Deterministic Cache-Oblivious Funnelselect

Gerth Stølting Brodal
AARHUS UNIVERSITY

Sebastian Wild
UNIVERSITY OF LIVERPOOL

19th Scandinavian Symposium on Algorithm Theory (SWAT), June 12-14, 2024, Helsinki, Finland

Problem

Sorting

77 24 36 58 70 1 29 42 13 92

1 13 24 29 36 42 58 70 77 92

rank 2 4 10

Single selection

77 24 36 58 70 1 29 42 13 92

rank 4 → 29

Multiple selection

77 24 36 58 70 1 29 42 13 92

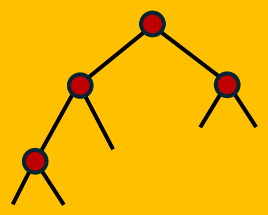
ranks 2 4 10 → 13 29 92

sorted

Algorithm

Randomized

Deterministic



Result of this talk

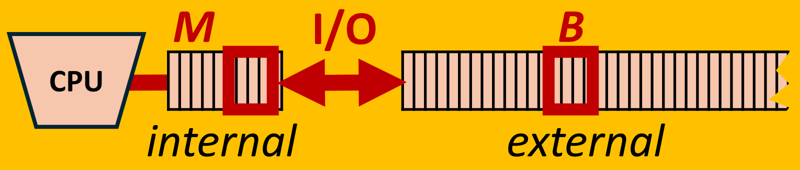
Computational Model

Internal memory

Memory access (element comparisons)

77 24 36 58 70 1 29 42 13 92

External memory Cache aware, I/O

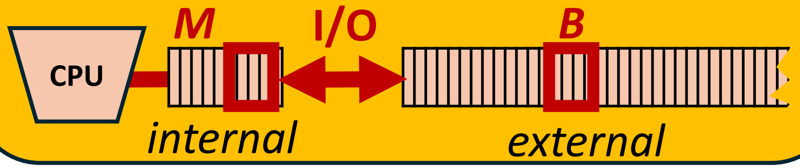


Aggarwal, Vitter 1988

Cache oblivious

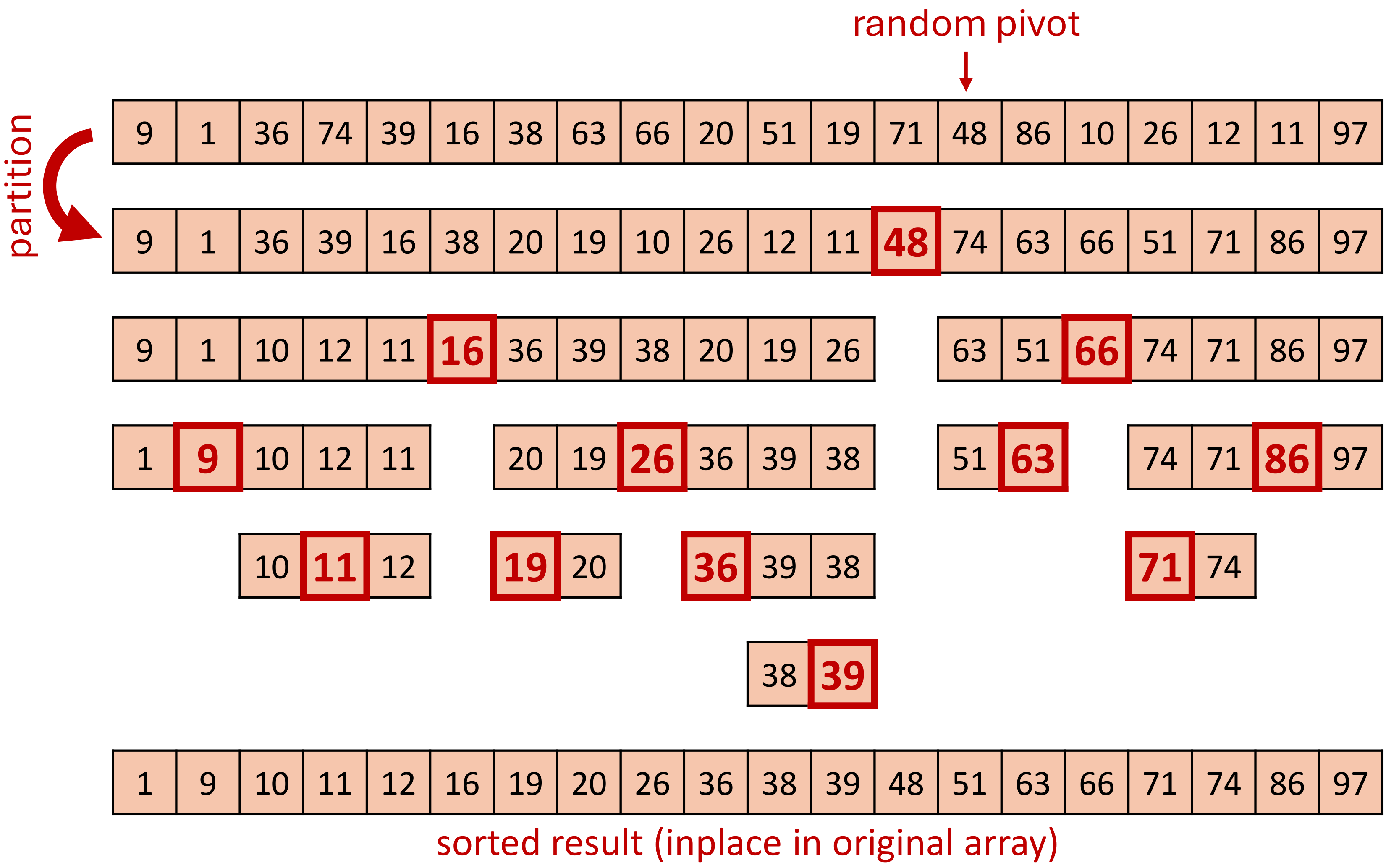
Algorithms do not know *B* and *M*

(optimal offline cache replacement)



Frigo, Leiserson, Prokop, Ramachandran 1999

Internal Memory **Sorting** – Randomized Quicksort

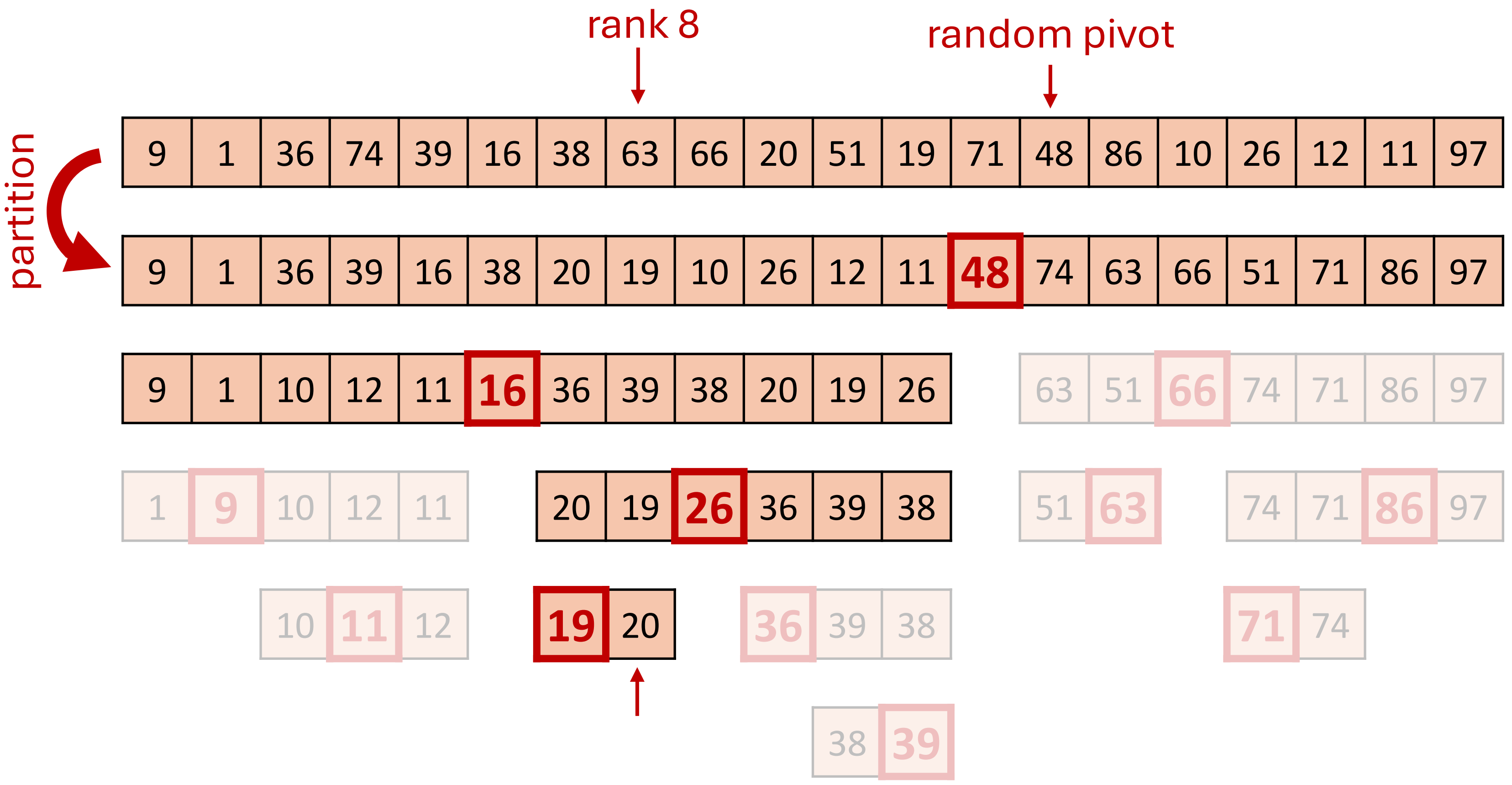


- [Hoare 1959]

  Expected time

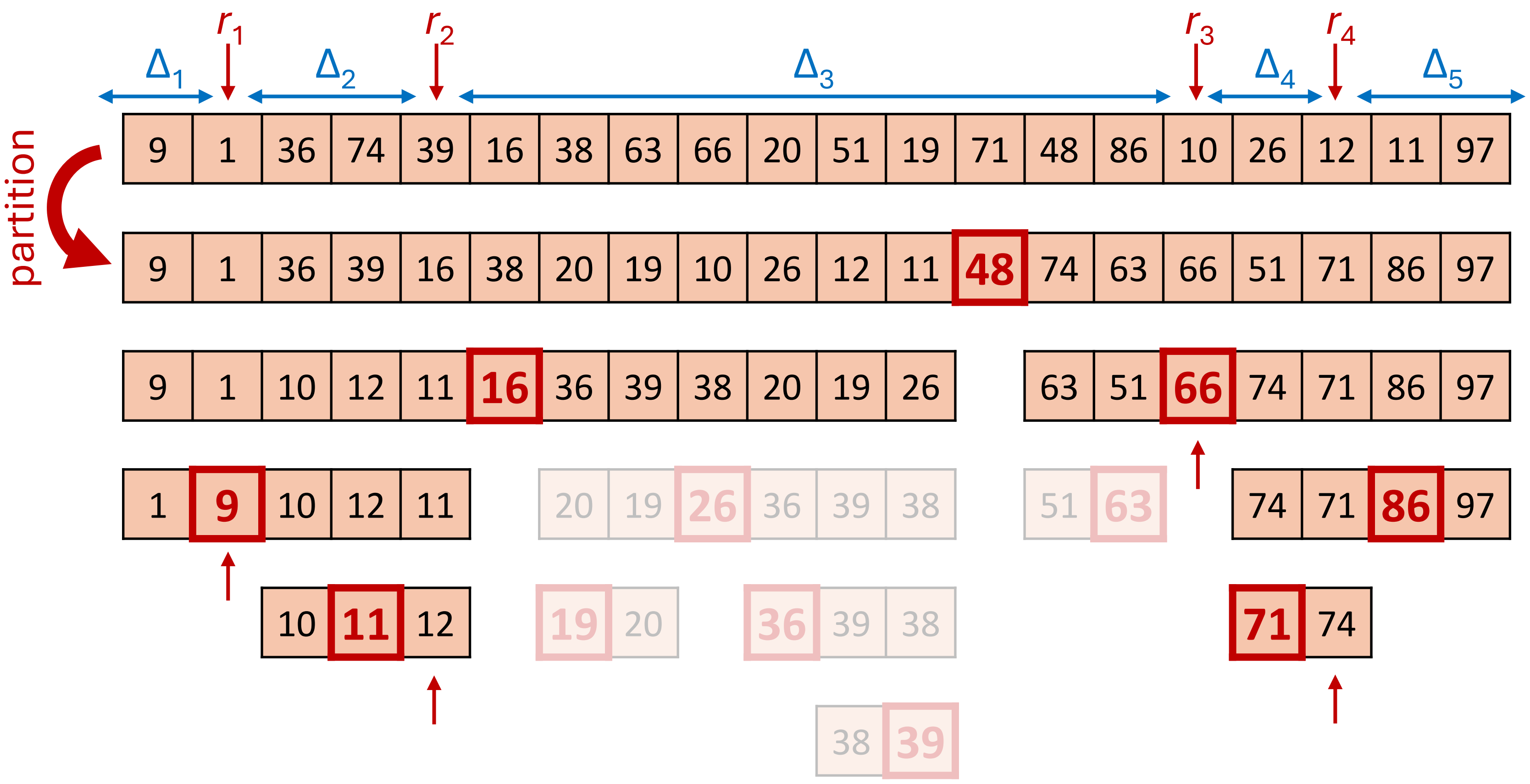
  $O(n \log_2 n)$

Internal Memory **Single Selection** – Randomized



- Quickselect [Hoare 1961]
- Expected time
  $O(n)$

Internal Memory **Multiple Selection** – Randomized



- Multiple selection [Chambers 1971]
- Expected time

  $O(n \log_2 q)$

  for q queries
- Expected time [Prodinger 1995]

$O(\mathcal{B} + n)$

$$\mathcal{B} = \sum_{i=1}^{q+1} \Delta_i \log_2 \frac{n}{\Delta_i}$$

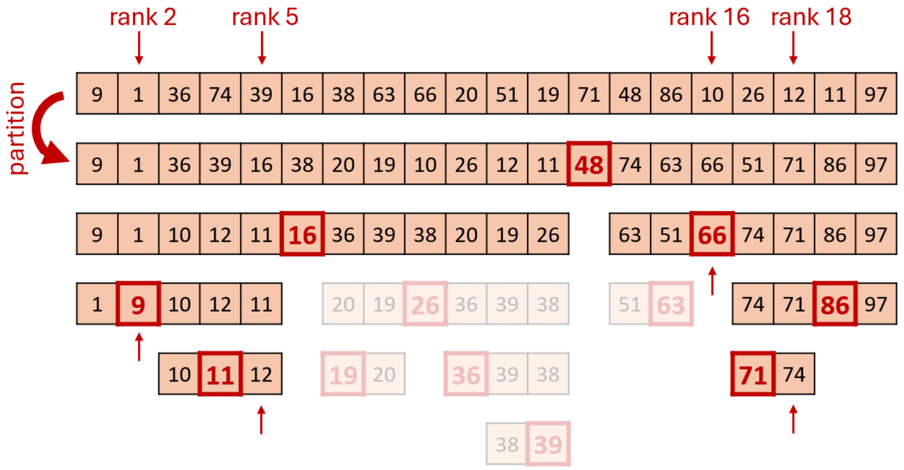
(query rank entropy)

Internal Memory **Multiple Selection** – Deterministic

Multiple selection
[Chambers 1971]

+

Deterministic linear median → pivots
[Blum, Floyd, Pratt, Rivest, Tarjan 1973]



↓

Deterministic multiple selection
[Dobkin, Munro 1981]

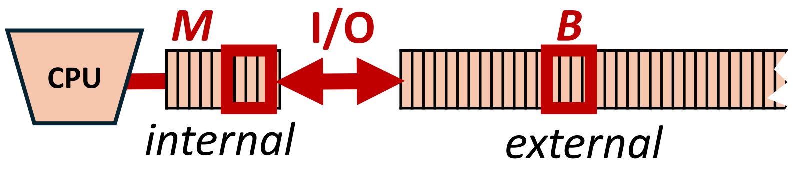
optimal $O(\mathcal{B} + n)$ time

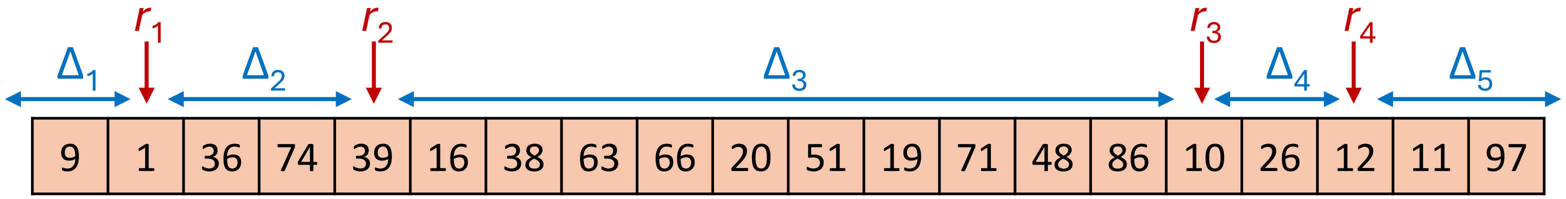
[Frigo, Leiserson, Prokop, Ramachandran 1999]

	Internal memory	Cache-oblivious I/Os	Cache-oblivious Optimal I/Os ?
Sorting	$O(n \log_2 n)$	$O\left(\frac{n}{B} \log_2 \frac{n}{M}\right)$	$O\left(\frac{n}{B} \log_{M/B} \frac{n}{M}\right)$
Single selection	$O(n)$	$O(n / B)$	$O(n / B)$
Multiple selection	$O(\mathcal{B} + n)$	$O(n / B + \mathcal{B} / B)$	$O(n / B + \mathcal{B} / B / \log_2 \frac{M}{B})$

$$\mathcal{B} = \sum_{i=1}^{q+1} \Delta_i \log_2 \frac{n}{\Delta_i}$$

(query rank entropy)

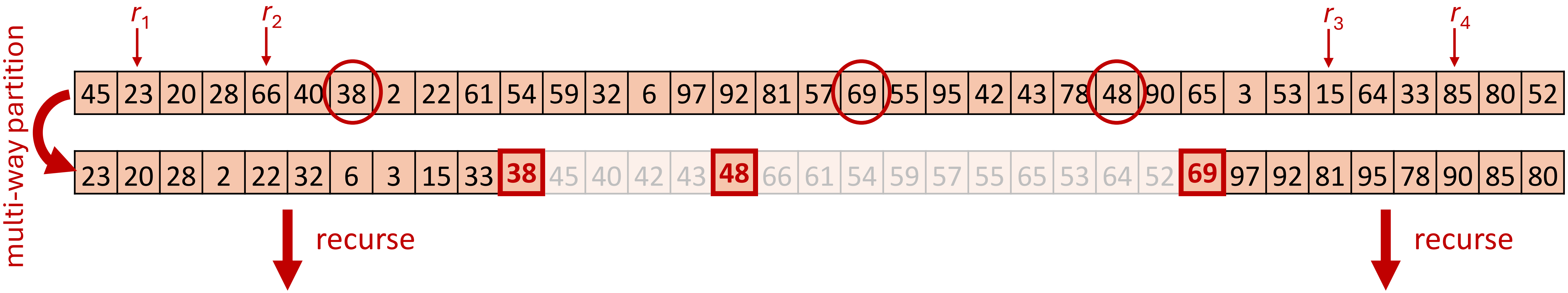




Lower bound
[Dobkin, Munro 1981] +
[Arge, Knudsen, Larsen 1993]

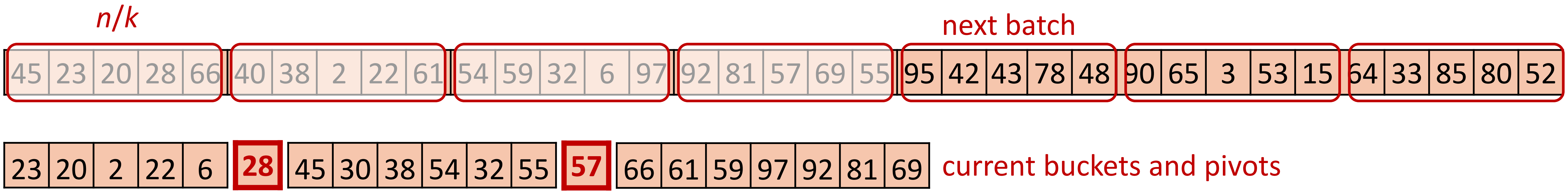
Upper bounds
Randomized [ESA 2023]
Deterministic [SWAT 2024]

Cache-aware **Multiple Selection** – Randomized



- Pick $\Theta(M/B)$ random pivots and distribute to buckets
- Recurse on buckets with queries (à la Chambers)
- Expected optimal $O(n / B + \mathcal{B} / B / \log_2 \frac{M}{B})$ I/Os

Cache-aware **Multi-way Partition** – Deterministic



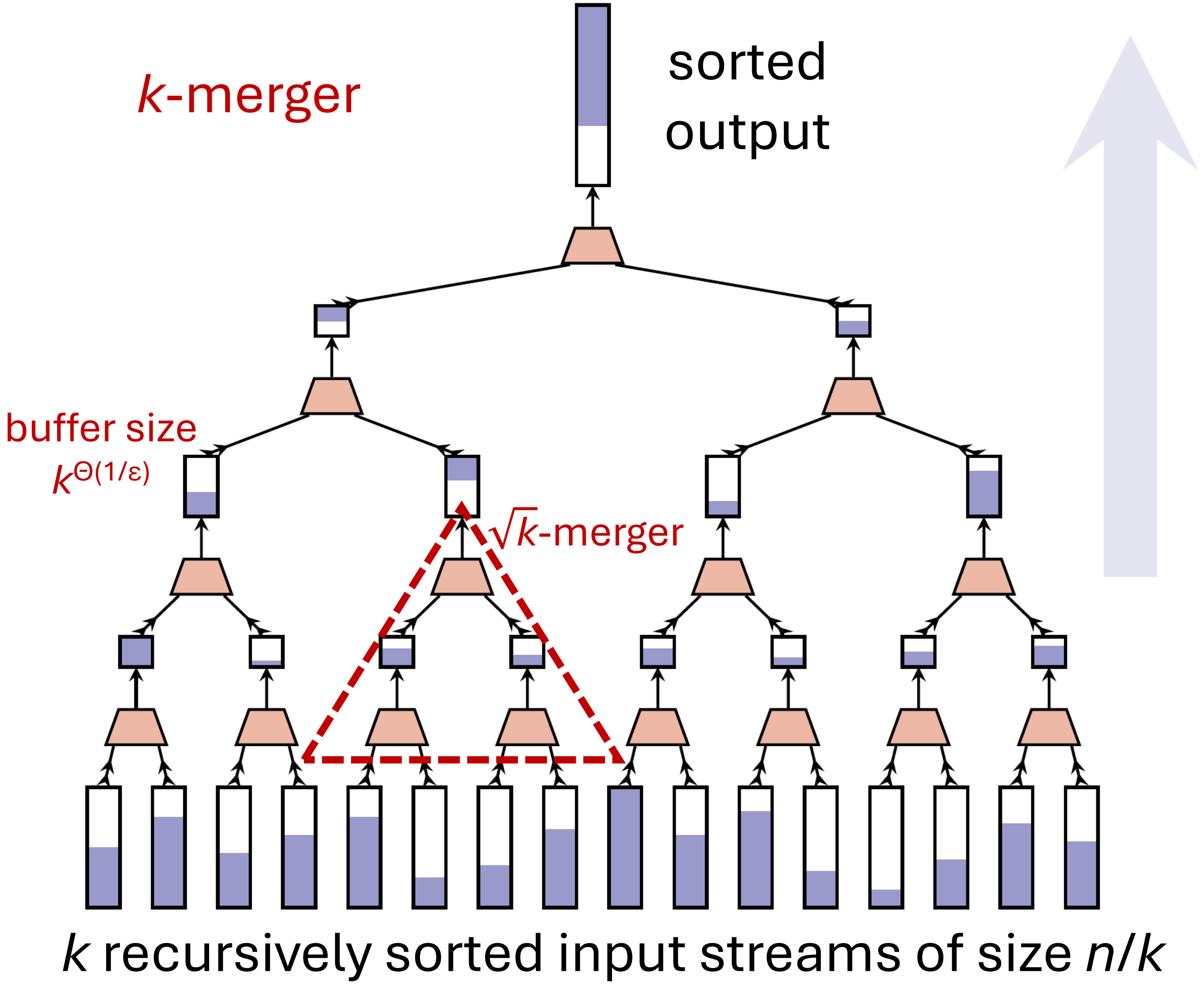
- **Goal** : $k = \Theta(M/B)$ buckets each of size $[n/k, 2n/k]$
- Repeatedly distribute batches of n/k elements into buckets (initially one)
- Split overflowing buckets ($> 2 \cdot n/k$ elements; at most $3n/k$) [using Blum *et al*. median finding algorithm; median new pivot]

- **Distribution sort** : $O\left(\frac{n}{B} \log_{M/B} \frac{n}{M}\right)$ I/Os and $O(n \log n)$ internal work
- **Multiple selection** : $O(n / B + \mathcal{B} / B / \log_2 \frac{M}{B})$ I/Os

[Hu, Tao, Yang, Zhou 2014] achieved both I/O and work optimality

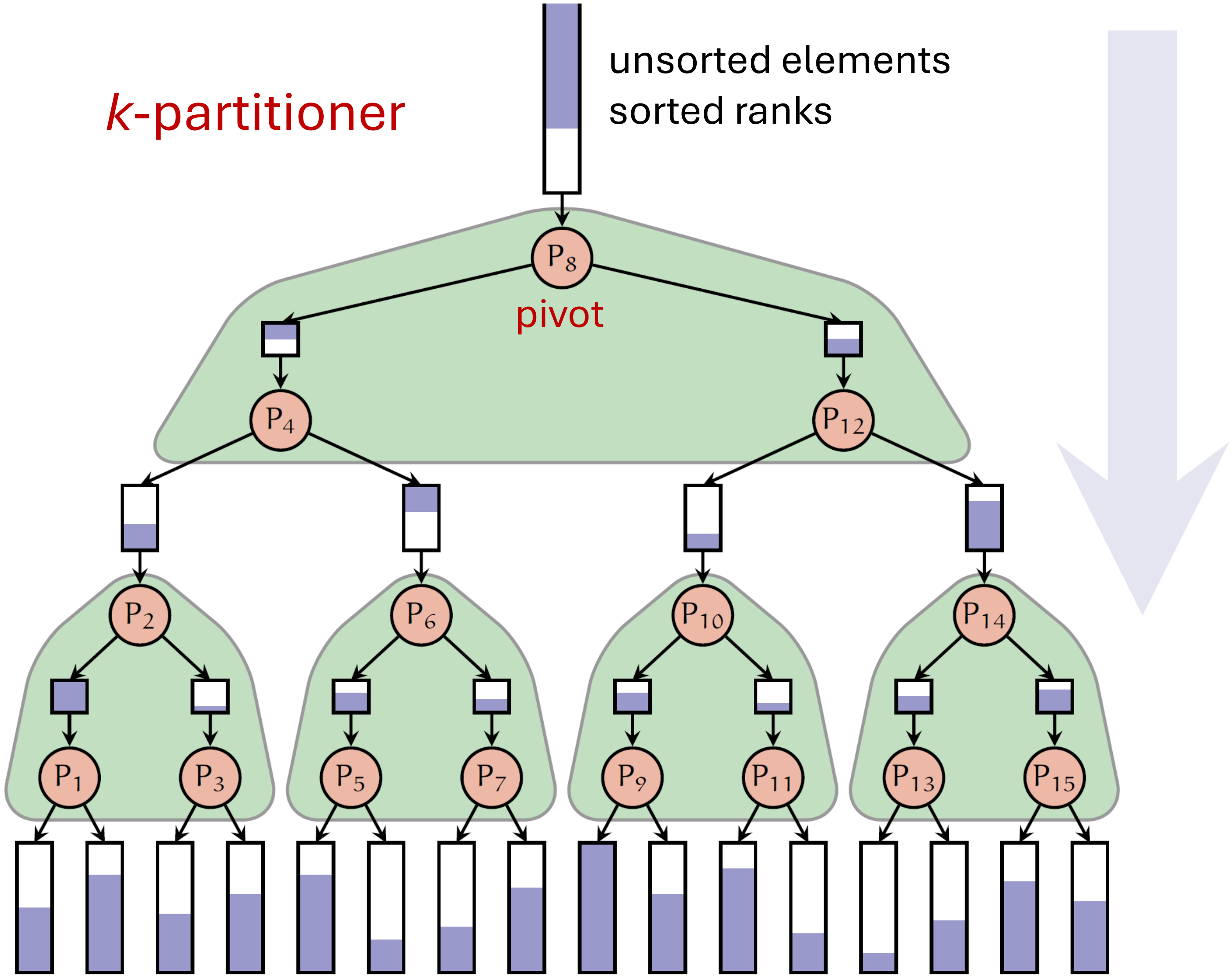
[Frigo, Leiserson, Prokop, Ramachandran 1999]

Cache-oblivious **Funnelsort** – Deterministic



- Tall cache assumption
  $M \geq B^{1+\varepsilon}$
  Neccessary [Brodal, Fagerberg 2002]
- Binary mergesort with buffered output
- *k*-merger , $k = n^{\Theta(\varepsilon)}$
- $O\left(\frac{n}{B}\log_{M/B}\frac{n}{M}\right)$ I/Os

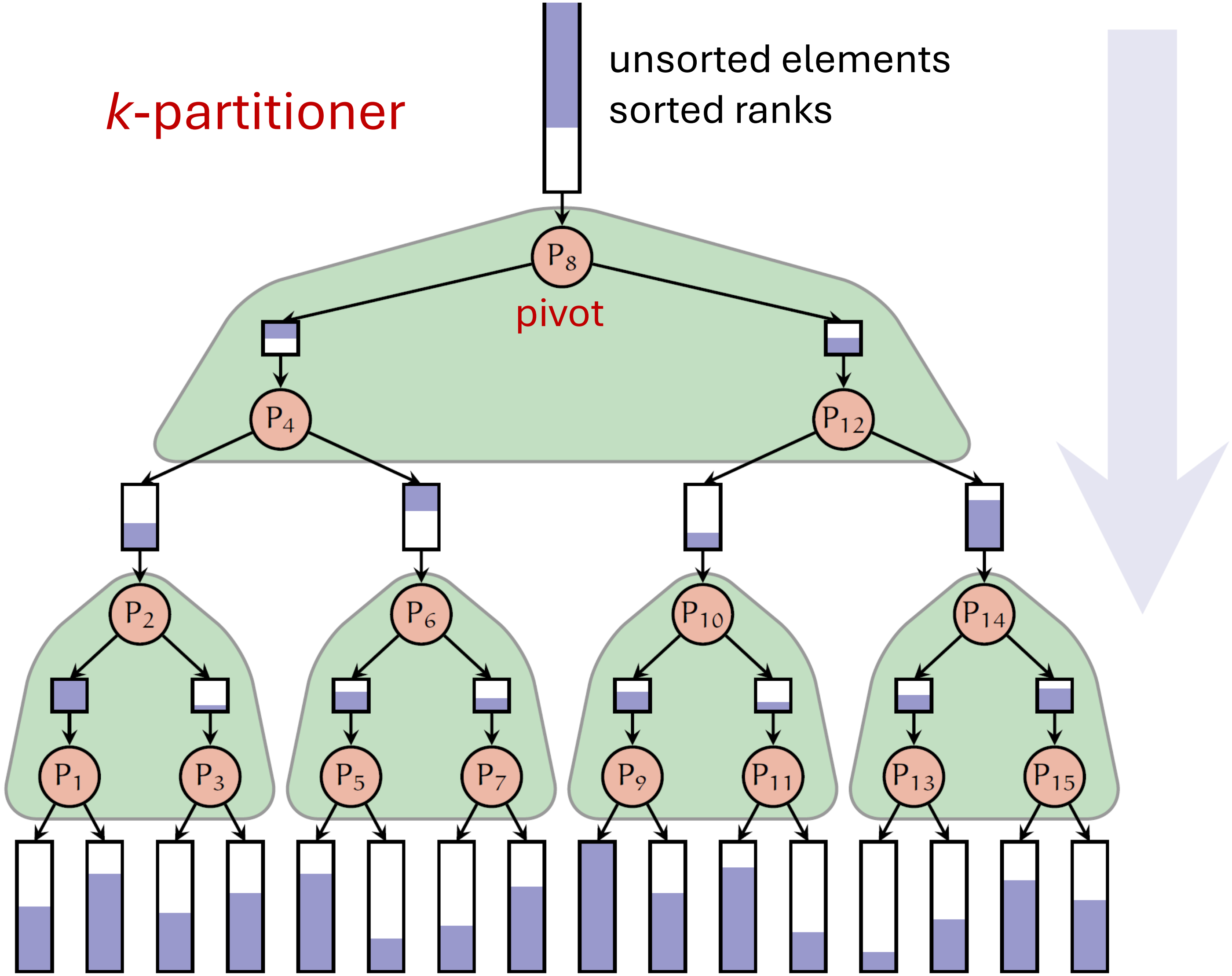
Cache-oblivious **Multiple Selection** – Idea



- Reverse computation of *k*-merger → *k*-partitioner
- algorithm à la Chambers
- Challenges
  - Pivots ?
  - Pruning subtree computations before knowning ranks of pivots ?

[Brodal, Wild 2023]

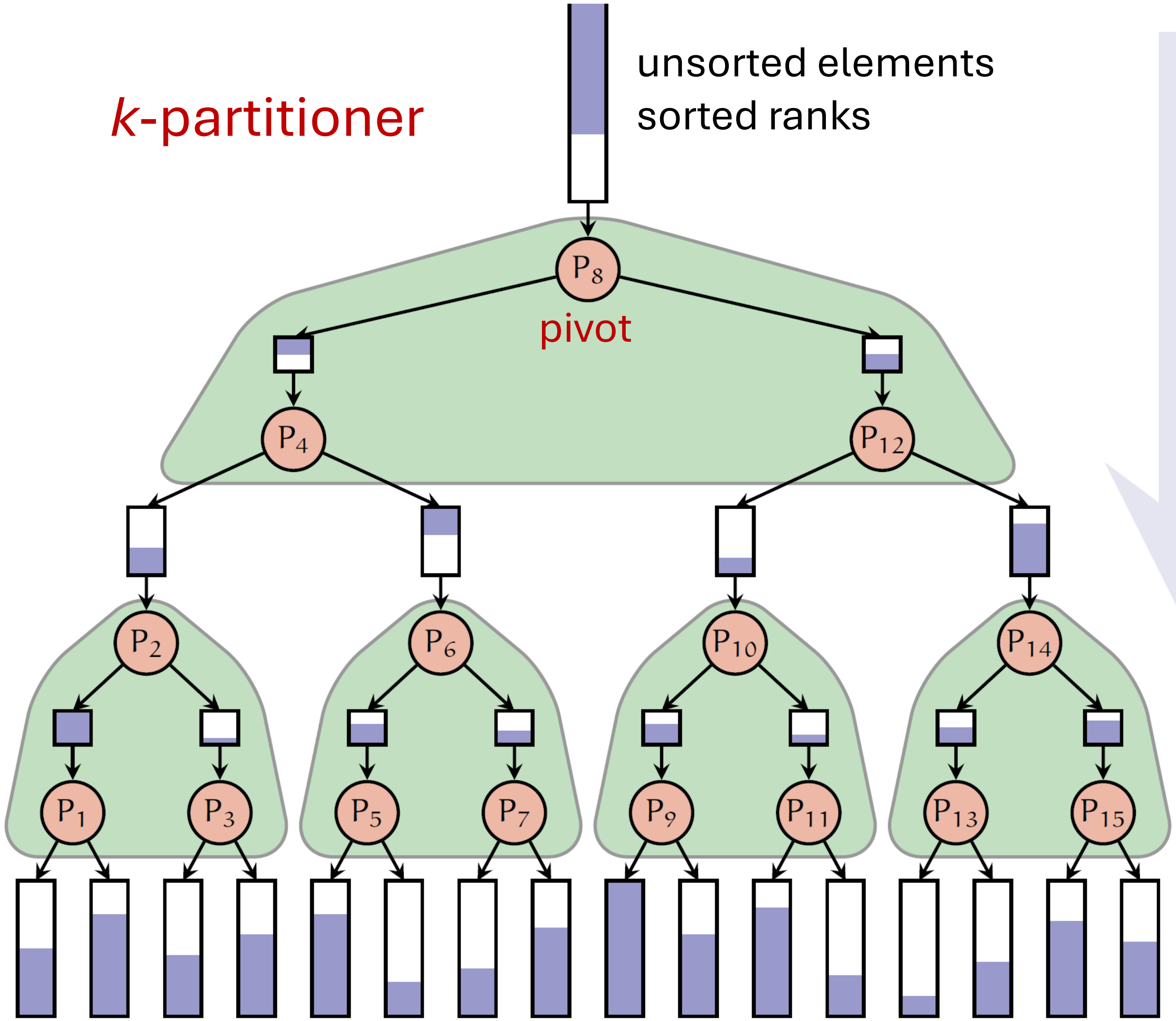
Cache-oblivious **Multiple Selection** – Randomized



- Sort n / log n size sample
- Select $k = n^{\Theta(\varepsilon)}$ pivots uniformly in sample
- Estimate pivot ranks within $\pm n^{2/3}$ w.h.p.
- Prune inside k-partitioner subtrees w.h.p. no query
- Buckets just sort
- Expected $O(n / B + \mathcal{B} / B / \log_2 \frac{M}{B})$ I/Os

[Brodal, Wild 2024]

Cache-oblivious **Multiple Selection** – Deterministic



- Entropy bound ≈ sort → sort
- Otherwise, small $k \le n^{\Theta(\varepsilon)}$
- Prune buckets with no query
  - ≥ $n/2$ elements pruned
- Recurse on buckets
- Pivots deterministically
  - Incremental batches of size n/k
  - Split bucket + rebuild k-partitioner
- $O(n / B + \mathcal{B} / B \,/\, \log_2 \frac{M}{B})$ I/Os

$$k = n/\Delta$$

$$\Delta = \min \{\Delta_i \mid \textstyle\sum_{\Delta_j \le \Delta_i} \Delta_j \ge n/2\}$$

Summary

Problems

Sorting

77 | 24 | 36 | 58 | 70 | 1 | 29 | 42 | 13 | 92

1 | 13 | 24 | 29 | 36 | 42 | 58 | 70 | 77 | 92

2, 4, 10

Single selection

77 | 24 | 36 | 58 | 70 | 1 | 29 | 42 | 13 | 92

rank 4 → 29

Multiple selection

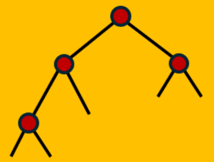
77 | 24 | 36 | 58 | 70 | 1 | 29 | 42 | 13 | 92

ranks 2 | 4 | 10 (*sorted*) → 13 | 29 | 92

Algorithm

Randomized

Deterministic



Result of this talk

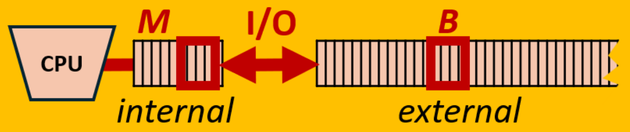
Computational Model

Internal memory

Memory access (element comparisons)

77 | 24 | 36 | 58 | 70 | 1 | 29 | 42 | 13 | 92

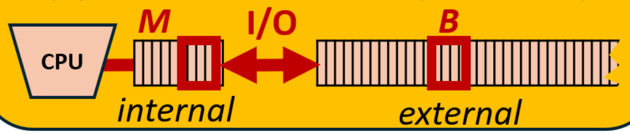
External memory Cache aware, I/O



Aggarwal, Vitter 1988

Cache oblivious

Algorithms do not know B and M (optimal offline cache replacement)



Frigo, Leiserson, Prokop, Ramachandran 1999

Optimal $O(n / B + \mathcal{B} / B / \log_2 \frac{M}{B})$ I/Os