

A Forward Scan based Plane Sweep Algorithm for Parallel Interval Joins



AARHUS UNIVERSITY



Panagiotis Bouros¹ and Nikos Mamoulis²

¹Department of Computer Science, Aarhus University, Denmark

²Department of Computer Science & Engineering, University of Ioannina, Greece

pbour@cs.au.dk, nikos@cs.uoi.gr

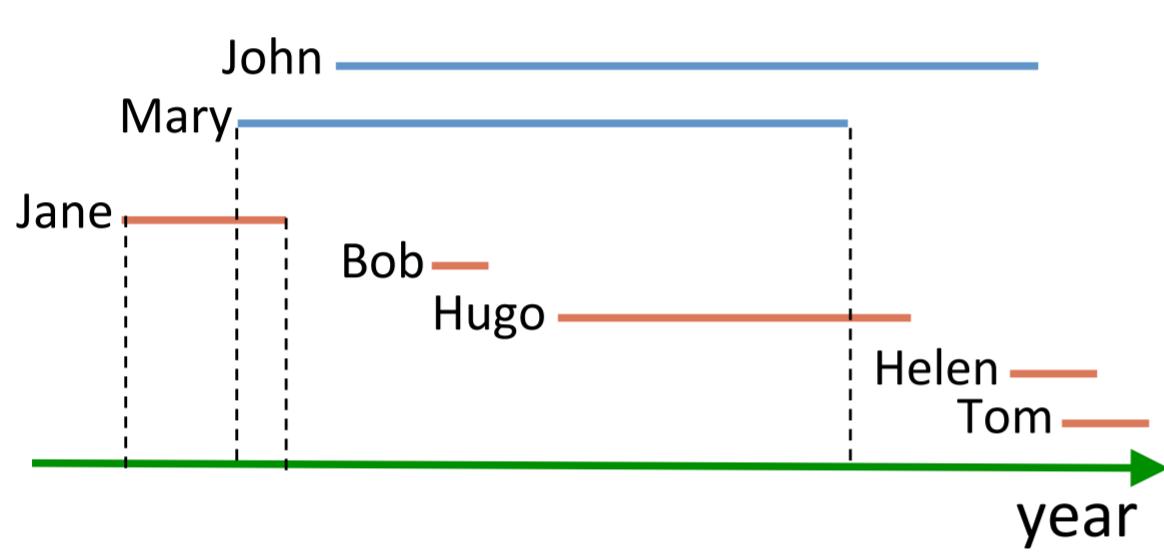
Interval Joins

Example

Find all pairs of employees whose working periods on departments D1 and D2 intersect

employee	start	end
John	1994	2006
Mary	1992	2002

employee	start	end
Jane	1990	1993
Bob	1995	1996
Hugo	1997	2003
Helen	2005	2007
Tom	2006	2008



Applications

- Temporal databases
- Multidimensional data management
- Uncertain data management

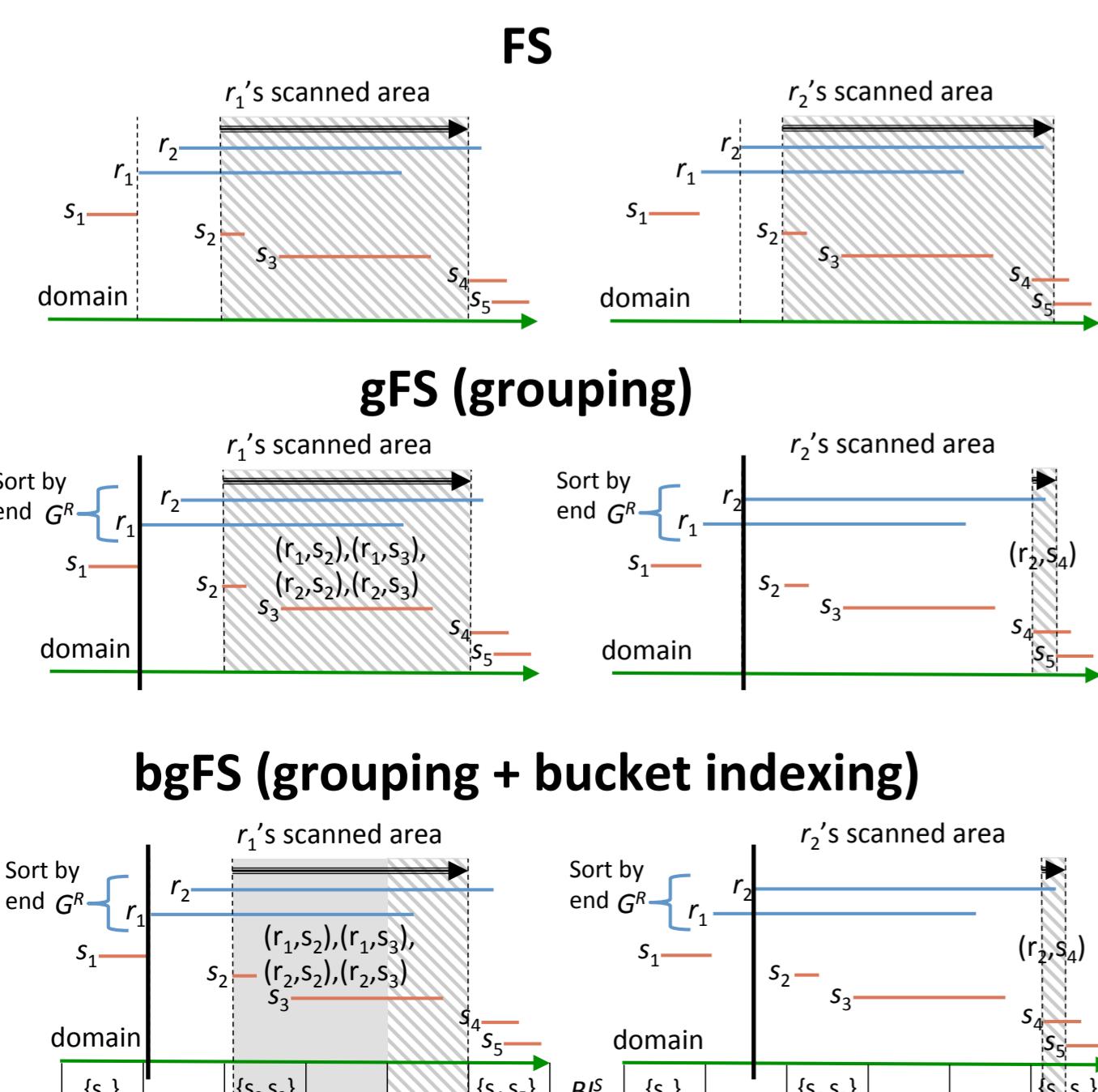
Single-threaded Processing

Related work

- Nested loops, sort-merge join
- Index-based
- Partitioning-based
 - ✓ OIP [3], DIP [2]
- Plane-sweep based
 - ✓ FS [1], EBI/LEBI [4]

Optimizing FS

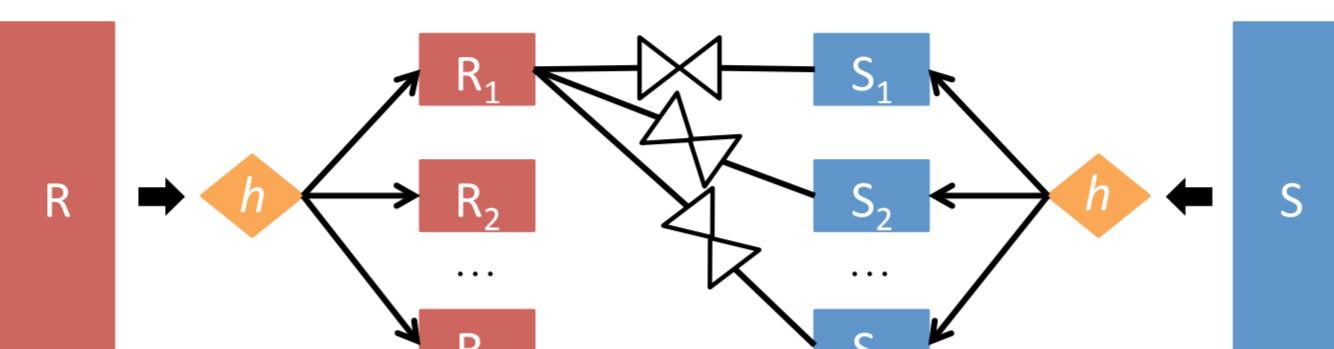
- Reduce comparisons to produce results
- Grouping
 - ✓ Group consecutive intervals from same input
 - ✓ Avoid redundant comparisons
- Bucket indexing
 - ✓ Produce results with no comparisons



Parallel Processing

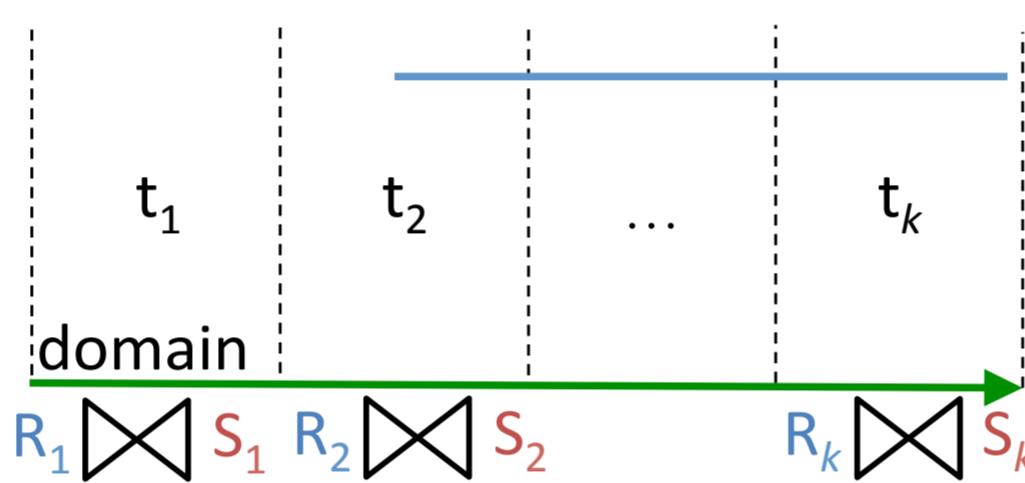
Hash-based Partitioning [4]

- Split inputs into k partitions using hashing
- Evaluate all pair-wise partition joins

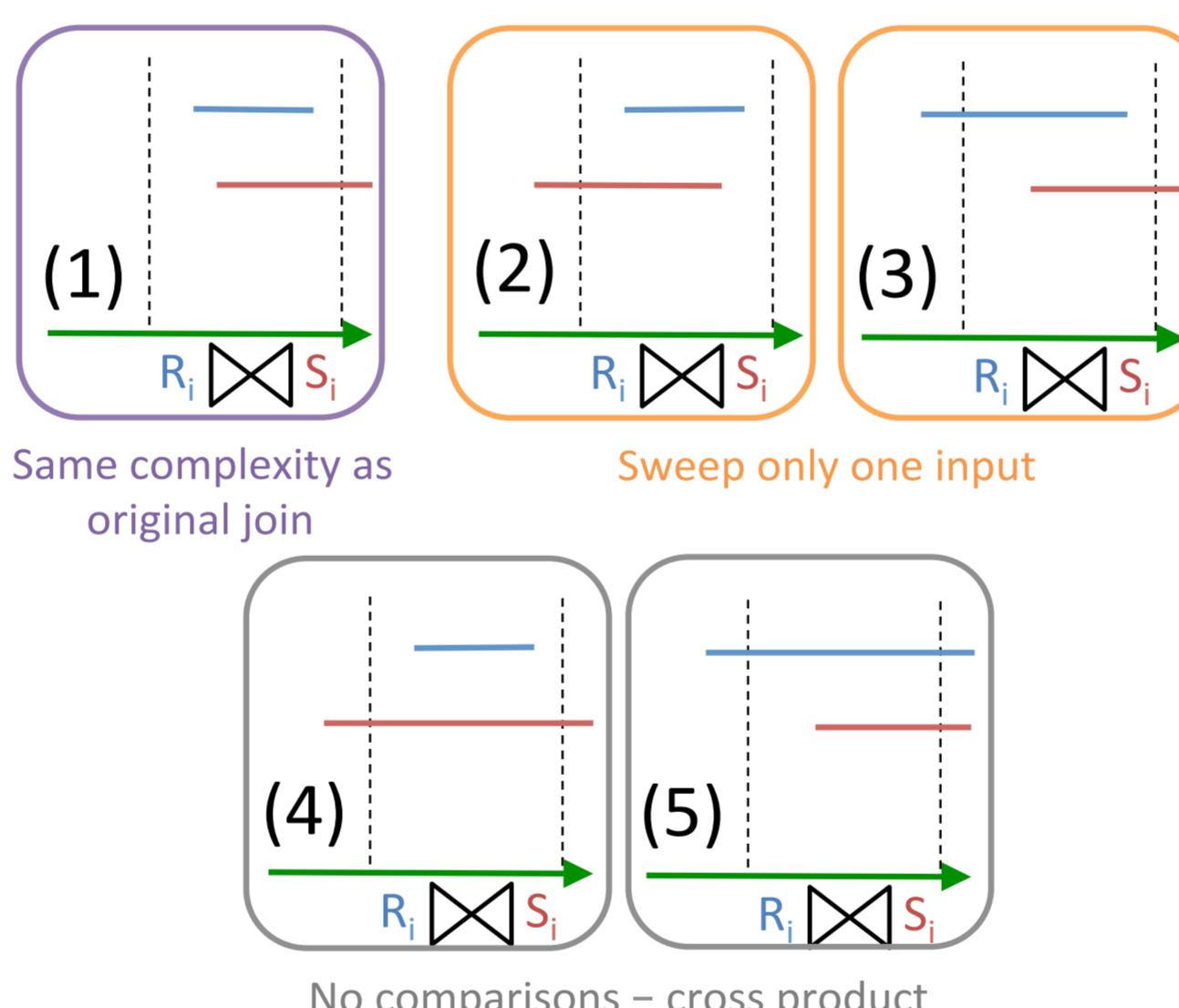


Domain-based Partitioning

- Split domain into k tiles
- Replicate intervals spanning multiple tiles
- Evaluate k independent partition joins
- Load balancing

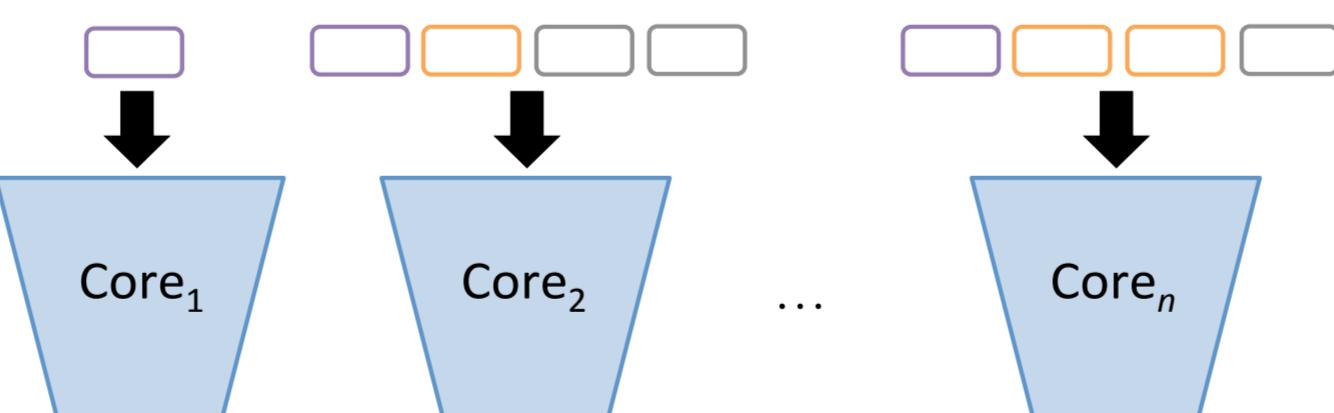


Mini-joins Breakdown



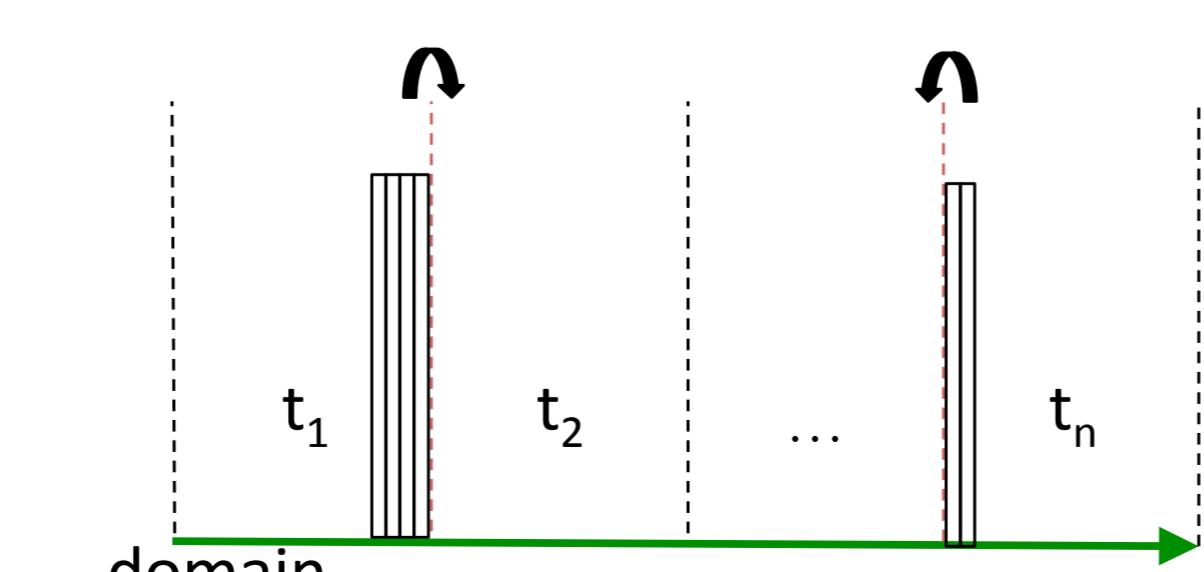
Greedy Scheduling

- Distribute mini-joins to different cores
- Minimize maximum load
- NP-hard assignment problem



Adaptive Partitioning

- Modify initial uniform partitioning
- Reposition tile borders
- Move load between neighboring tiles

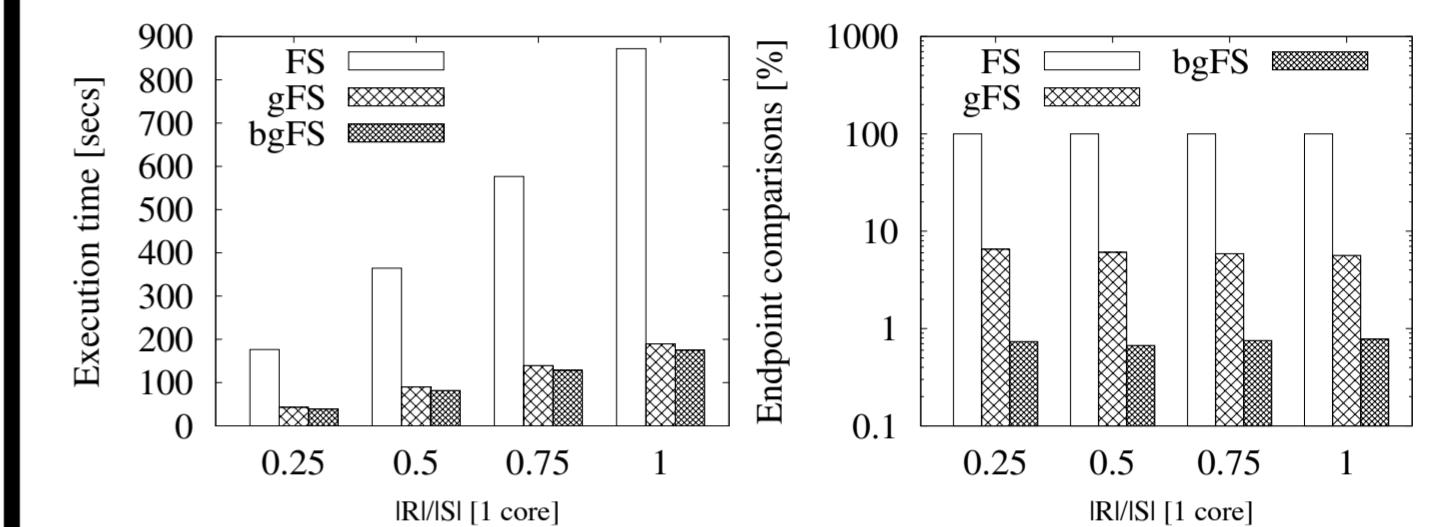


Experiments

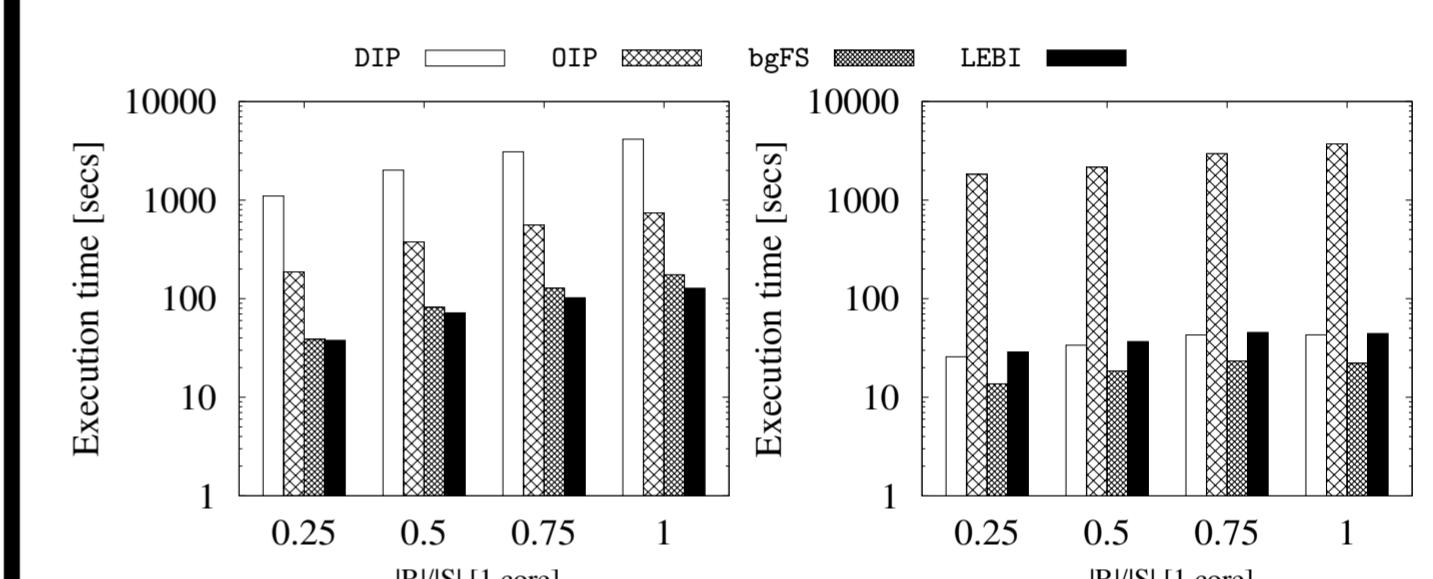
Setup

- In-memory processing
- Hyper-threading enabled, up to 40 threads
- Workload as [4], XOR of start attributes
- Loop unrolling forced, OpenMP

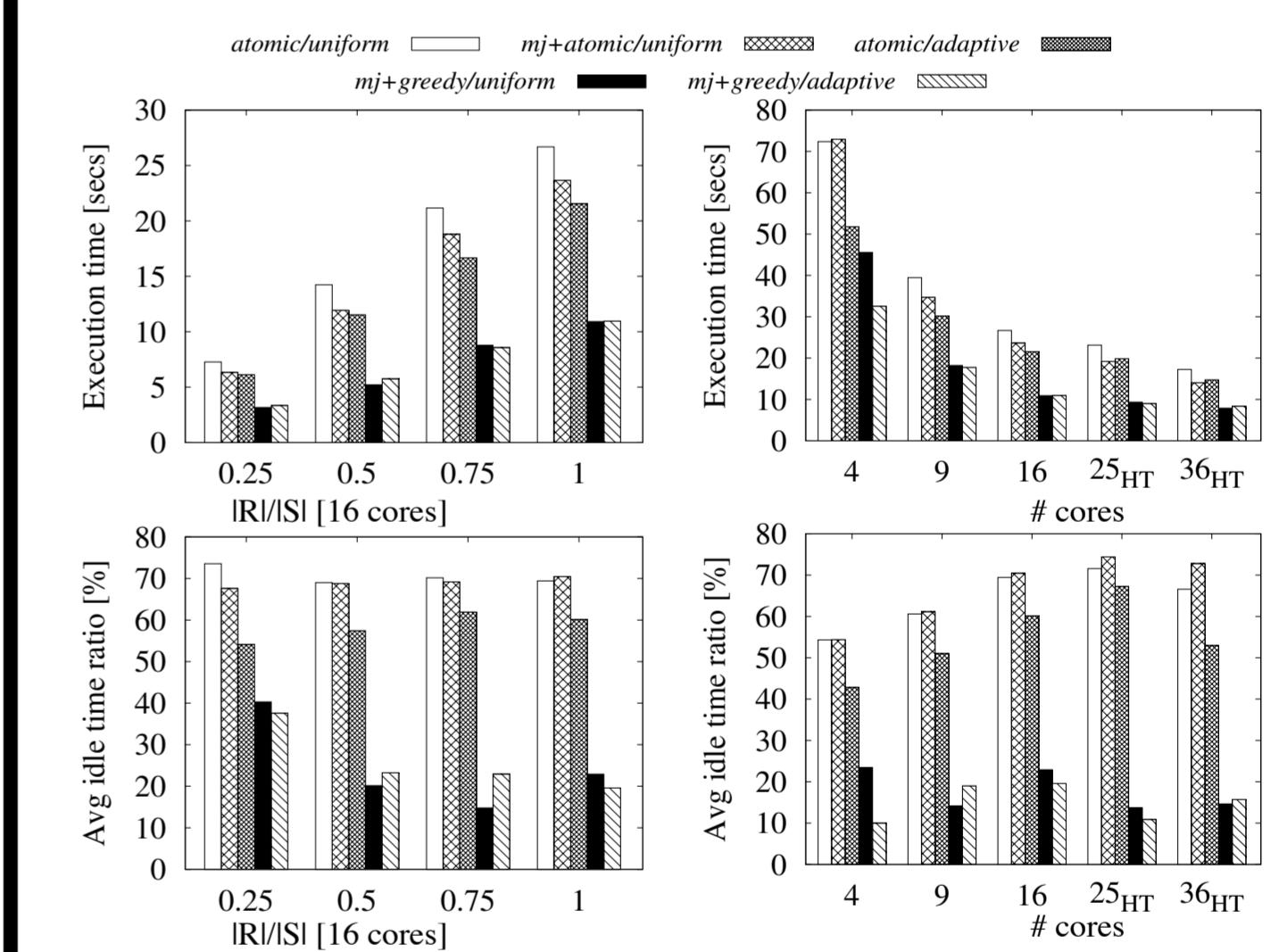
Optimizing FS



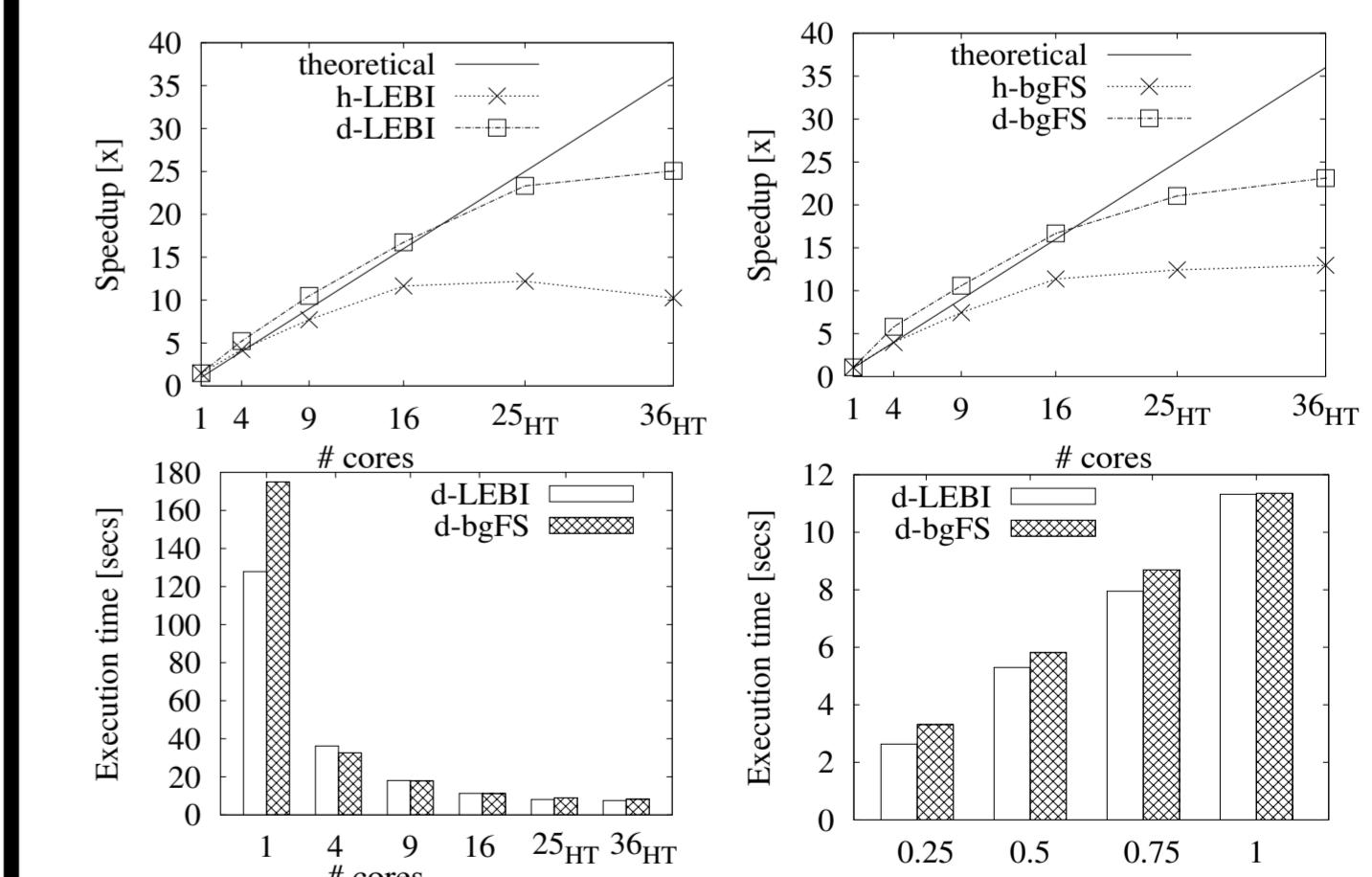
Single-threaded Processing



Optimizing Domain-based Partitioning



Parallel Processing



References

- [1] T. Brinkhoff, H.-P. Kriegel, and B. Seeger. Efficient processing of spatial joins using r-trees. In SIGMOD, 1993.
- [2] F. Cafagna and M. H. Böhlen. Disjoint interval partitioning. VLDB J., 26(3):447–466, 2017.
- [3] A. Dignös, M. H. Böhlen, and J. Gamper. Overlap interval partition join. In SIGMOD, 2014.
- [4] D. Piatov, S. Helmer, and A. Dignös. An interval join optimized for modern hardware. In ICDE, 2016.