

Comparator Networks for Binary Heap Construction

Gerth Stølting Brodal

**Max-Planck-Institut für Informatik
Saarbrücken, Germany**

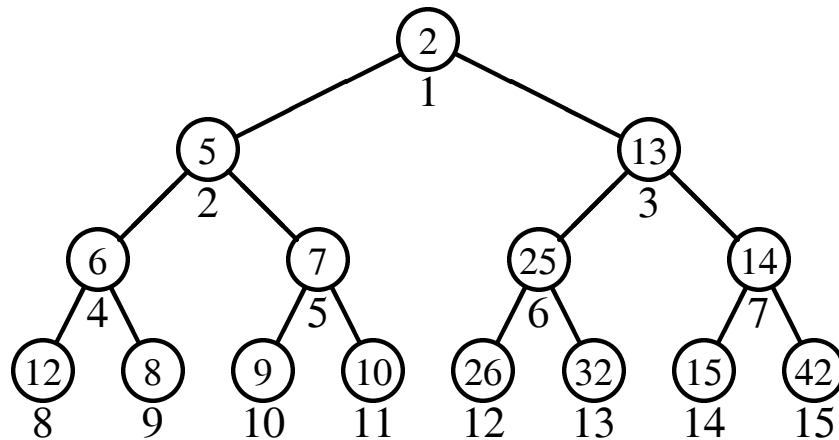
M. Cristina Pinotti

**Istituto di Elaborazione della Informazione
CNR, Pisa, Italy**

July 1998

Binary Heap

– *Williams 1964*



2	5	13	6	7	25	14	12	8	9	10	26	32	15	42
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Priority queue

– Insert, DeleteMin $\mathcal{O}(\log n)$

Sorting $\mathcal{O}(n \log n)$

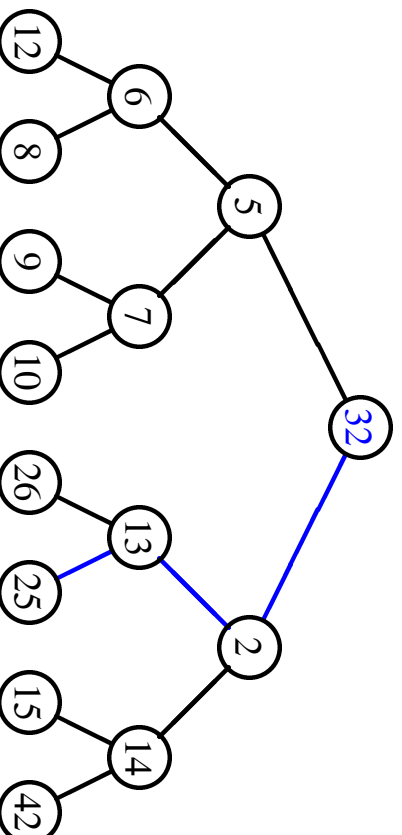
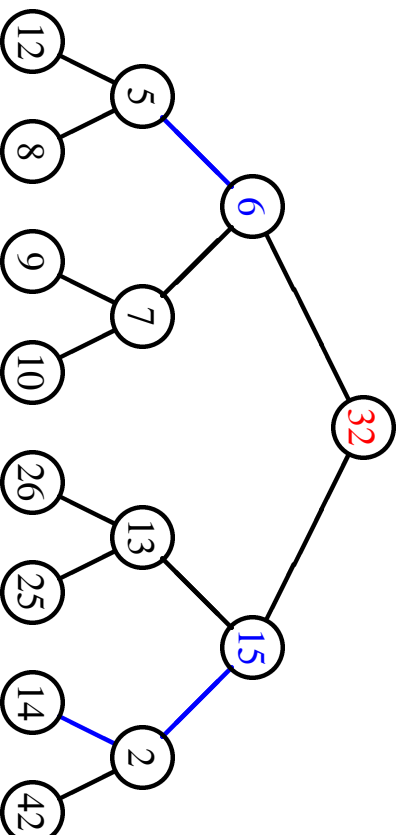
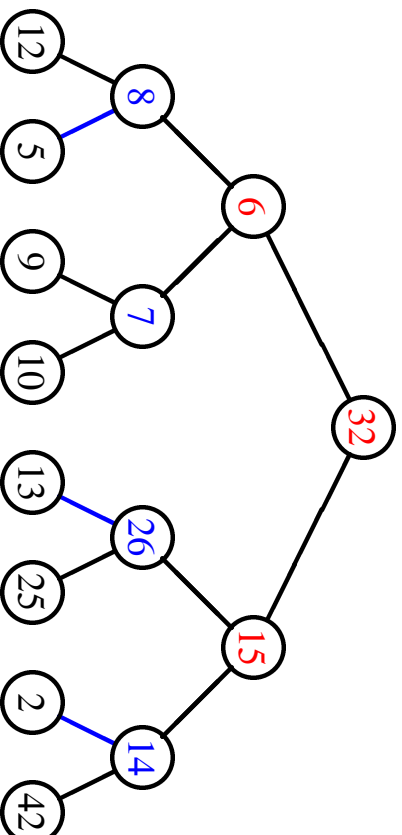
Construction $\mathcal{O}(n)$

Selecting the t smallest $\mathcal{O}(t)$

Implicit representation

The Sequential Binary Heap Construction Algorithm

– Floyd 1964



Alg. For each node $i = n, \dots, 1$ apply **SiftDown**(i)

Binary Heap Construction

Sequential

- $\mathcal{O}(n)$ – *Floyd 1964*

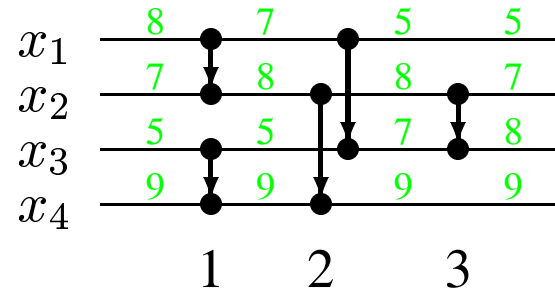
Optimal parallel algorithms

- EREW PRAM: $\mathcal{O}(\log n)$ – *Olariu, Wen 1991*
- CRCW PRAM: $\mathcal{O}(\log \log n)$ – *Dietz, Raman 1994*
- Randomized parallel comparison tree model: $\mathcal{O}(\alpha(n))$ – *Dietz 1992*

Question

- Comparator networks (the simplest parallel model of computation) ?

Comparator Networks



An optimal sorting network for $n = 4$

Merging networks: $\mathcal{O}(n \log n)$

– *Batcher 1960's*

Sorting networks: $\mathcal{O}(n \log^2 n)$

– *Batcher 1960's*

AKS sorting networks: $\mathcal{O}(n \log n)$

– *Ajtai, Komlós, Szemerédi 1983*

Median selection networks: $\mathcal{O}(n \log n)$

– *Ajtai, Komlós, Szemerédi 1983*

Sorting, merging, and median require networks of size $\Omega(n \log n)$

– *Alekseev 1969*

– *Miltersen, Paterson, 1996*

Note Sequentially merging and median can be solved in linear time

– *Blum, Floyd, Pratt, Rivest, Tarjan 1973*

Binary Heaps and Comparator Networks

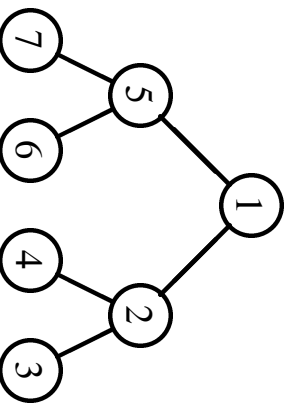
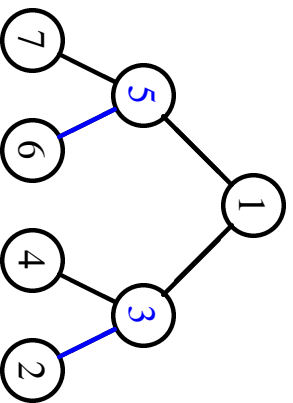
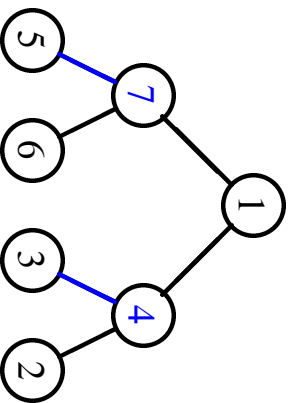
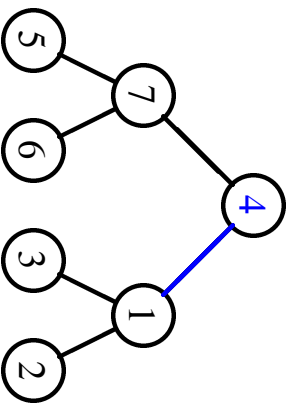
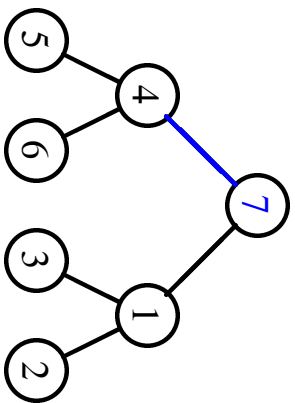
Heap Construction Networks

- Comparator networks that permute n inputs to an implicit heap
- $H(n)$ = minimum size of a heap construction network
- $H(n) = \mathcal{O}(n \log n)$ because of $\mathcal{O}(n \log n)$ sorting networks

(n, t) -Selection Networks

- Comparator networks that select the t smallest elements from n inputs
- $S(n, t)$ = minimum size (n, t) -selection network

$\mathcal{O}(n \log n)$ Heap Construction Networks



SiftDown networks

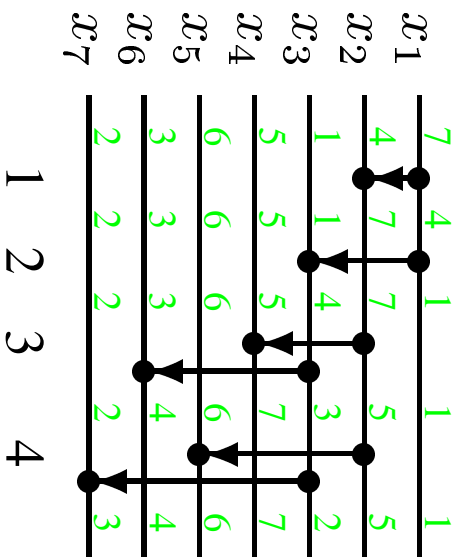
- Each tree edge \equiv comparator
- Size $n - 1$, depth $\mathcal{O}(\log n)$

Heap construction network

- Size

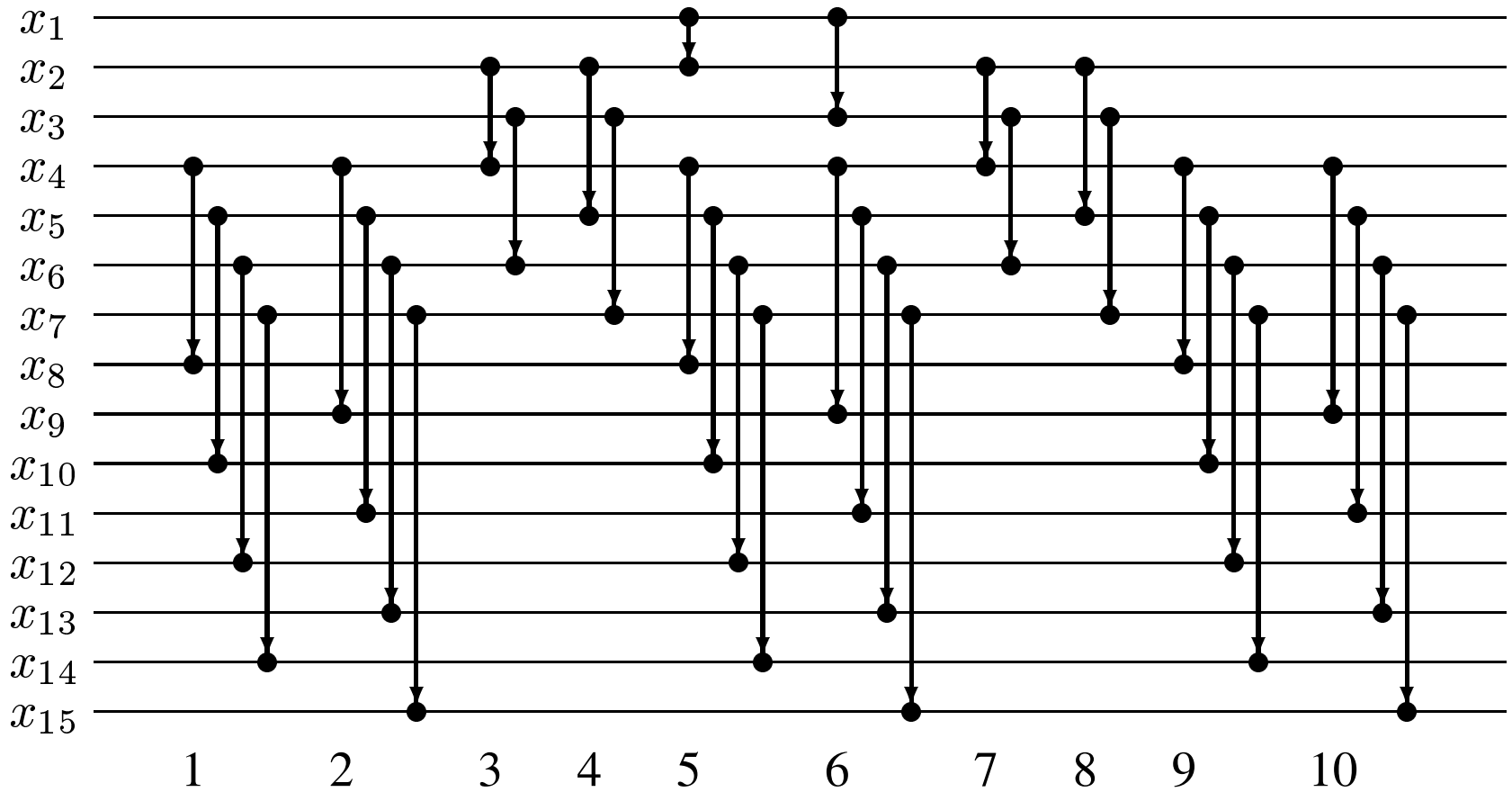
$$\mathcal{O}\left(\sum_{i=0}^{\log n} 2^i \cdot \frac{n}{2^i}\right) = \mathcal{O}(n \log n)$$

- Depth $\mathcal{O}(\log n)$



SiftDown network, $n = 7$

$\mathcal{O}(n \log n)$ Heap Construction Network



$n = 15$

Size = 34

Depth = 10

$\mathcal{O}(n \log \log n)$ Heap Construction Networks

Lemma $S(n, \lfloor \frac{n}{2} \rfloor) \leq Cn \log n + \mathcal{O}(n)$, $C > \frac{3}{\log 3}$

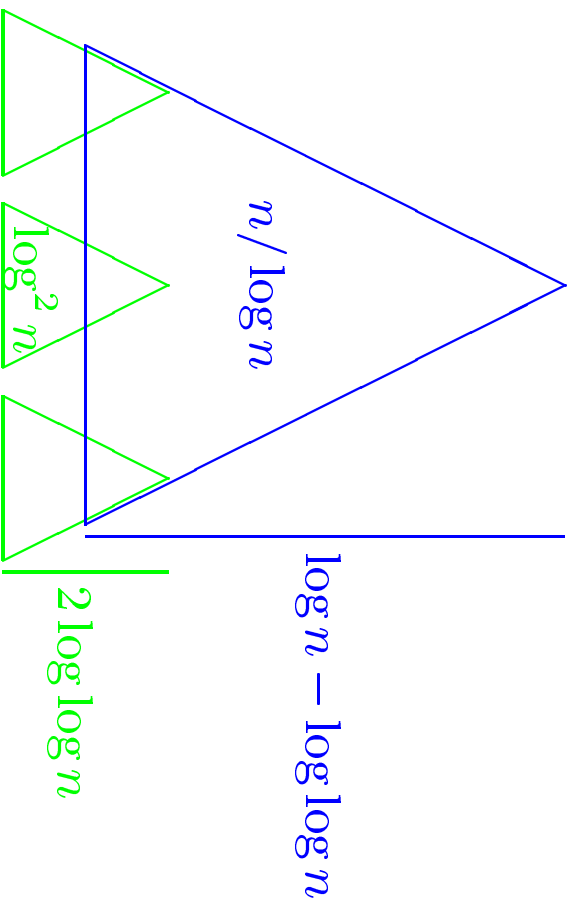
— *Jimbo, Maruoka 1996*

Corollary $S(n, t) = \mathcal{O}(n \log t)$

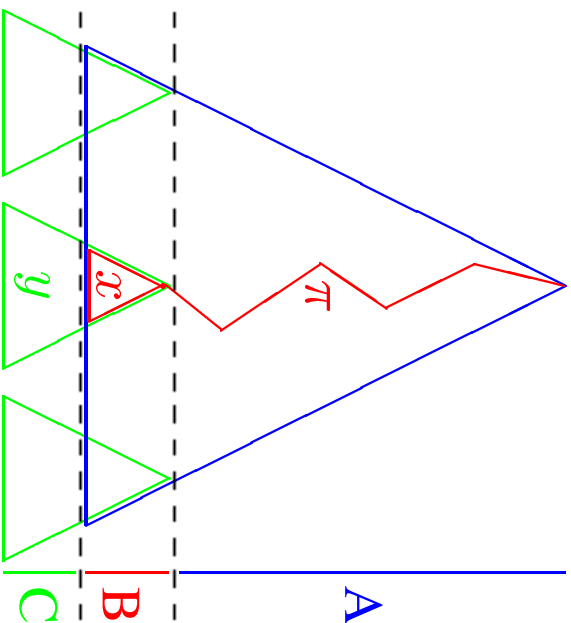
Theorem $H(n) = \mathcal{O}(n \log \log n)$

Proof of Theorem

1. Apply $(\log^2 n, \log n)$ -selection networks to each **green** subtree $\mathcal{O}(n \log \log n)$
2. Apply $\mathcal{O}(n \log n)$ -heap construction network to the **blue** subtree $\mathcal{O}(n)$
3. Apply $\mathcal{O}(n \log n)$ -heap construction network to each **green** subtree $\mathcal{O}(n \log \log n)$



$\mathcal{O}(n \log \log n)$ Heap Construction Networks



i) **A** is heap-ordered after Step 2

ii) **BC** is heap-ordered after Step 3

In the $\mathcal{O}(n \log n)$ -heap construction networks
tree edges \equiv comparators \Rightarrow elements in $\pi \cup x$ can
only be replaced by smaller elements in Step 2

For Step 2

$$x \leq y \quad \wedge \quad |\pi| \leq |x| \quad \Rightarrow \quad \pi' \leq x' \cup y$$

iii) **ABC** is heap-ordered after Step 3

An $\Omega(n \log \log n)$ Lower Bound

Lemma $S(n, t) \geq (n - t) \lceil \log(t + 1) \rceil$

— *Alekseev 1969*

Theorem $H(n) \geq S(n, \lfloor \log n \rfloor) = \mathcal{O}(n)$

Corollary $H(n) \geq n \log \log n - \mathcal{O}(n)$

Proof of Theorem

Idea Reduce $(n, \lfloor \log n \rfloor)$ -selection networks to heap construction networks

1. Construct a heap $H(n)$
2. Iteratively *delete* the $\lfloor \log n \rfloor$ smallest elements

To find the t smallest elements only the t topmost levels of the heap need to be considered

Sufficient to apply the linear size **SiftDown** networks to heaps of size $n, n/2, n/4, n/8, \dots$

Total size of the **SiftDown** networks

$$\sum_{i=0}^{\log n} \frac{n}{2^i} = \mathcal{O}(n)$$

Conclusion

- Heap construction networks with size $\mathcal{O}(n \log \log n)$ and depth $\mathcal{O}(\log n)$
- An $\Omega(n \log \log n)$ lower bound for the size of heap construction networks

Theorem If for constants C_1 and C_2

$$C_1 n \log t - \mathcal{O}(n) \leq S(n, t) \leq C_2 n \log t + \mathcal{O}(n)$$

then

$$C_1 n \log \log n - \mathcal{O}(n) \leq H(n) \leq C_2 n \log \log n + \mathcal{O}(n \log \log \log n)$$