Memory Hierarchies

[FLPR12] Matteo Frigo, Charles E. Leiserson, Harald Prokop, Sridhar Ramachandran. Cache-Oblivious Algorithms. ACM Transactions on Algorithms, 8(1), Article No. 4, 2012.

[BFJ02] Gerth Stølting Brodal, Rolf Fagerberg, Riko Jacob. Cache-Oblivious Search Trees via Binary Trees of Small Height. In Proc. 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 39-48, 2002.

[JM13] Tomasz Jurkiewicz, Kurt Mehlhorn. The cost of address translation, In Proc. 15th Annual Meeting on Algorithm Engineering & Experiments (ALENEX), 148-162, 2013.

Memory Hierarchies vs Efficiency

- Cache misses (L1, L2, L3, ...)
- Prefetching
- Cache associativity
- Virtual to physical mapping
- Translation Look-aside Buffer (TLB)
- TLB misses

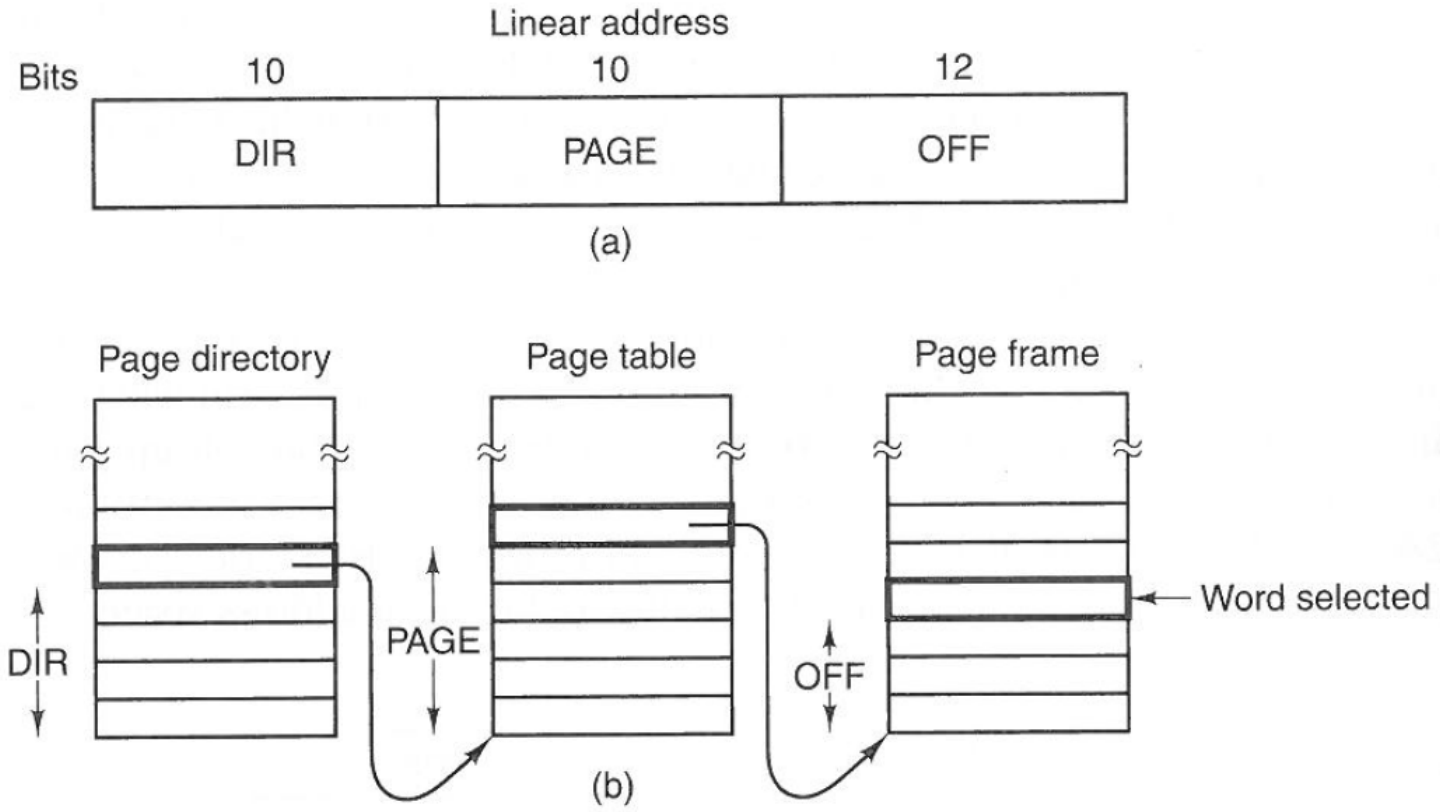
Some Typical Access times

Level	Access time	Cache line size
L1 Data ~16 KB L1 Instruction ~16 KB	5 ns	64 bytes
L2 ~512 KB	20 ns	64 bytes
L3 ~10 MB	30 ns	64 bytes
Main memory	60 ns	
Disk	10 ms	4 KB

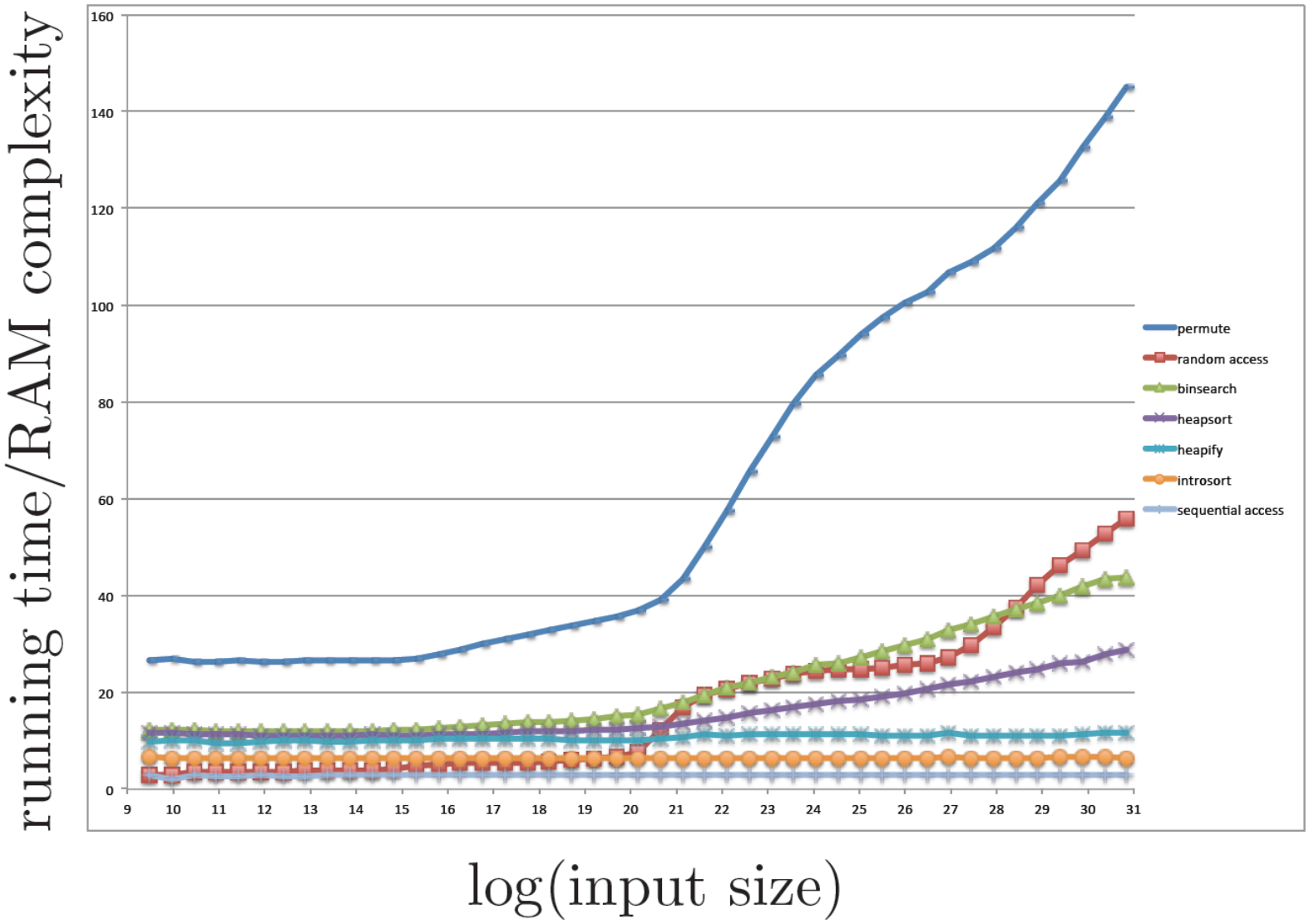
Intel(R) Core(TM) i7-3820 CPU @ 3.60GHz

- 32nm, 4 core [8 threads], L1, L2 and L3 line size 64 bytes
- L1 instruction 32K 8-way write-through per core
- L1 data 32K 8-way write-back per core
- L1 cache latency 3 clock cycles
- L2 256KB 8-way write-back unified cache per core
- L2 cache latency 12 clock cycles
- L3 10MB 20-way write-back unified cache shared by ALL cores
- L3 cache latency 26-31 clock cycles
- L1 instruction TLB , 4K pages, 64 entries, 4-way
- L1 data TLB, 4K pages, 64 entries, 4-way
- L2 TLB, 4K pages, 512 entries, 4-way
- ALL caches and TLBs use a pseudo LRU replacement policy

Virtual to Physical Address Mapping



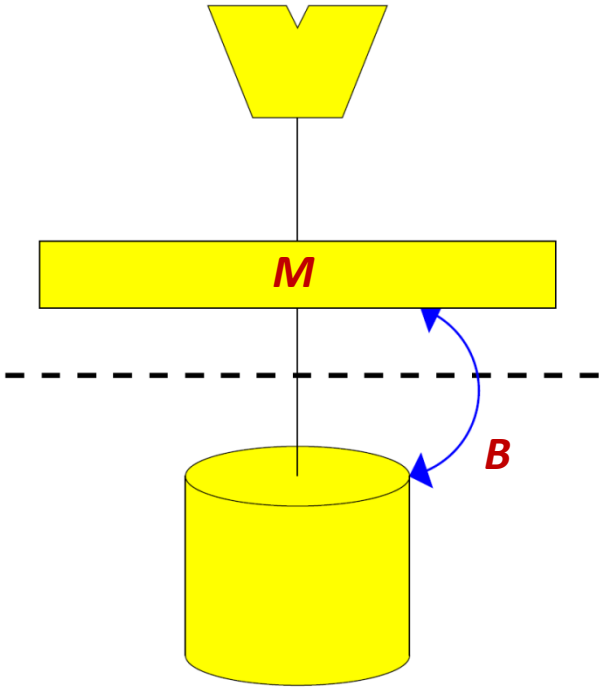
Cost of Address Translation



[JM13] Tomasz Jurkiewicz, Kurt Mehlhorn. The cost of address translation, In Proc. 15th Annual Meeting on Algorithm Engineering & Experiments (ALENEX), 148-162, 2013.

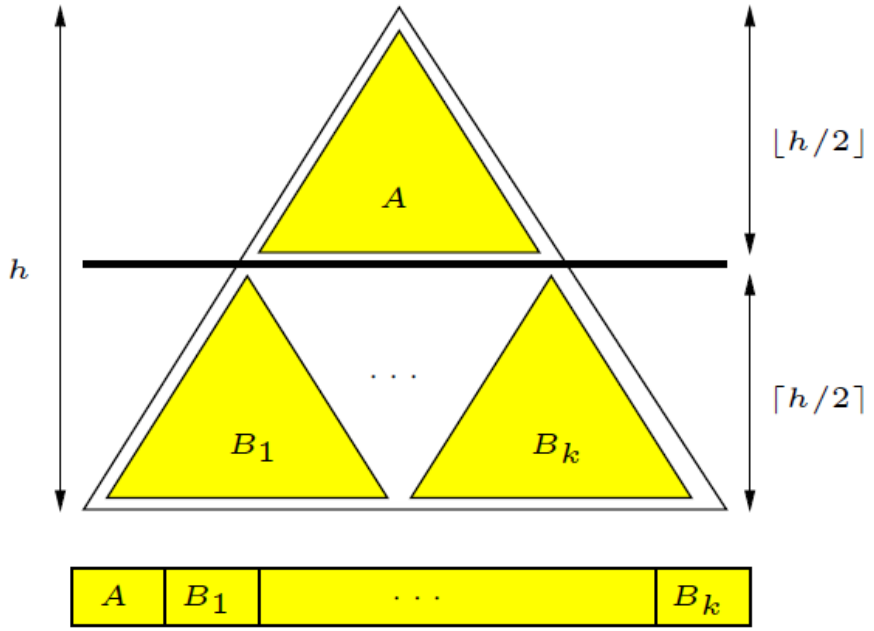
Cache-Oblivious Model

- I/O model...but algorithms do not know *B* and *M*
- Assume optimal cache replacement strategy
- Optimal on all levels (under some assumptions)

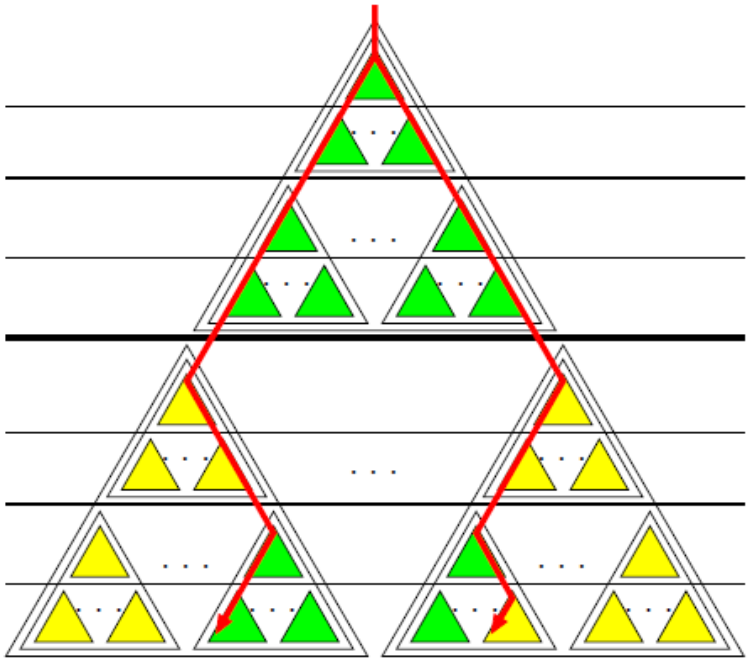


[FLPR12] Matteo Frigo, Charles E. Leiserson, Harald Prokop, Sridhar Ramachandran. Cache-Oblivious Algorithms. ACM Transactions on Algorithms, 8(1), Article No. 4, 2012.

Recursive Tree Layout (van Emde Boas layout)



Binary tree

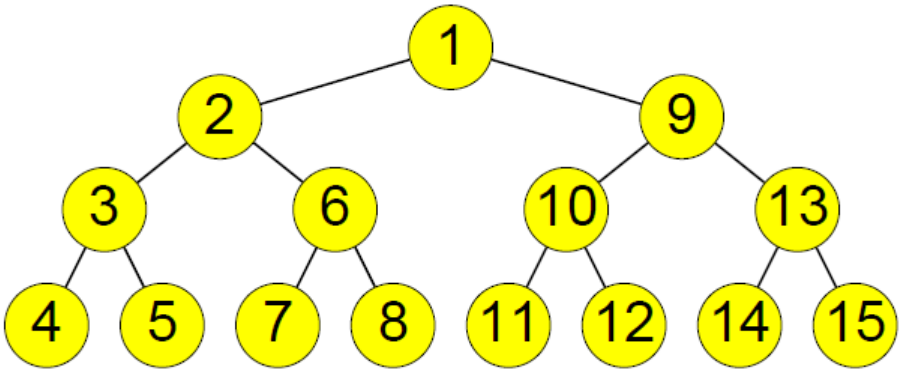


Searches O($\log_B N$) IOs

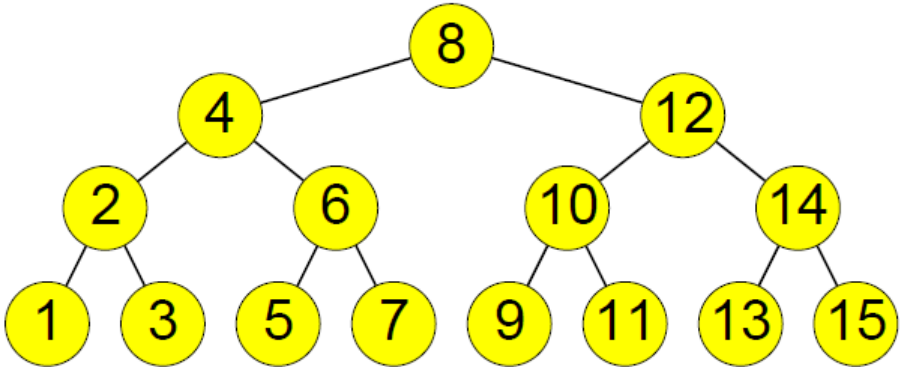
Range Searches O($\log_B N + k/B$)

Harald Prokop 1999, MIT MSc thesis "*Cache-Oblivious Algorithms*", June 1999

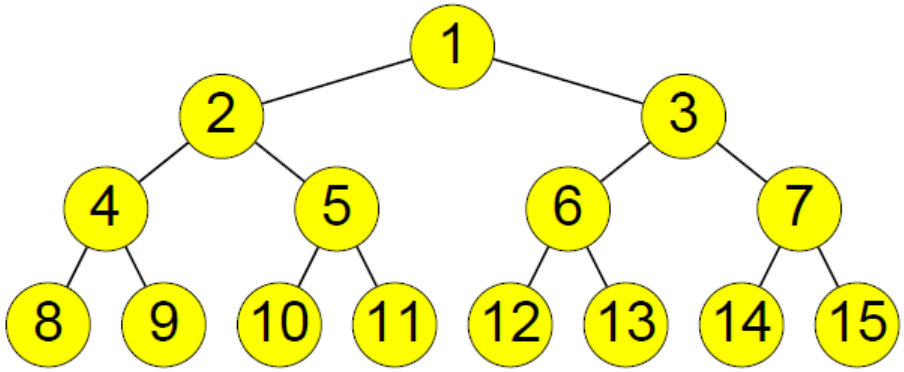
Four Tree Layouts



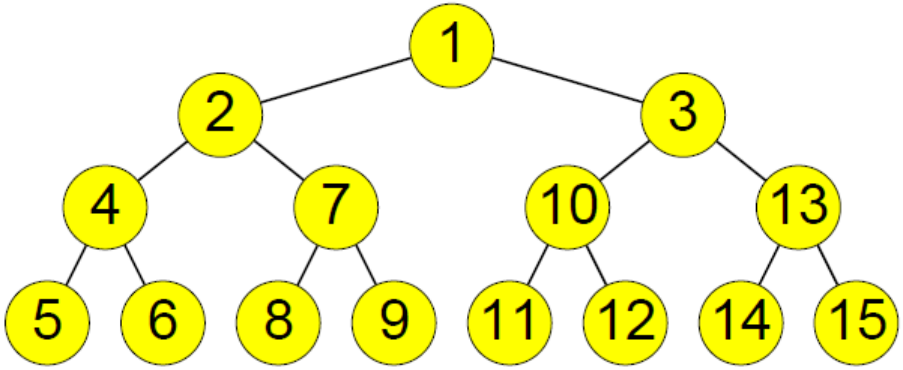
DFS



Inorder

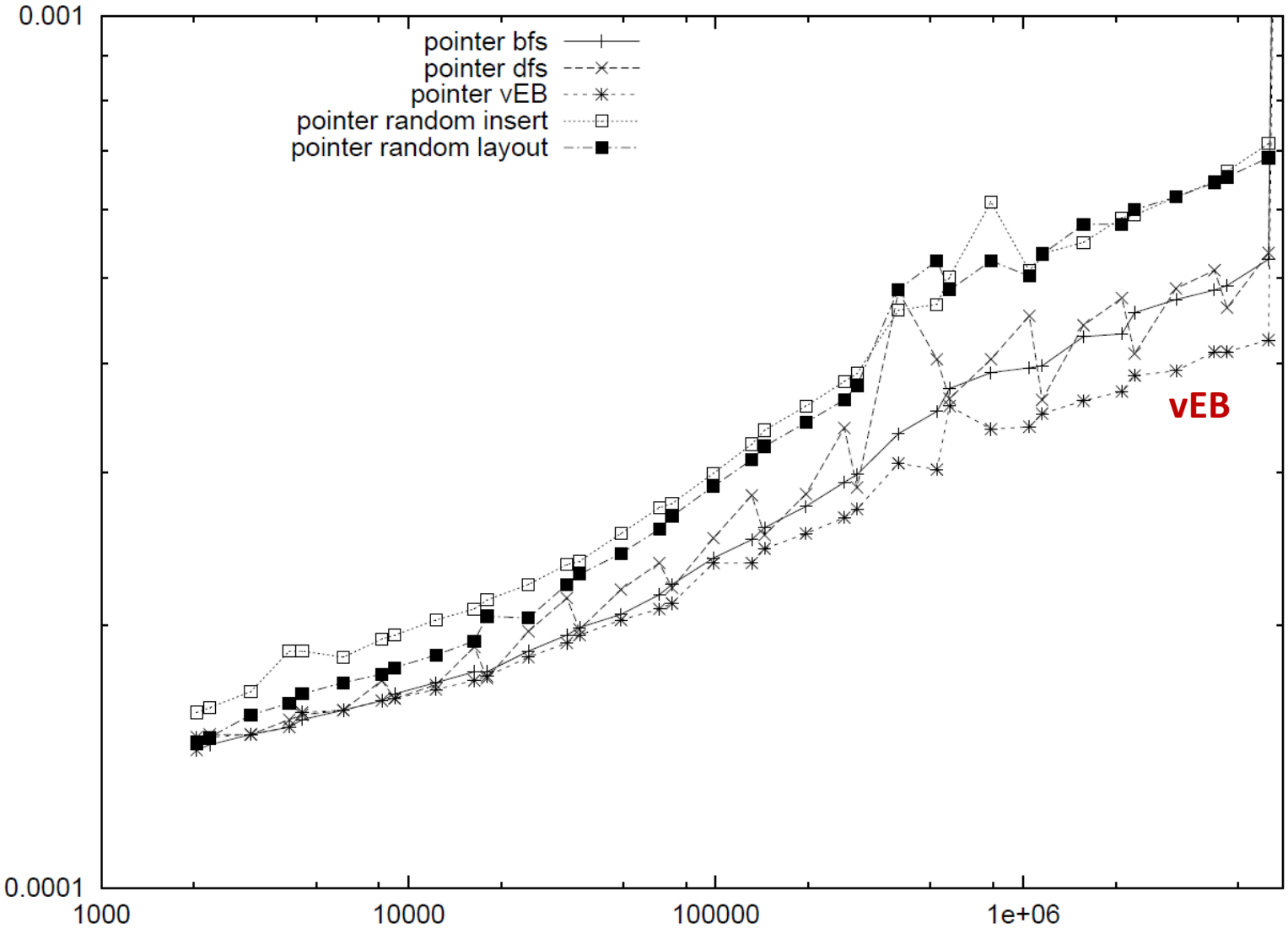


BFS

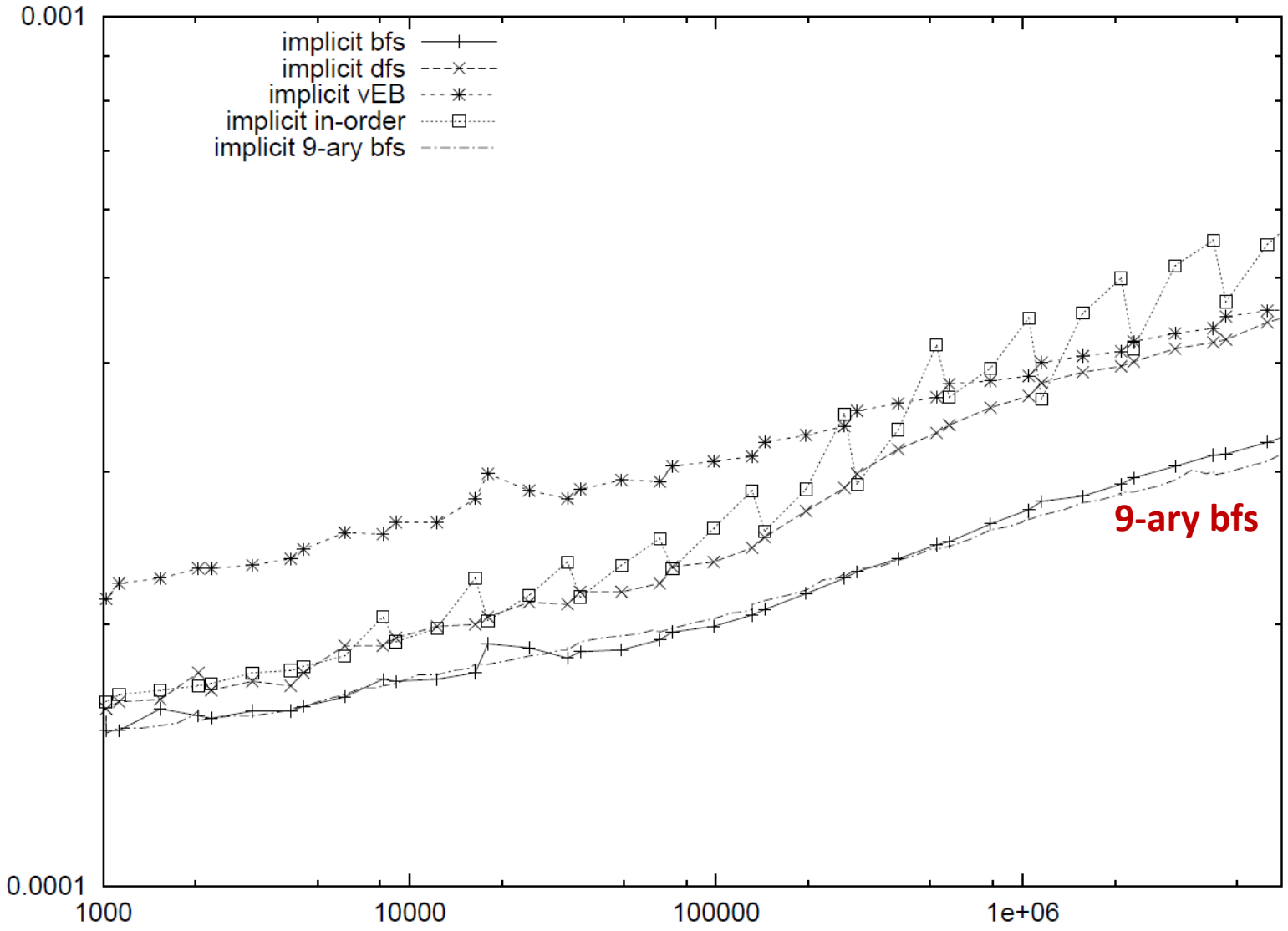


Recursive / van Emde Boas

Random Searches in Pointer Layouts



Random Searches in Implicit Layouts

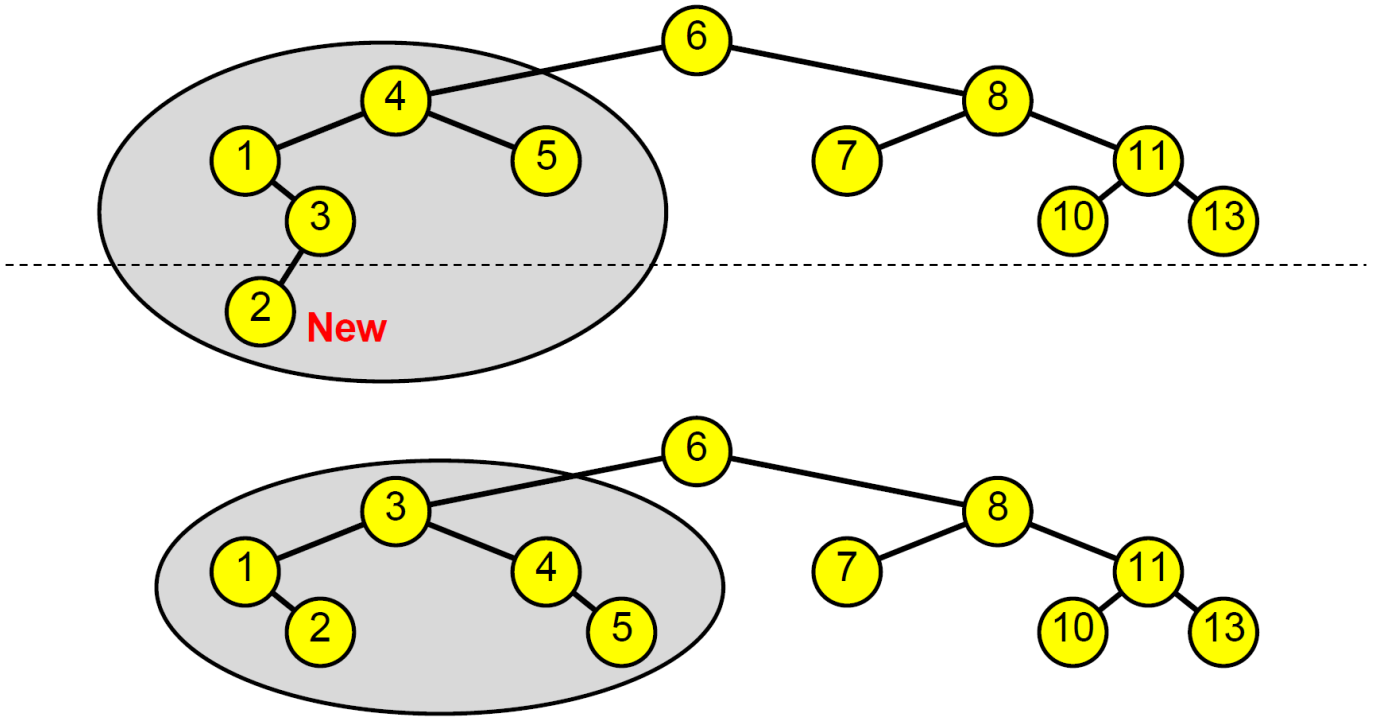


Making Trees Dynamic ?

- Trees of bounded depth

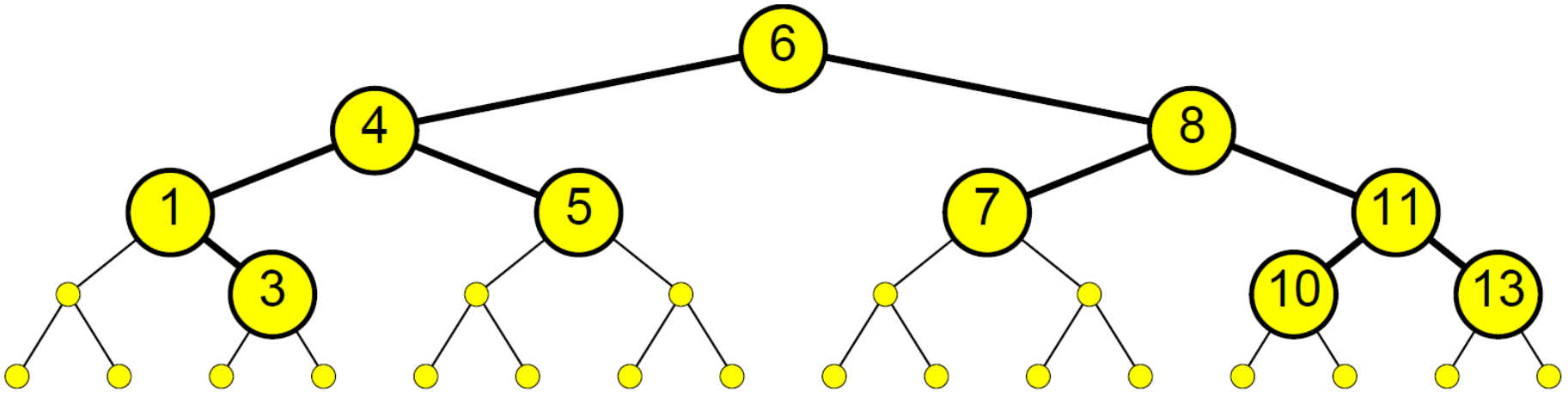
  Andersson and Lai 1990

- Rebuild subtrees when depth $\geq \log n + O(1)$
- Insert: $O(\log^2 n)$ amortized



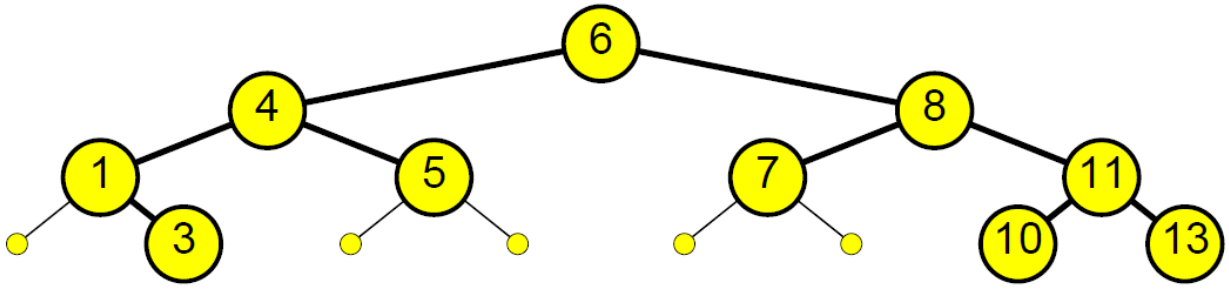
Static ⇒ Dynamic

- Emded dynamic tree into a complete tree
- Static layout of tree (e.g. van Emde Boas layout)



- Search $O(\log_B N)$
- Update $O(\log_B N + (\log^2 N)/B)$

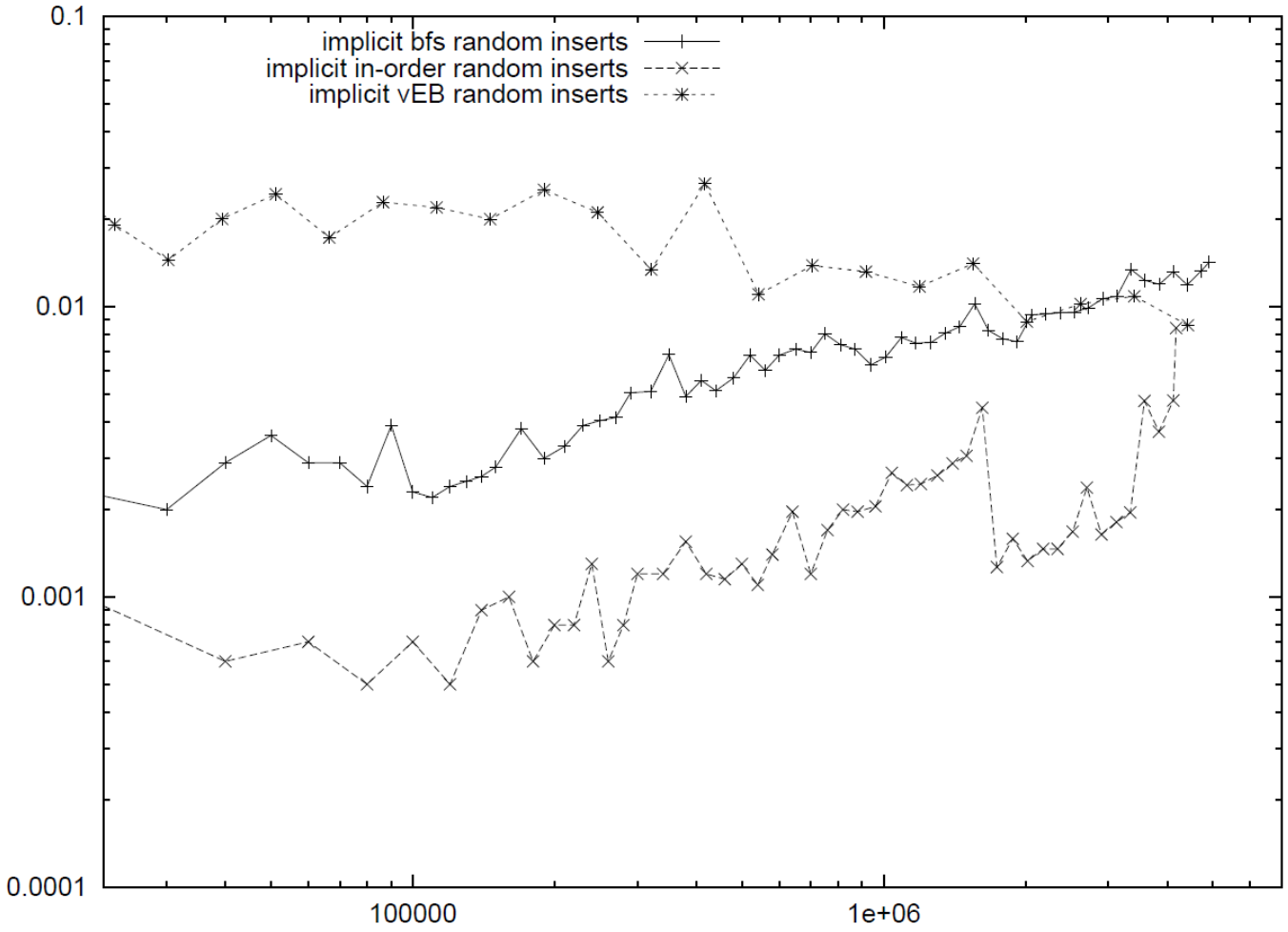
[BFJ02] Gerth Stølting Brodal, Rolf Fagerberg, Riko Jacob. Cache-Oblivious Search Trees via Binary Trees of Small Height. In Proc. 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 39-48, 2002.



6	4	8	1	–	3	5	–	–	7	–	–	11	10	13

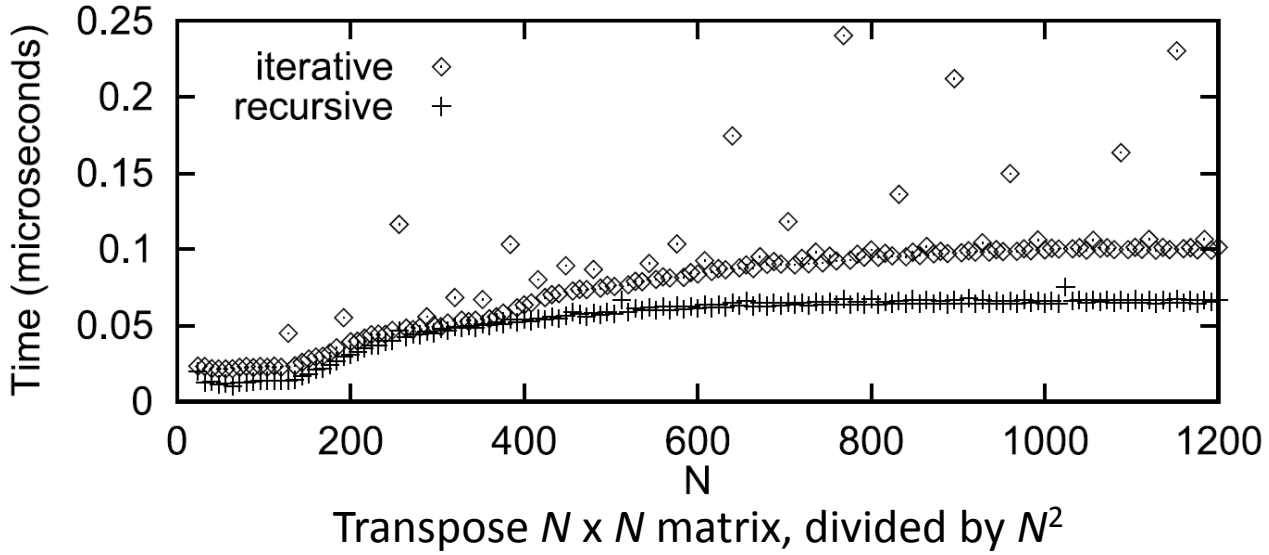
Search(9)

Insertions into Implicit Layout

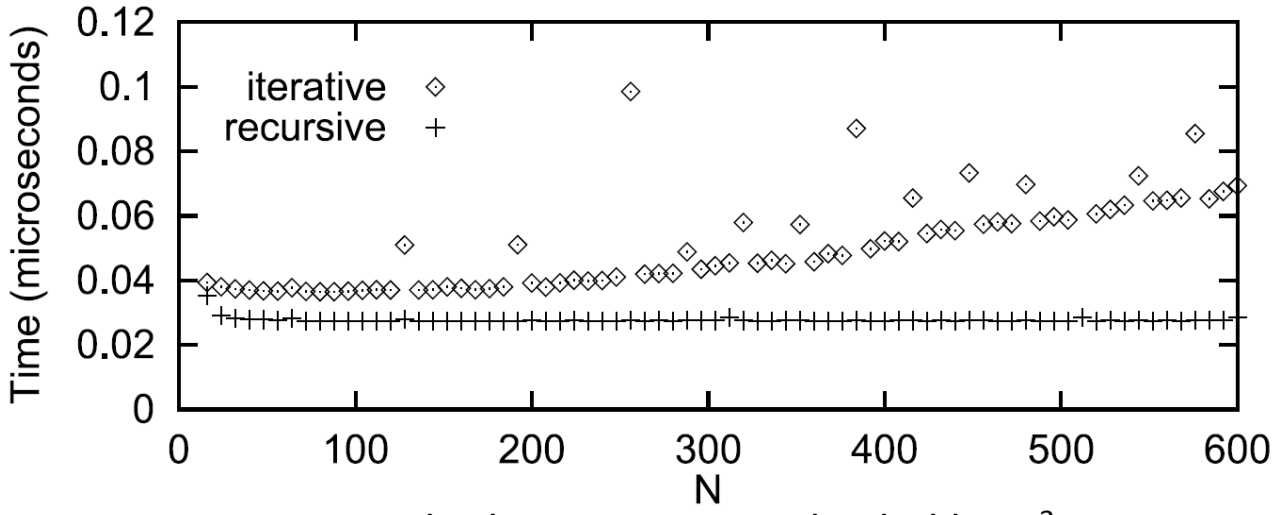


- Insertions factor 10-100 slower than searches

Matrix



Transpose N x N matrix, divided by N^2



Multiply N x N matrix, divided by N^3

[FLPR12] Matteo Frigo, Charles E. Leiserson, Harald Prokop, Sridhar Ramachandran. Cache-Oblivious Algorithms. ACM Transactions on Algorithms, 8(1), Article No. 4, 2012.