 2^k$ for $k \in \mathbb{N}$

neut. Elm in group G

$\leadsto$ Access pattern:

for $t = 0$ to $n-1$ do

$\quad i = \text{reverse\_bit}(t)$

$\quad \text{Query}(i)$

$\quad \text{Update}(i, x_i)$

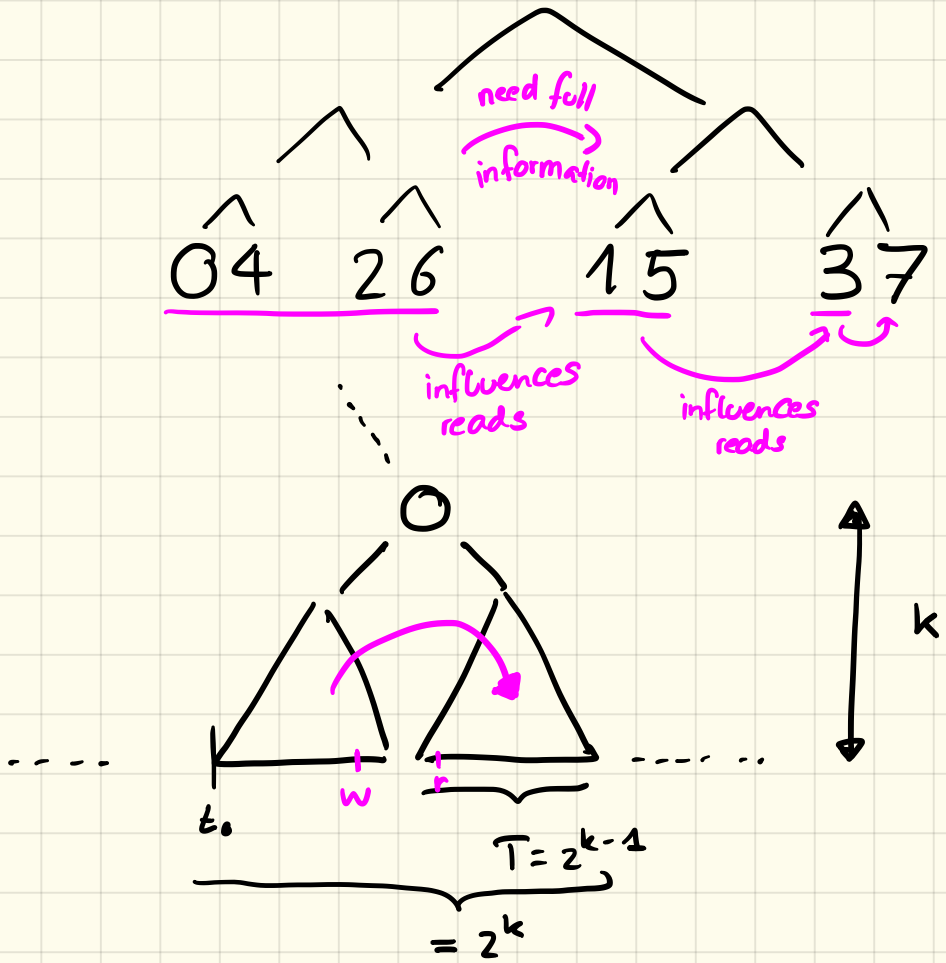
end for

reverse_bit:

0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

x_i: independent uniform random from G

Access sequence



We want to measure information transfer from left to right

Information Transfer:

<u>Lemma</u> $\Omega(T)$ memory cells are written on the left and then read on the right

Lemma implies $\Omega(n \log n)$ work

$\rightarrow \Omega(\log n)/OP$

actually expected values....

proof of Lemma:

- Each x_i has $\Theta(\log n)$ bits of entropy
- R - memory cells read on the right
- W - memory cells written on the left

Given state of DS. at t_0 & queries on the right → defines updates on the left

E.g. Query(1): X_0

Query (5): $\cancel{X_0} + \cancel{X_2} + X_4 + \cancel{X_1}$

Query (3): $\cancel{X_0} + X_2 + \cancel{X_1}$

Query (7): $\cancel{X_0} + \cancel{X_2} + \cancel{X_4} + X_6 + \cancel{X_1 + X_3 + X_5}$

known by right - queries

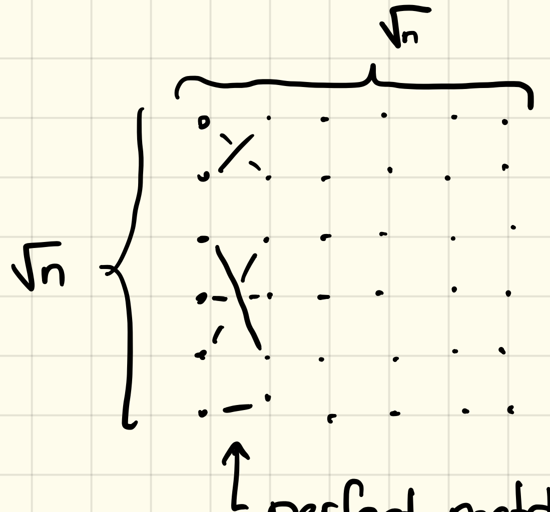
- $|R \cap W| \cdot \log n$ ← explicitly encode

⇝ We extracted $\Omega(T \log n)$ bits of entropy using $|R \cap W| \cdot \log n$ bits

$$\Rightarrow \Omega(T \log n) \leq |R \cap W| \log n$$

Dynamic Connectivity Lower Bound

<u>Claim</u>: $\Omega(\log n)$/OP (update or query)



perfect matching on each layer $\rightarrow \pi_i \in S_{\sqrt{n}}$

$\rightarrow$ Can be seen as Composition of Permutations

$\exists$ path $V_{1,i} \rightarrow V_{k,j}$ iff

$\pi_k(\pi_{k-1}(\cdots \pi_2(\pi_1(i))\cdots)) = j$

Each π_i requires $\sqrt{n} \log n$ bits !

<u>Idea</u>: Maintain $\pi_1, \ldots, \pi_{\sqrt{n}}$

Query $\pi_1 \circ \cdots \circ \pi_i$

Update π_i

Problem: Extracting $\pi_1 \circ \dots \pi_i$ using graph connectivity queries is infeasible

we can only test : x, y connected?

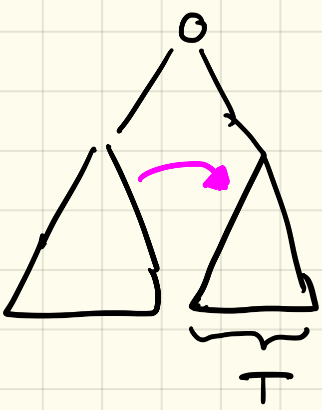
→ Maintain: $\pi_1, \dots, \pi_{\sqrt{n}}$

Query(i, x) verify $\pi_1 \circ \dots \circ \pi_i = x$

we will verify given $\pi_1 \circ \dots \circ \pi_i$

Update(i, x) $\pi_i \leftarrow x$

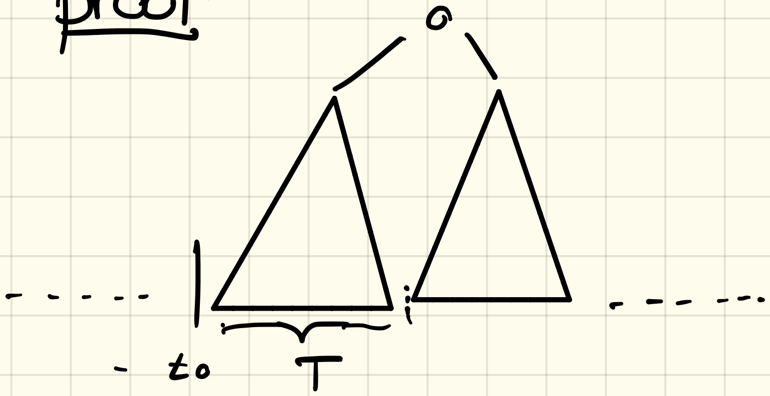
Lemma: $\Omega(T \cdot \sqrt{n} \cdot \log n)$ bits of information transfer from left to right.

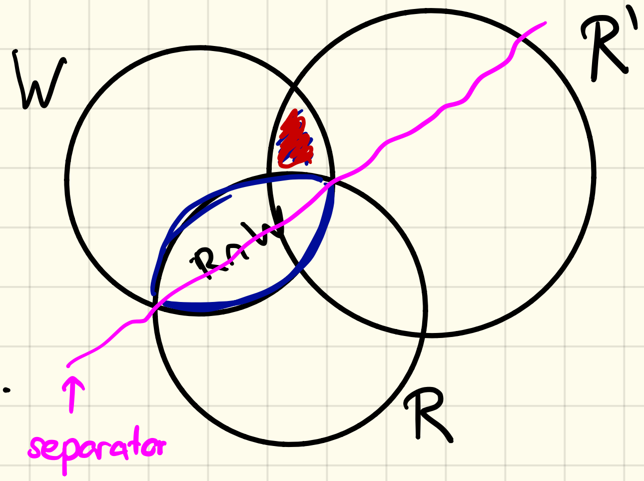


→ $\Omega(T \cdot \sqrt{n})$ mem. cells accessed

→ In total $\Omega(\sqrt{n} \cdot \sqrt{n} \cdot \log n)$.

__proof__





- $\Omega(T \cdot \sqrt{n} \cdot \log n)$ bits of entropy
- Simulate Query(i, x) for __ALL__ $x \in S_{\sqrt{n}}$
- Trick: Simulator gets separator S on $R \setminus W$ & $W \setminus R$
  - written in right subtree - easy
  - $R \cap W$: explicitly
  - in S $\rightarrow$ known, past
  - not in S $\rightarrow$ abort

should be small!

__Thm__ size m sets family $\mathcal{S} \subset 2^U$

$\forall A, B \subseteq U \quad |A|, |B| \leq m$

$\exists C \in \mathcal{S} \quad A \subseteq C, \; B \subseteq U \setminus C$

$\exists\ S \qquad |S| \leq 2^{O(m + \log\log U)}$

see exercise class