

Lower bounds for succinct data structures

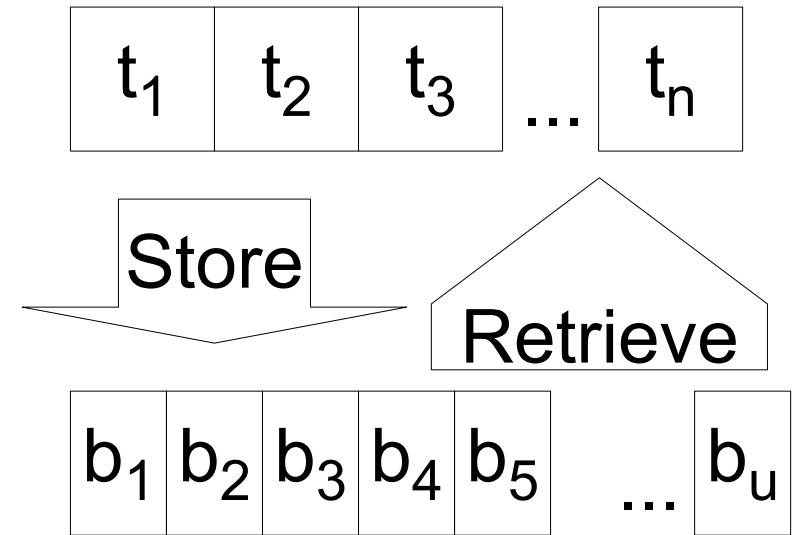
Emanuele Viola

Northeastern University

December 2009

Bits vs. trits

- Store n “trits” $t_1, t_2, \dots, t_n \in \{0,1,2\}$



In u bits $b_1, b_2, \dots, b_u \in \{0,1\}$

- Want:

Small space u (optimal = $\lceil n \lg_2 3 \rceil$)

Fast retrieval: Get t_i by probing few bits (optimal = 2)

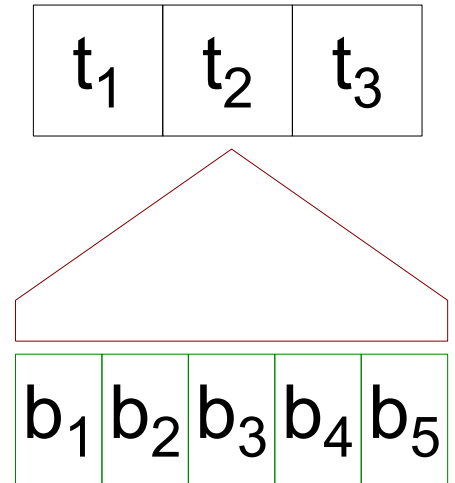
Two solutions

- Arithmetic coding:

Store bits of $(t_1, \dots, t_n) \in \{0, 1, \dots, 3^n - 1\}$

Optimal space: $\lceil n \lg_2 3 \rceil \approx n \cdot 1.584$

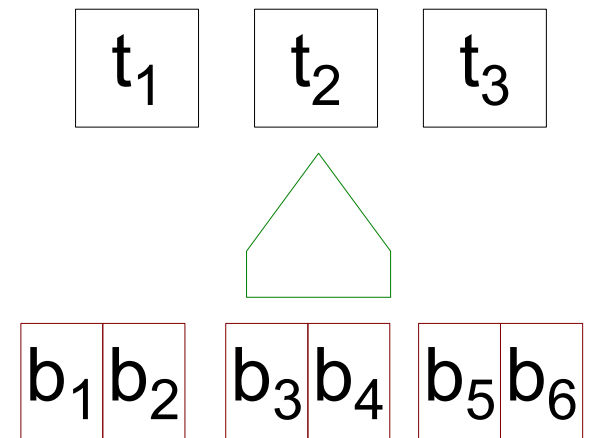
Bad retrieval: To get t_i probe all $> n$ bits



- Two bits per trit

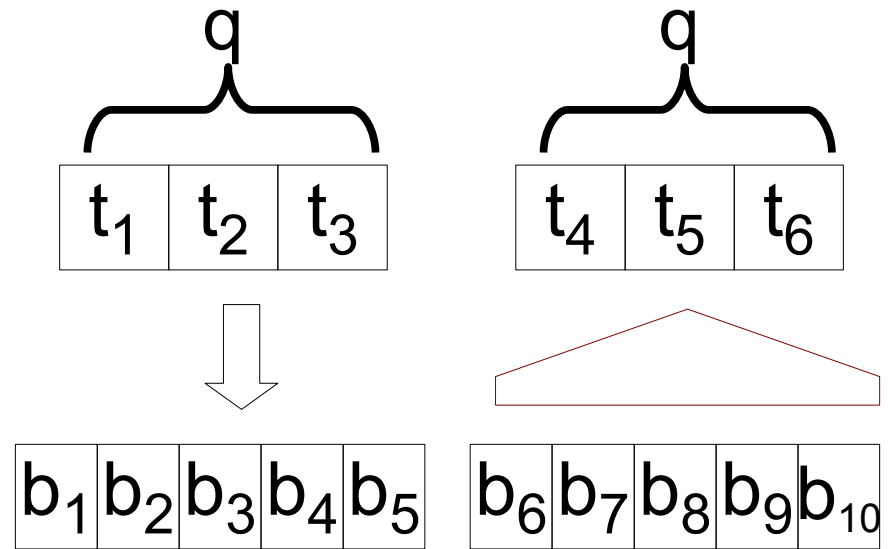
Bad space: $n \cdot 2$

Optimal retrieval: Probe 2 bits



Polynomial tradeoff

- Divide n trits $t_1, \dots, t_n \in \{0, 1, 2\}$ in blocks of q
- Arithmetic-code each block



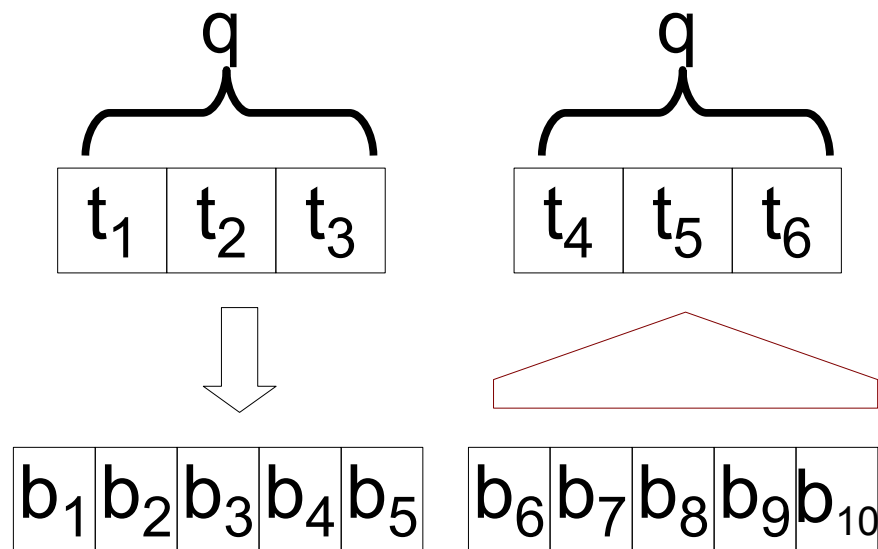
$$\begin{aligned} \text{Space: } \lceil q \lg_2 3 \rceil n/q &< (q \lg_2 3 + 1) n/q \\ &= n \lg_2 3 + n/q \end{aligned}$$

Retrieval: Probe $O(q)$ bits

polynomial
tradeoff
between
redundancy,
probes

Polynomial tradeoff

- Divide n trits $t_1, \dots, t_n \in \{0, 1, 2\}$ in blocks of q
- Arithmetic-code each block



$$\text{Space: } \lceil q \lg_2 3 \rceil n/q = (q \lg_2 3 + 1/q^{\Theta(1)}) n/q$$

$$= n \lg_2 3 + n/q^{\Theta(1)}$$

Retrieval: Probe $O(q)$ bits

polynomial
tradeoff
between
redundancy,
probes

Logarithmic forms

Exponential tradeoff

- Breakthrough [Pătraşcu '08, later + Thorup]

Space: $n \lg_2 3 + n/2^{\Omega(q)}$

Retrieval: Probe q bits

exponential
tradeoff
between
redundancy,
probes

- E.g., optimal space $\lceil n \lg_2 3 \rceil$, probe $O(\lg n)$

Our results

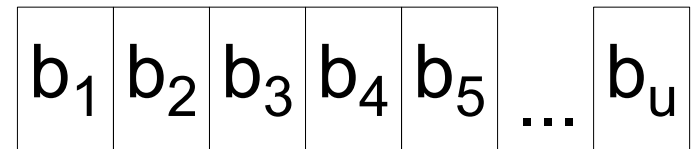
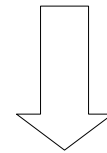
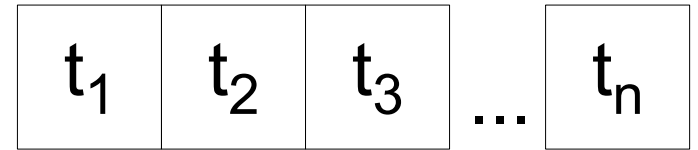
- **Theorem[V.]**:

Store n trits $t_1, \dots, t_n \in \{0,1,2\}$

in u bits $b_1, \dots, b_u \in \{0,1\}$.

If get t_i by probing q bits

then space $u > n \lg_2 3 + n/2^{O(q)}$.



- Matches [Pătrașcu Thorup]: space $< n \lg_2 3 + n/2^{\Omega(q)}$

Outline

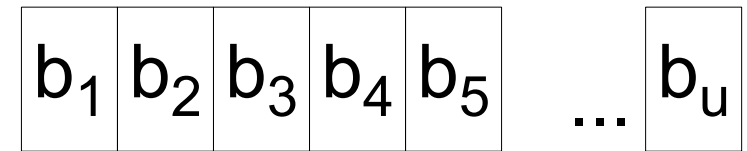
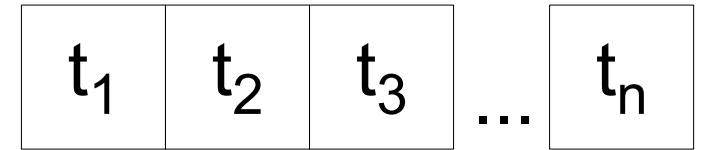
- Bits vs. trits
- Proof bits vs. trits
- Bits vs. sets
- Cells vs. prefix sums

Recall our results

- Theorem:**

Store n trits $t_1, \dots, t_n \in \{0,1,2\}$

in u bits $b_1, \dots, b_u \in \{0,1\}$.



If get t_i by probing q bits

then space $u > n \lg_2 3 + n/2^{O(q)}$.

- For now, assume non-adaptive probes:

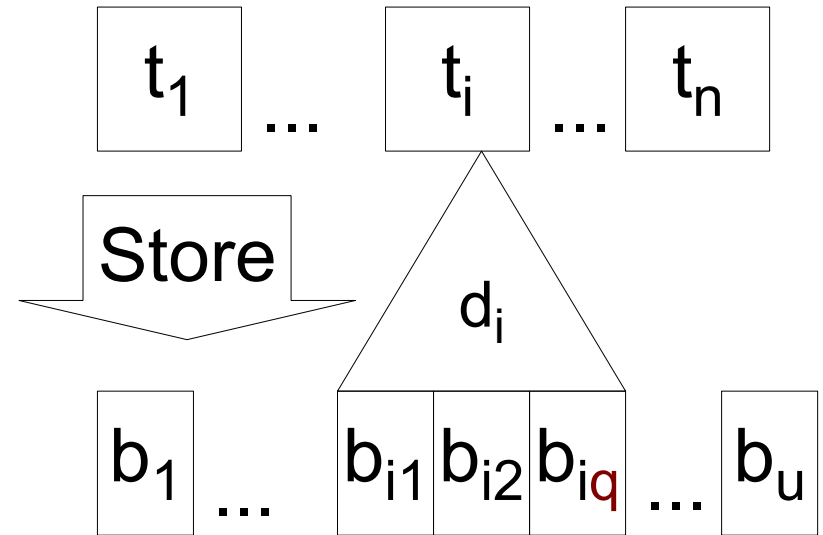
$$t_i = d_i (b_{i1}, b_{i2}, \dots, b_{iq})$$

Proof idea

- $t_i = d_i (b_{i1}, b_{i2}, \dots, b_{iq})$

- Uniform $(t_1, \dots, t_n) \in \{0,1,2\}^n$

Let $(b_1, \dots, b_u) := \text{Store}(t_1, \dots, t_n)$



- Space $u \approx \text{optimal} \Rightarrow (b_1, \dots, b_u) \in \{0,1\}^u \approx \text{uniform} \Rightarrow$

$$1/3 = \Pr [t_i = 2] = \Pr [d_i (b_{i1}, \dots, b_{iq}) = 2] \approx A / 2^q \neq 1/3$$

Contradiction, so space $u \gg \text{optimal}$

Q.e.d.

Information-theory lemma

[Edmonds Rudich Impagliazzo Sgall, Raz, Shaltiel V.]

Lemma: Random (b_1, \dots, b_u) uniform in $\mathbf{B} \subseteq \{0, 1\}^u$

$|\mathbf{B}| \approx 2^u \Rightarrow$ there is large set $\mathbf{G} \subseteq [u]$:

for every $i_1, \dots, i_q \in \mathbf{G} : (b_{i_1}, \dots, b_{i_q}) \approx$ uniform in $\{0, 1\}^q$

Proof: $|\mathbf{B}| \approx 2^u \Rightarrow H(b_1, \dots, b_u)$ large

$\Rightarrow H(b_i | b_1, \dots, b_{i-1})$ large for many i ($\in \mathbf{G}$)

Closeness[$(b_{i_1}, \dots, b_{i_q})$, uniform] $\geq H(b_{i_1}, \dots, b_{i_q})$

$\geq H(b_{i_q} | b_1, \dots, b_{i_q-1}) + \dots + H(b_{i_1} | b_1, \dots, b_{i_1-1})$, large Q.e.d.

Proof

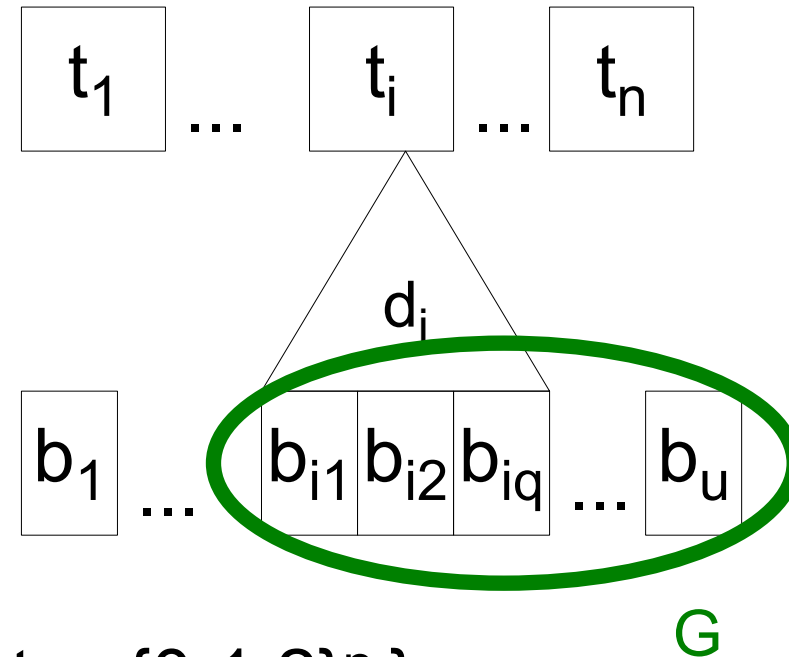
- Argument OK **if** probes in G

- $t_i = d_i(b_{i1}, b_{i2}, \dots, b_{iq})$

- Uniform $(t_1, \dots, t_n) \in \{0, 1, 2\}^n$



uniform $(b_1, \dots, b_u) \in \mathbf{B} := \{\text{Store}(t) \mid t \in \{0, 1, 2\}^n\}$

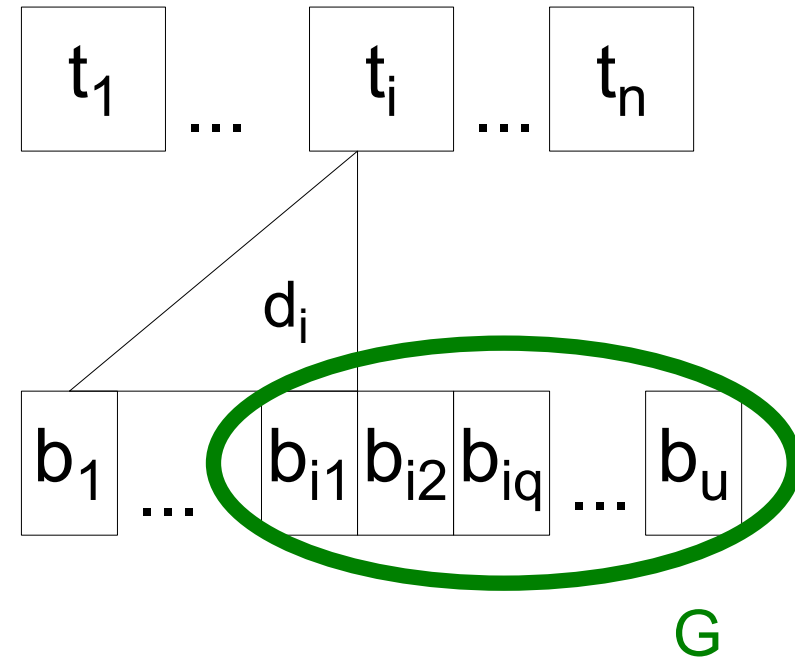


$$|\mathbf{B}| = 3^n \approx 2^u \Rightarrow (\text{Lemma}) \Rightarrow (b_{i1}, \dots, b_{iq}) \approx \text{uniform} \Rightarrow$$

$$1/3 = \Pr [t_i = 2] = \Pr [d_i (b_{i1}, \dots, b_{iq}) = 2] \approx A / 2^q \neq 1/3$$

Probes not in G

- If every t_i probes bits not in G



- Argue as in [Shaltiel V.]:
- Condition on **heavy** bits := probed by many t_i
- Can find $t_i \approx$ uniform in $\{0,1,2\}$, all probes in G

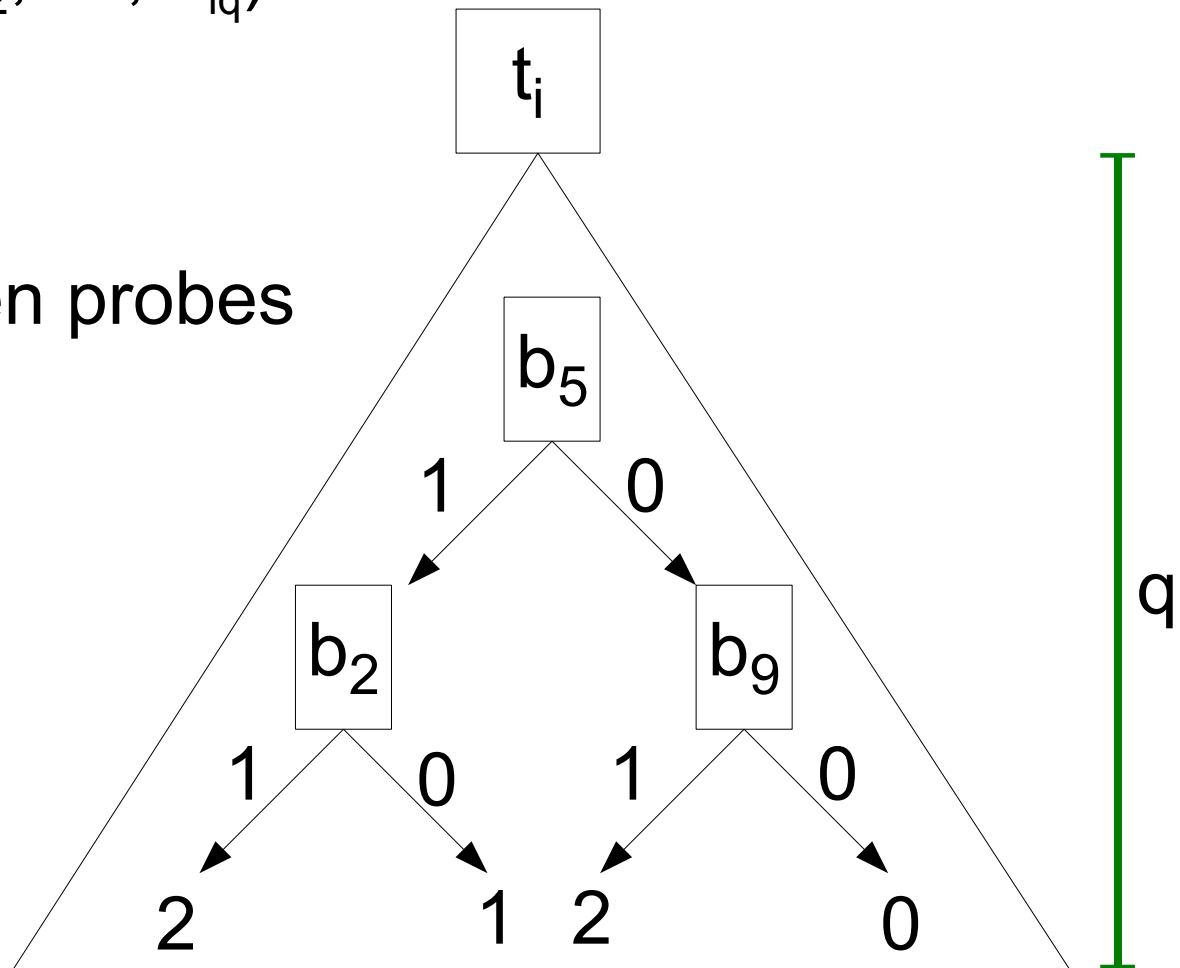
Handling adaptivity

- So far $t_i = d_i (b_{i1}, b_{i2}, \dots, b_{iq})$

- In general,
q **adaptively** chosen probes
= decision tree

2^q bits

depth q



$$1/3 = \Pr [t_i = 2] = \Pr [d_i (b_{i1}, \dots, b_{i2q}) = 2] \approx A / 2^q \neq 1/3$$

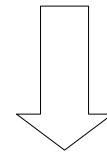
Outline

- Bits vs. trits
- Proof bits vs. trits
- Bits vs. sets
- Cells vs. prefix sums

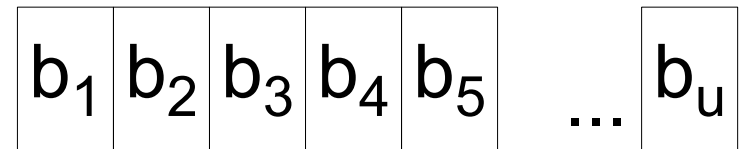
Bits vs. sets

- Store $S \subseteq \{1, 2, \dots, n\}$ of size $|S| = k$

01001001101011



In u bits $b_1, \dots, b_u \in \{0,1\}$



- Want:

Small space u (optimal = $\lceil \lg_2 (n \text{ choose } k) \rceil$)

Answer “ $i \in S$?” by probing few bits (optimal = 1)

Previous results

- Store $S \subseteq \{1, 2, \dots, n\}$, $|S| = k$ in bits, answer “ $i \in S?$ ”
- [Minsky Papert '69] Average-case study
- [Buhrman Miltersen Radhakrishnan Venkatesh; Pagh '00]
Space $O(\text{optimal})$, probe $O(\lg(n/k))$
Lower bounds for $k < n^{1-\epsilon}$
- No lower bound was known for $k = \Omega(n)$

Our results

- **Theorem[V.]**:

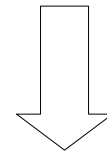
Store $S \subseteq \{1, 2, \dots, n\}$, $|S| = n/3$

in u bits $b_1, \dots, b_u \in \{0,1\}$

If answer “ $i \in S?$ ” probing q bits
then space $u > \text{optimal} + n/2^{O(q)}$.

- First lower bound for $|S| = \Omega(n)$

01001001101011



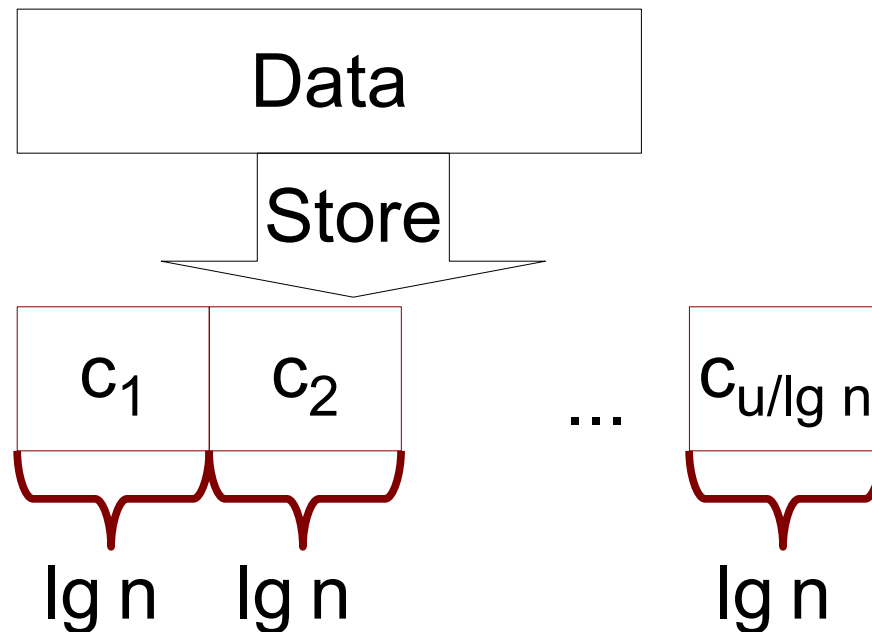
b_1	b_2	b_3	b_4	b_5	...	b_u
-------	-------	-------	-------	-------	-----	-------

Outline

- Bits vs. trits
- Proof bits vs. trits
- Bits vs. sets
- Cells vs. prefix sums

Cell-probe model

- So far: q = number of **bit** probes
- Cell model: q = number of probes in **cells of $\lg(n)$ bits**



- Relationship: $q \text{ bit} \subseteq q \text{ cell} \subseteq q \lg(n) \text{ bit}$

Results in cell-probe model

- **Cells vs. trits:**

$q = O(1)$, optimal space = $\lceil n \lg_2 3 \rceil$ [Pătrașcu Thorup]

$q = 1 \Rightarrow$ space $> n \lg_2 3 + n / \lg^{O(1)} n$ [this work]

- **Cells vs. sets:**

q probes, space = optimal + $n / \lg^{\Omega(q)} n$ [Pagh, Pătrașcu]

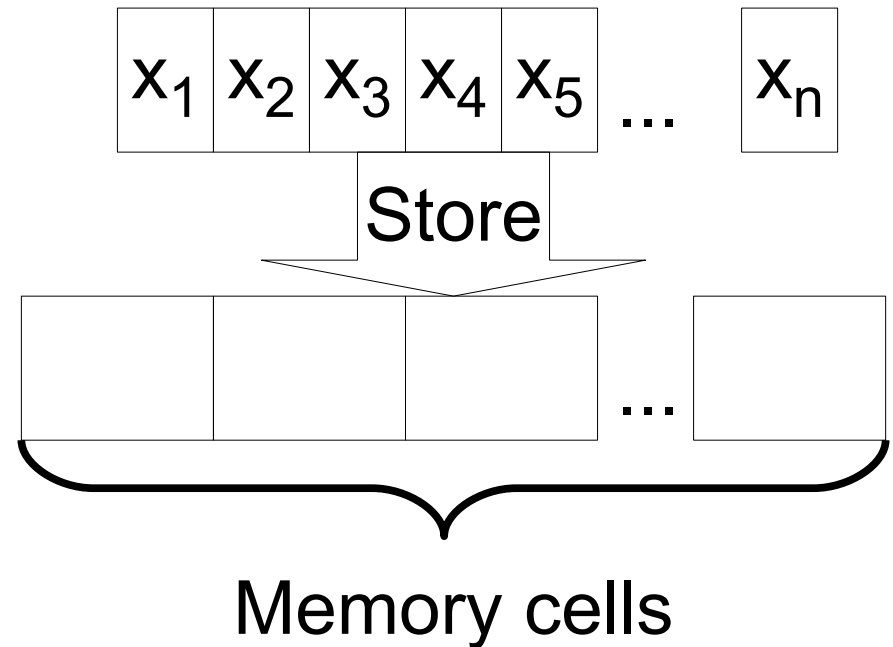
Lower bounds?

Outline

- Bits vs. trits
- Proof bits vs. trits
- Bits vs. sets
- Cells vs. prefix sums

Prefix sums

- Store n bits $x_1, x_2, \dots, x_n \in \{0,1\}$ in memory cells



- Want:

Small space

Fast answer **prefix sum** (a.k.a. Rank) queries:

$$\text{Sum}(i) := \sum_{k \leq i} x_k \in \{0, 1, 2, \dots, n\}$$

History

- Fundamental problem: succinct trees, sets, ...
- Trivial
 - Space = $n \lg n$
 - Time = 1 cell probe
- [Jacobson '89]
 - Space = $n + O(n / \lg n)$
 - Time = $O(1)$ cell probes
- [Pătraşcu '08]
 - Space = $n + n / \lg^q n$
 - Time = $O(q)$ cell probes

Our results

- **Theorem**[Pătrașcu V.]:
Store n bits in memory

If answer $\text{Sum}(i) := \sum_{k \leq i} x_k$ queries

by probing q cells then space $> n + n / \lg^{O(q)} n$.

- Matches [Pătrașcu]: space $< n + n / \lg^{\Omega(q)} n$

Proof idea

- Efficient data structure \Rightarrow Break queries' correlations
- For $i < j$, $A \subseteq \{0,1\}^n$

$$0 = \Pr_{x \in A} [\text{Sum}(i) > t \text{ AND } \text{Sum}(j) < t]$$

$$\approx \Pr_{x \in A} [\text{Sum}(i) > t] \Pr_{x \in A} [\text{Sum}(j) < t]$$

$$> \quad (1/10) \quad (1/10) \quad \gg 0$$

- Contradiction, so data structure cannot be efficient

Proof idea

$$0 = \Pr_{x \in A} [\text{Sum}(i) > t \text{ AND } \text{Sum}(j) < t]$$

$$\approx \Pr_{x \in A} [\text{Sum}(i) > t] \Pr_{x \in A} [\text{Sum}(j) < t] \quad (1)$$

$$> \quad (1/10) \quad (1/10) \quad (2)$$

- Reasoning:

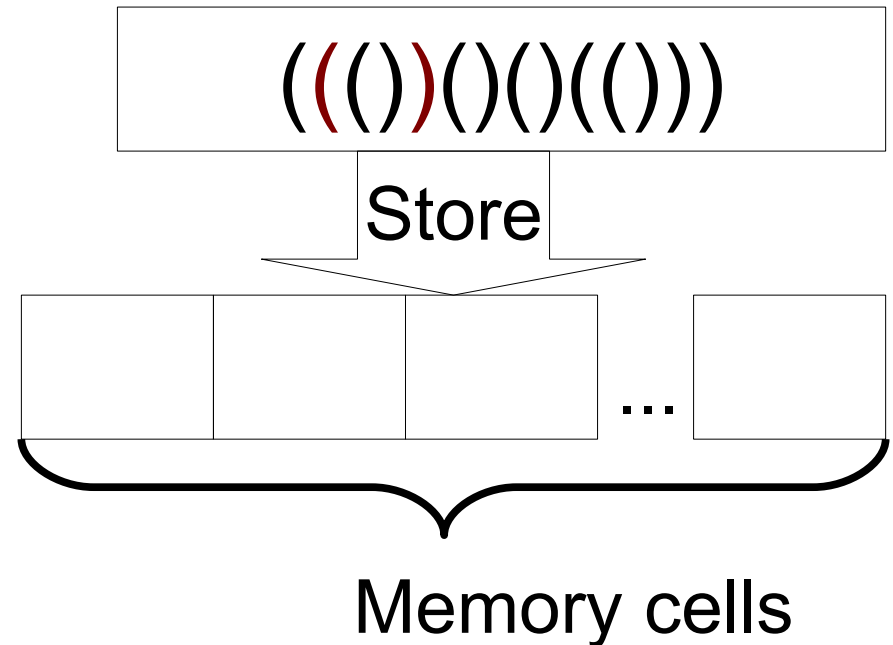
Fix heavy cells. Then $\exists i, j$ s.t. $\text{Sum}(i)$ and $\text{Sum}(j)$:

(1) depend on disjoint, nearly uniform cells \Rightarrow independent

(2) have high entropy

Balanced brackets

- Store n balanced brackets
- Want:
Small space
Fast answer **match** queries:



- **Theorem[V.]**: space $>$ optimal + $n / \lg^{2^{O(q)}} n$.
for **non-adaptive** q probes
- [Pătraşcu]: space $<$ optimal + $n / \lg^{\Omega(q)} n$ **non-adaptive**

Summary

- New lower bounds for basic data structures:

Representing trits, sets, prefix sums, balanced brackets
using space = optimal + redundancy

- Sometimes matching [Pătraşcu]

- **Open problems:** storing sets:

2 cell probes and optimal space?

Bit-probe lower bounds for set-size $n/4$? (have $n/3$)

Future directions

- Lower bounds for generating distributions
- **Example:** $f : \{0,1\}^r \rightarrow \{0,1\}^n$
each bit f_i depends on $\leq q$ input bits
prove $f(\text{uniform})$ **far** from uniform on sets of size $n/4$
- **Known[V.]:** distance $\geq 1/2^{O(q)}$
- **Open:** distance $\geq 1 - o(1)$
 \Rightarrow Lower bound for storing sets of size $n/4$

- $\Sigma \Pi \sqrt{\neq} \cup \supseteq \not\subset \subseteq \in \Downarrow \Rightarrow \Uparrow \Leftarrow \Leftrightarrow \vee \wedge \geq \leq \forall \exists \Omega \alpha \beta \epsilon \gamma \delta \rightarrow$
- $\neq \approx$