

7 Sortierte Listen



Operationen

Ausgabegröße L , Δ = Abstand zu Finger

insert, remove, update, locate

$O(\log \Delta)$

$(M.\text{locate}(k) := \min \{e \in M : e \geq k\})$

rangeSearch

$O(\log \Delta) + L$

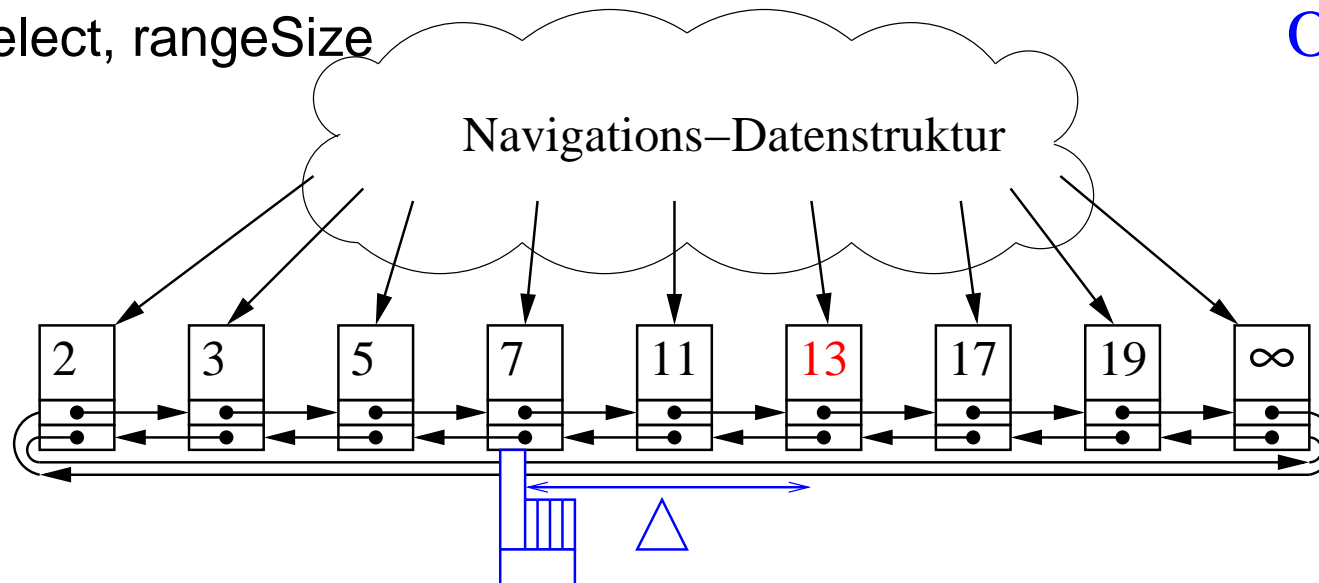
split, concat

$O(\log n)$

Teilbaumgrößen-Augmentierung:

rank, select, rangeSize

$O(\log n)$



Abgrenzung

Hash-Tabelle: nur insert, remove, find. Kein locate, rangeQuery

Sortiertes Feld: nur bulk-Updates. Aber:

Hybrid-Datenstruktur oder $\log \frac{n}{M}$ geometrisch wachsende
statische Datenstrukturen

Prioritätsliste: nur insert, deleteMin, (decreaseKey, remove). Dafür:
schnelles merge

Anwendungsbeispiel: Scheduling

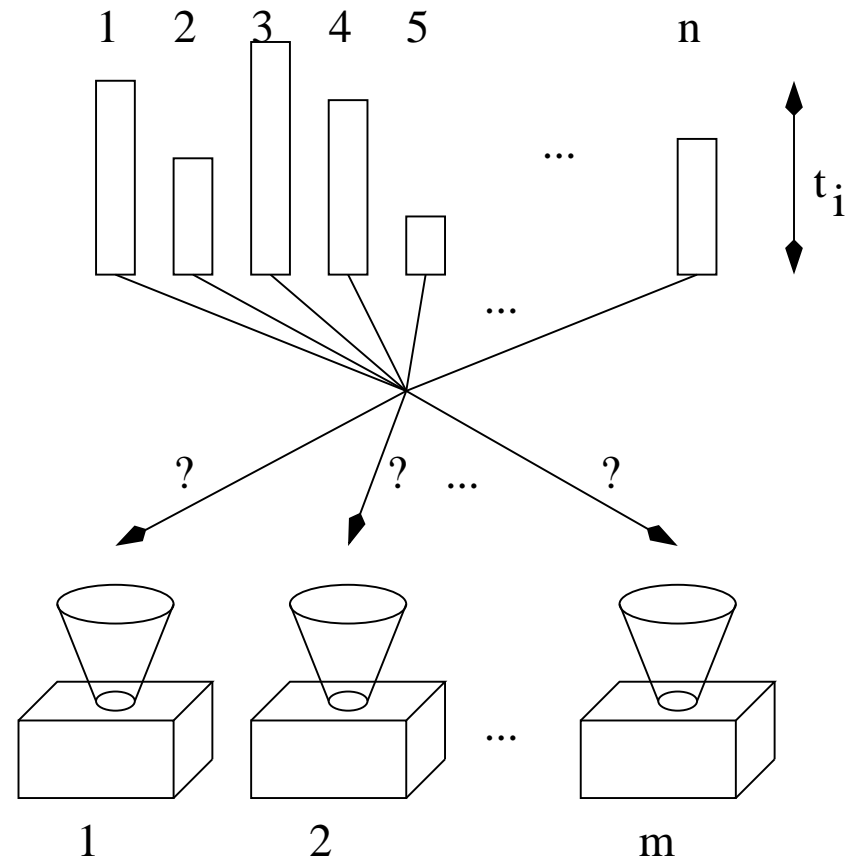
unabhängiger, gewichteter Jobs, parallele Maschinen

$x(j)$: Maschine auf der
Job j ausgeführt wird

L_i : $\sum_{x(j)=i} t_j$, Last
von Maschine i

Zielfunktion: Minimiere **Makespan**

$$L_{\max} = \max_i L_i$$



Anwendungsbeispiel: Scheduling

- NP-hart
- Wir geben eine $7/6$ -Approximation, die in Zeit $O(n \log m \log \text{opt})$ läuft

Best-Fit-Decreasing Algorithmus

assert $t_1 \geq \dots \geq t_n$

SortedSequence: gap = $\langle (B, 1), \dots, (B, m) \rangle$ // free time per machine

for $j := 1$ **to** n **do**

$h := \text{gap.locate}(t_j)$ // best fitting machine

if nothing found **then return** fail

$(g, i) := *h$

$x(j) := i$

 gap.update($h, (i, g - t_j)$)

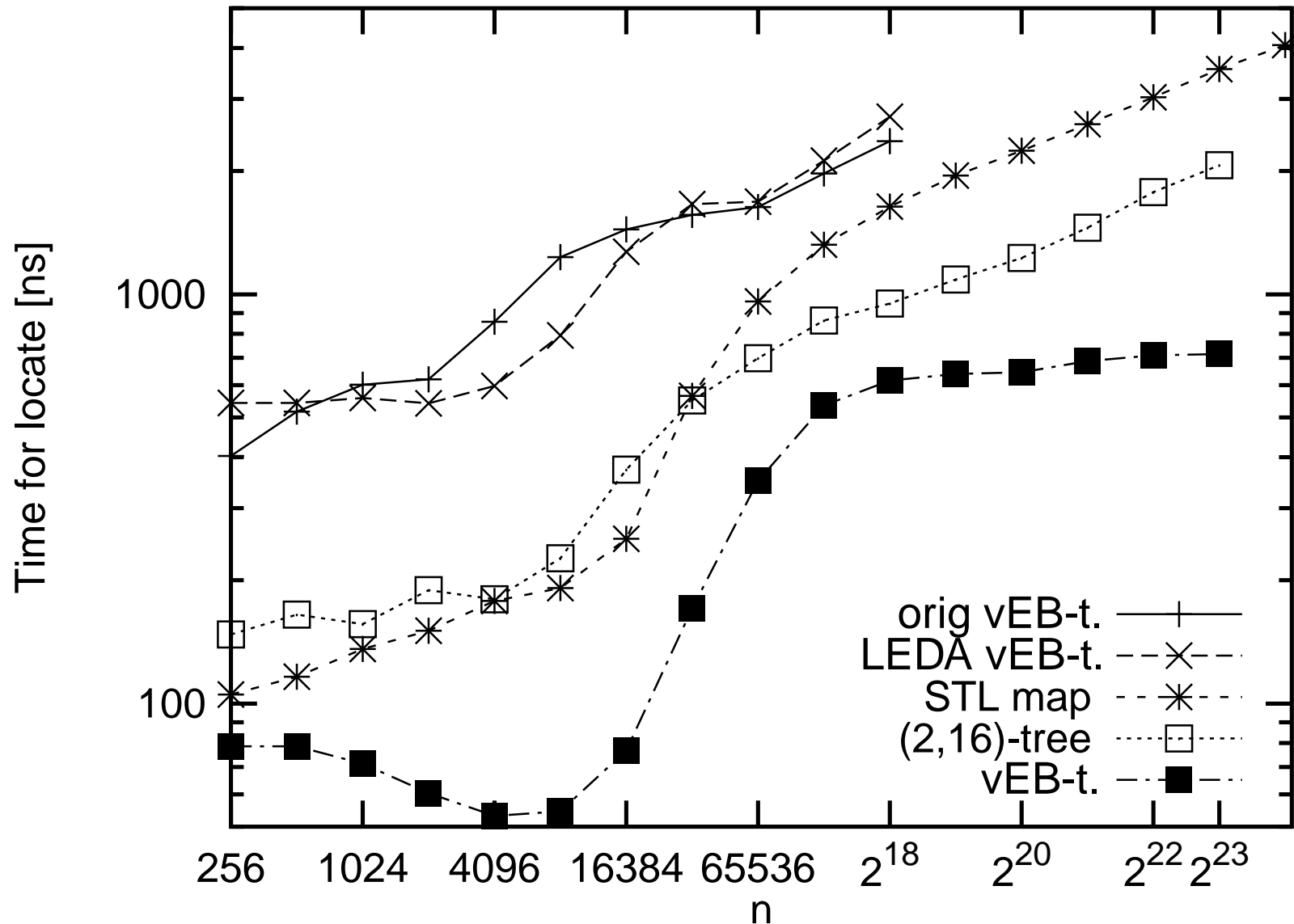
return x

Qualitäts-Analyse: nicht hier

Mehr Sortierte Listen

- Binäre Suchbäume
 - AVL-Bäume [Adelson, Velskii, Landis 1962]
 - Rot-Schwarzbäume (gern verwendet)
 - Randomisierte Suchbäume (Zeiger-Hashing benutzen!)
[Aragon Seidel 1996]
 - Splay Trees (selbstanpassend),...
- (a, b) -Bäume: feste Tiefe, $O(\log_a n)$ -I/Os
- skip-Listen
- Sparse Table
- vEB-Trees (ganzz. Schlüssel) Zeit $O(\log \log |U|)$ ($M \subseteq 0..U - 1$)
[van Emde Boas 1977..Dementiev Kettner Mehlhorn Sanders 2004]

vEB-Trees (mehr in Algorithm Engineering)



Noch mehr

Verallgemeinerungen, weitere Augmentierungen

- Geometrische** Suchbäume (Quad-Tree, KD-Tree, . . .)
Suche in höheren Dimensionen
- Segment-Bäume: **Intervall**aufspiessanfragen
- Zeichenketten**
- Persistente** Datenstrukturen: Man behält beliebige Versionen