Triton Join: Efficiently Scaling to a Large Join State on GPUs with Fast Interconnects

Clemens Lutz[1], Sebastian Breß[2], Steffen Zeuch[1], Tilmann Rabl[3], Volker Markl[1]

Technische Universität Berlin · German Research Center for Artificial Intelligence · HPI Hasso Plattner Institut, IT Systems Engineering | Universität Potsdam

[1]firstname.lastname@tu-berlin.de
[2]sebastian.bress@snowflake.com
[3]tilmann.rabl@hpi.de

Overview

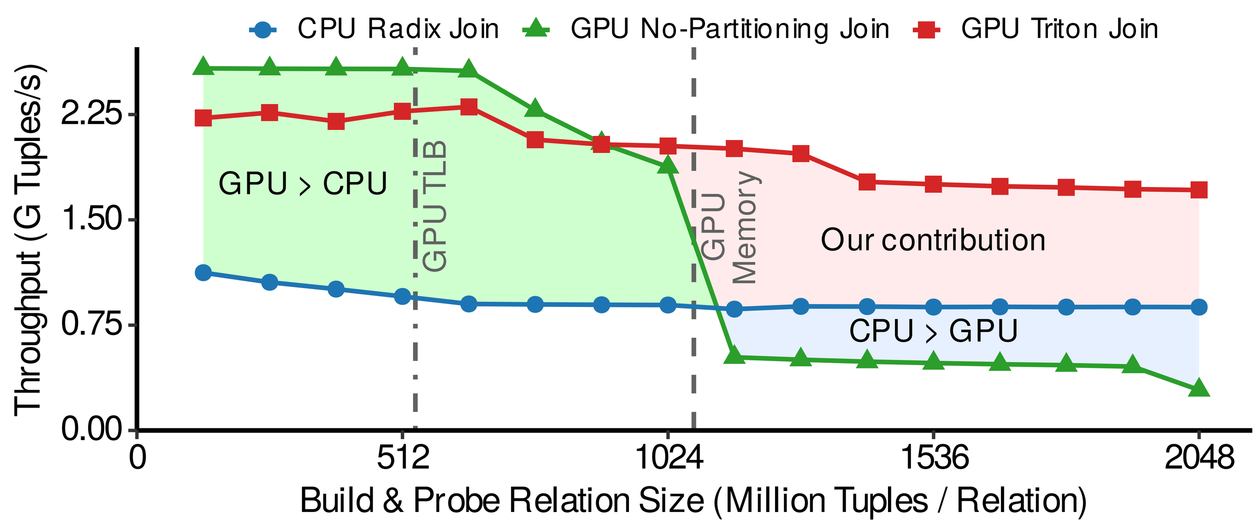
GPUs are well-equipped to quickly process joins and other stateful operators due to their high memory bandwidth.

However, GPUs do not scale to large joins because:

- large join state does not fit into GPU memory
- spilling state to main memory is constrained by interconnect bandwidth.

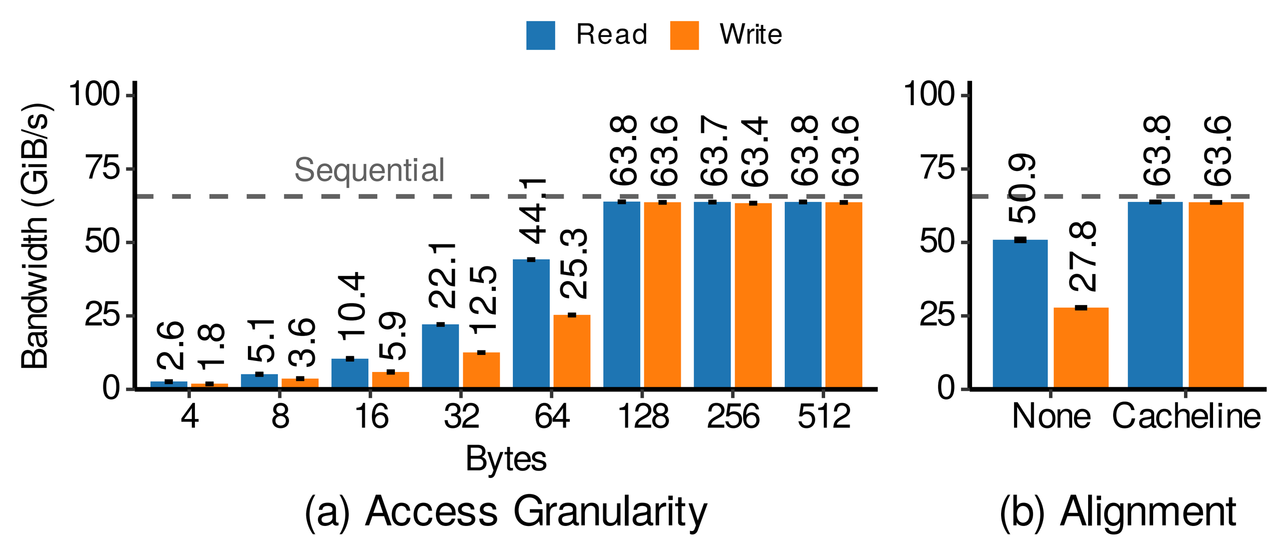
We propose a new join algorithm that scales to large data volumes by exploiting fast interconnects, e.g., NVLink.

Goal: Scalable Join Processing



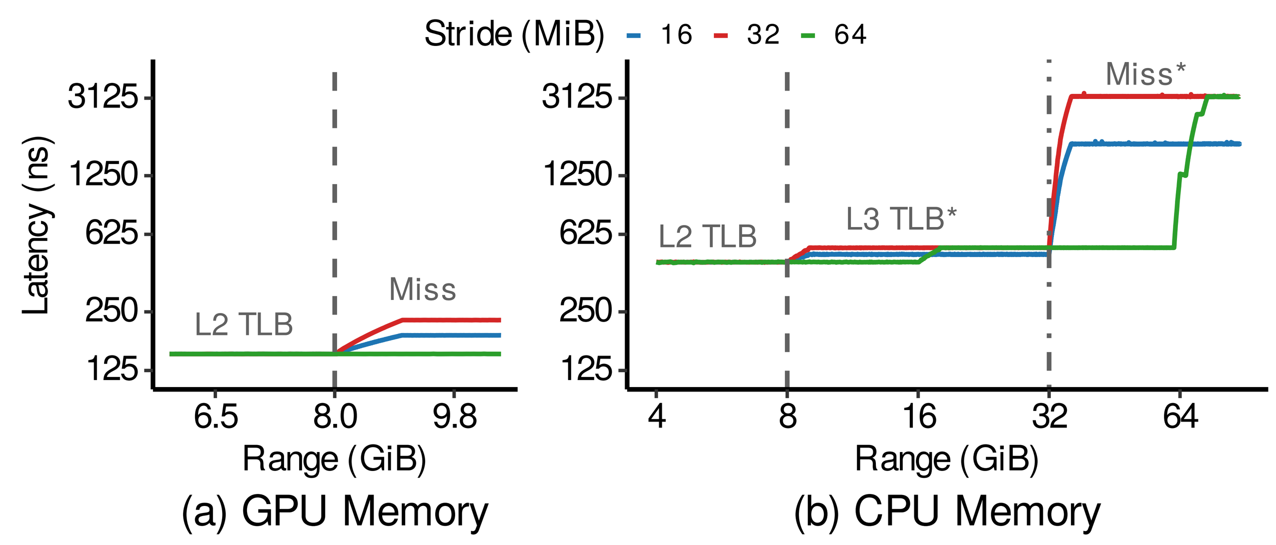
Out-of-core join state results in a performace cliff and slow-down, despite using a fast interconnect.

Problem 1: Transfer Granularity



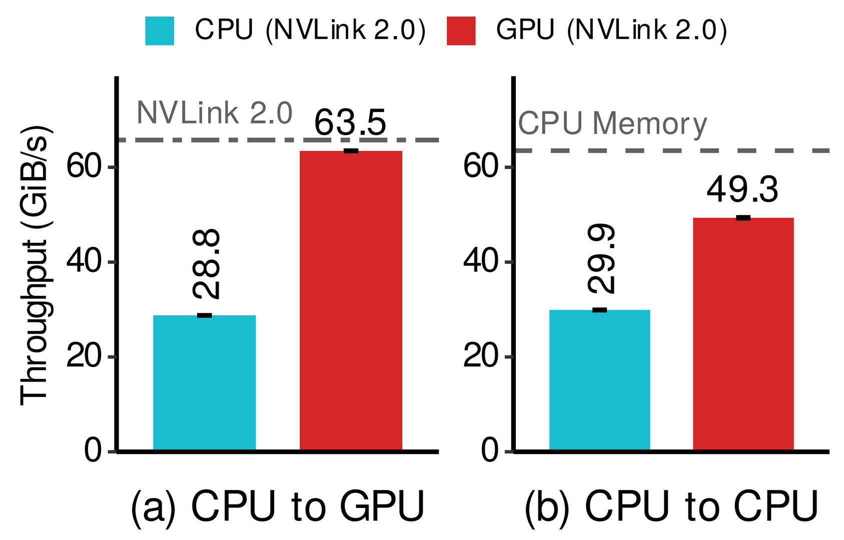
Fine-grained, random accesses to main memory are slow. However, cacheline-sized accesses are fast!

Problem 2: IO TLB misses



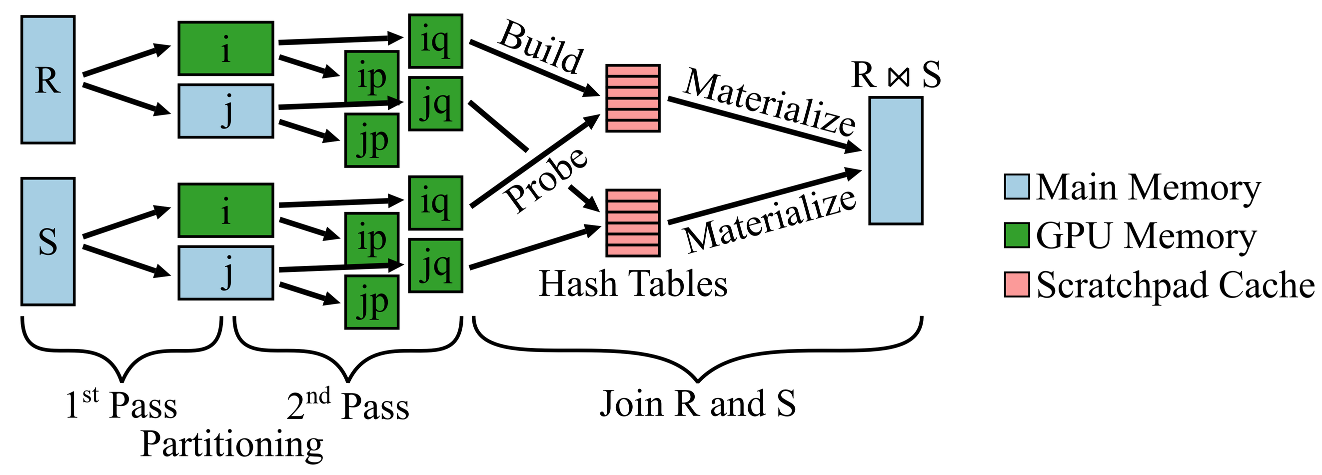
IO TLB misses slow down accesses to main memory by one order-of-magnitude.

Out-of-Core Radix Partitioning using a GPU



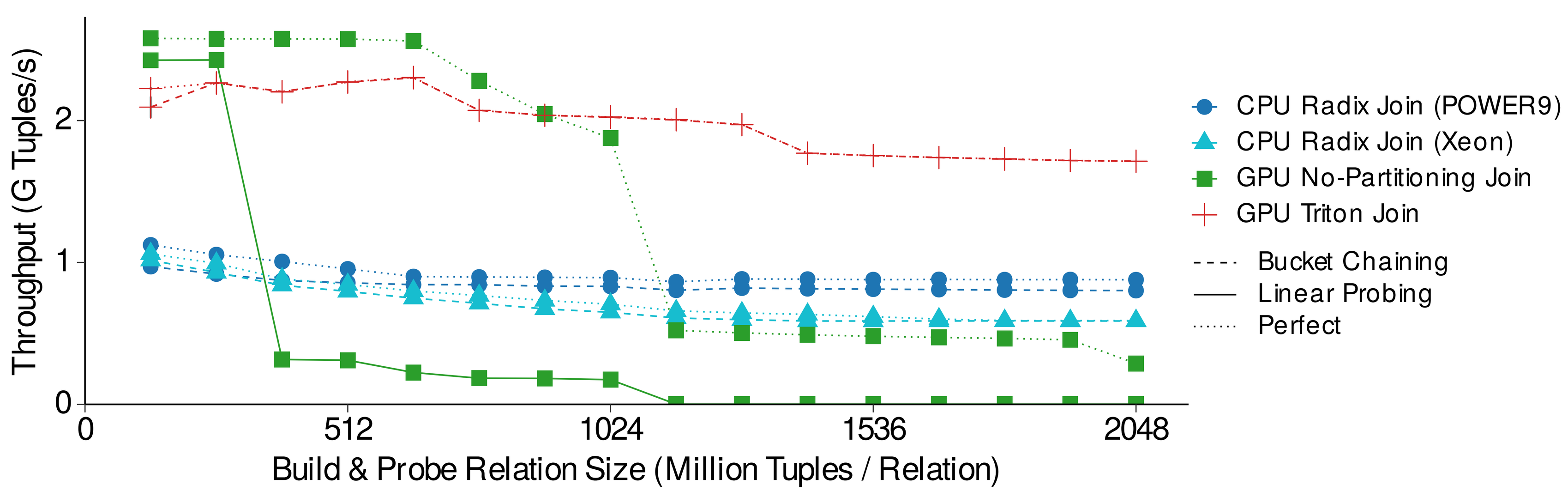
Partitioning is faster on a GPU with a fast interconnect than on a CPU.

The Triton Join Algorithm



Take advantage of data locality by two-pass radix partitioning and in-GPU partition caching.

Scaling to a Large, Out-of-Core Join State



Triton join achieves 1.9–2.6× speedup over CPU and up to 400× over no-partitioning hash join on same GPU.

Take Home

- **Scalable due to spilling join state to main memory via a fast interconnect.**
- **Robust due to graceful performance degradation under an increasing join state size.**
- **Efficient due to offloading nearly all processing from the CPU to the GPU.**

Preprint is available!

www.clemenslutz.com

Funding Acknowledgements

This work was funded by the EU Horizon 2020 programme as E2Data (780245), the German Ministry for Education and Research as BIFOLD — “Berlin Institute for the Foundations of Learning and Data” (01IS18025A and 01IS18037A), and the German Federal Ministry for Economic Affairs and Energy as Project ExDra (01MD19002B).

BIFOLD · Federal Ministry for Economic Affairs and Climate Action · Federal Ministry of Education and Research · European Commission