Sparse Matrix to Matrix Multiplication:

A Representation and Architecture for Acceleration

Pareesa A. Golnari
Sharad Malik

1. Proposing InCRS to accelerate random data access in **sparse** formats

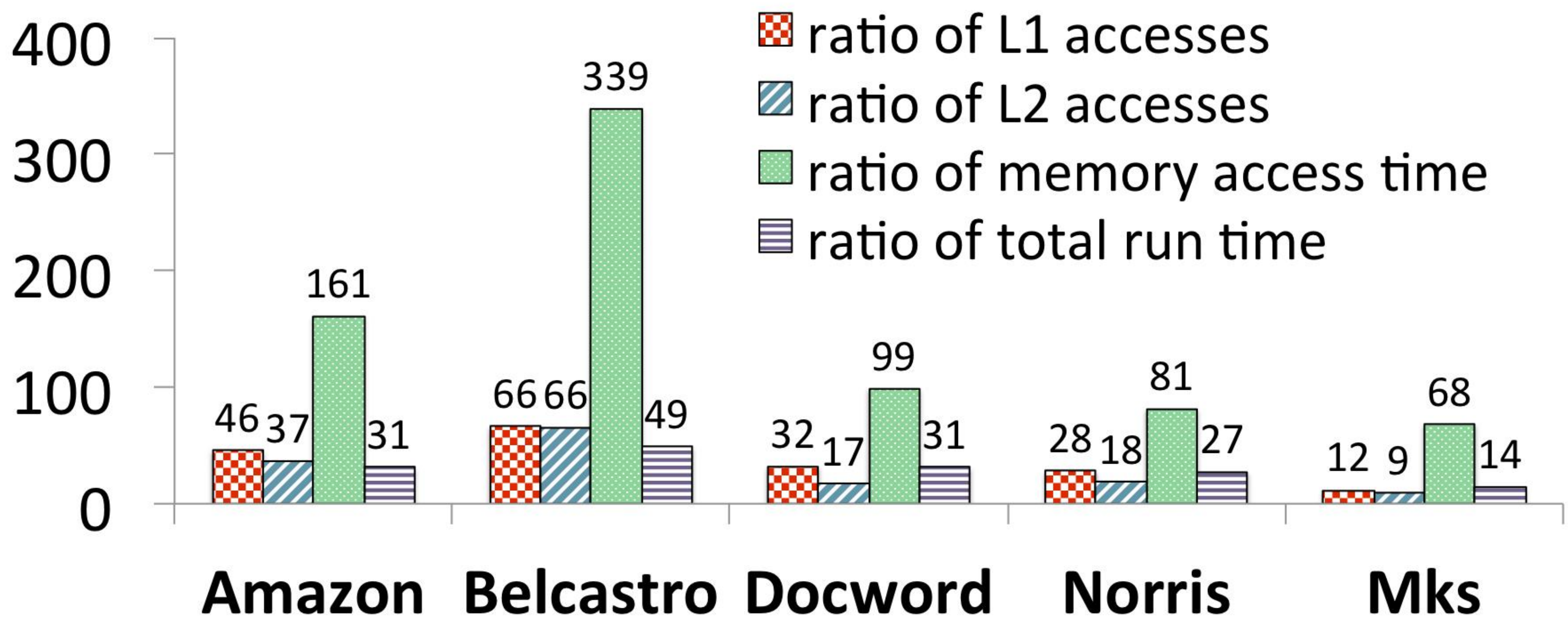
2. Proposing a high performance systolic **SpMM** architecture

1. Proposing InCRS to accelerate random data access in **sparse** formats

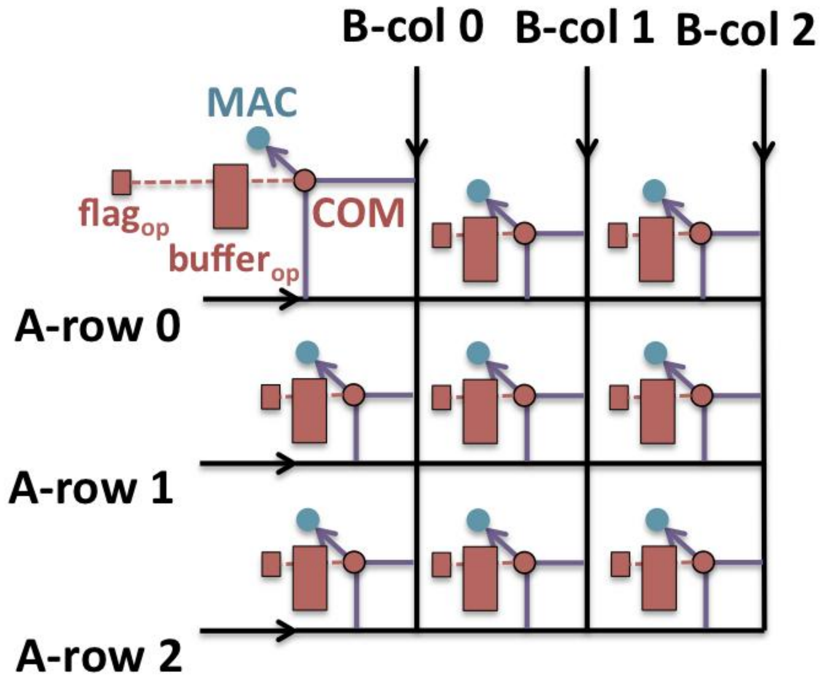
   a. Sparse formats store data in one order
   b. Accessing data in the second/random order is non-trivial
   c. A x **B** and **B** x C can both appear in the same program:
      i. A x **B** → **B** is accessed in column order. But,
      ii. **B** x C → **B** is accessed in row order
   d. *How to accelerate column-order access of **B** if it is stored in row-order?*

Sparse Matrix to Matrix Multiplication: A Representation and Architecture for Acceleration

- InCRS augments CRS by storing info about NZ distribution of the data.
- 68~339 times memory access reduction
- 14~49 times Spmm Acceleration



2. Proposing a high performance systolic **SpMM** architecture
   - Adopting systolic MM architecture
   - Synchronous movement of data through a mesh of comparators
   - Buffering the unmatched operand

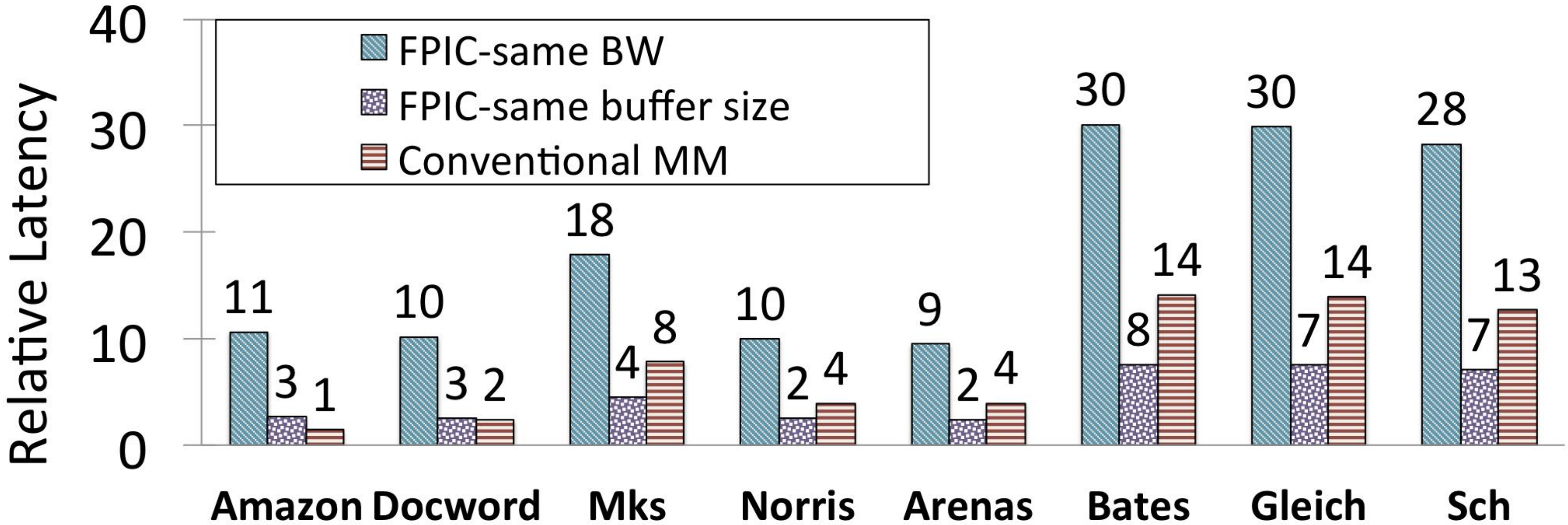


2. Proposing a high performance systolic **SpMM** architecture
    - Adopting systolic MM architecture
    - Synchronous movement of data through a mesh of comparators
    - Buffering the unmatched operand

        - Fast **operand consumption**
        - → High **data reuse**
        - **Scalability**

Sparse Matrix to Matrix Multiplication:
A Representation and Architecture for Acceleration

- Performed A x A^T on datasets:
  - Size: 1k x 1k to 7.5k x 7.5k
  - Density: 0.057%~14%
- **9~30** times SpMM acceleration compared to state-of-the-art (FPIC)



Sparse Matrix to Matrix Multiplication:
A Representation and Architecture for Acceleration

- Benefits:
  - High throughput
  - Reduced BW requirement
- Cost: extra buffering
- For instance: 64 x 64 MM with same BW requirement

→ 3~8 times acceleration at the cost of 768kB buffers (FPIC needs 192KB)