

# Analysis of Multi-Join Benchmarking

Harshad Deshmukh  
University of Wisconsin-Madison  
harshad@cs.wisc.edu

Jignesh M. Patel  
University of Wisconsin-Madison  
jignesh@cs.wisc.edu

## ABSTRACT

### 1. INTRODUCTION

Intro goes here.

### 2. EXPERIMENTAL EVALUATION

Evaluation description goes here.

### 3. CONCLUSIONS

Conclusions go here.

### 4. RELATED WORK

The database community has paid significant attention to improve join performance in the past decade. Blanas et al. [1] analyzed the join performance in single socket, large main-memory server environments using various join implementations such as radix-based partitioning, no-partitioned join and using uniform as well as skewed data sets.

Schuh et al. [2] reviewed various join implementations from the research literature from the past decade. They classified the join algorithms in two categories - partitioning based and no-partitioning based join algorithms, and compared their performance characteristics. As they observed,

prior work on join performance has mostly focused on synthetic queries and synthetic datasets. In the interest of discussing the join algorithm performance in the context of real life queries, they measured the performance of various join algorithms on a real query (TPC-H Query 19).

### 5. REFERENCES

- [1] S. Blanas, Y. Li, and J. M. Patel. Design and evaluation of main memory hash join algorithms for multi-core cpus. In *SIGMOD*, pages 37–48, 2011.
- [2] S. Schuh, X. Chen, and J. Dittrich. An experimental comparison of thirteen relational equi-joins in main memory. In *SIGMOD*, pages 1961–1976, 2016.