Matrix Multiplications: A Lot of Ways of Handling a Few Nonzeros

Karl Rupp

Short-Term Visiting Scientist
Argonne National Laboratory

extending previous work at TU Wien with
A. Morhammer, F. Rudolf, and J. Weinbub

LANS Informal Seminar
September 27, 2017

Introduction

Positions

PhD student at TU Wien (2009-2011)

Postdoc at Argonne Natl. Lab. (09/2012-09/2013)

Postdoc at TU Wien (09/2013-10/2015; 01/2017-)

Freelancer, ETH Zürich (03/2016-06/2017)

Research Interests

Numerical solution of PDEs

Semiconductor device simulation

Parallel computing

Software Development

PETSc

ViennaCL

ViennaSHE

...

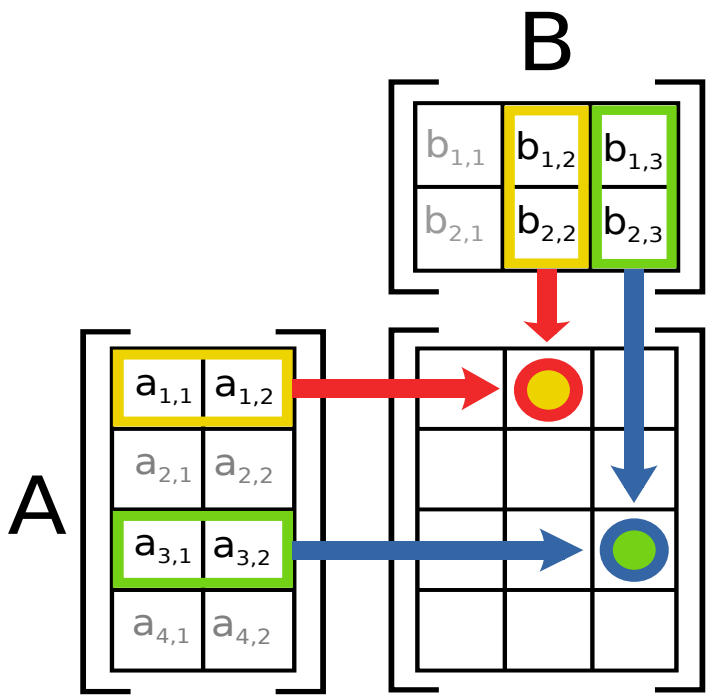
Matrix Multiplication

Dense Matrix-Matrix-Multiplications

Ubiquituous for: dense linear algebra (eigenvalues, LU factorization, etc.)

FLOP-limited, basis for TOP500

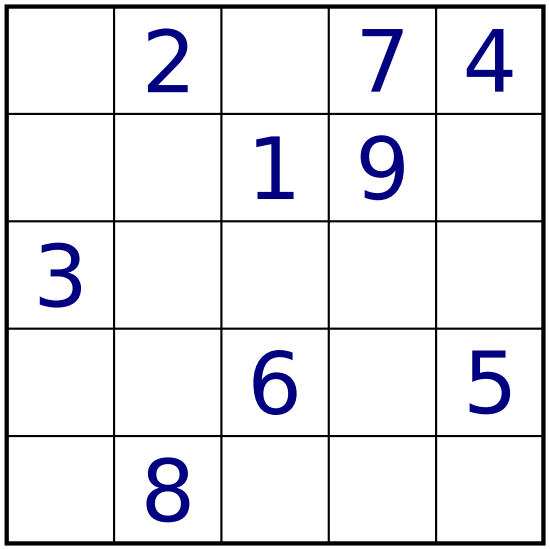
Computer scientist's darling



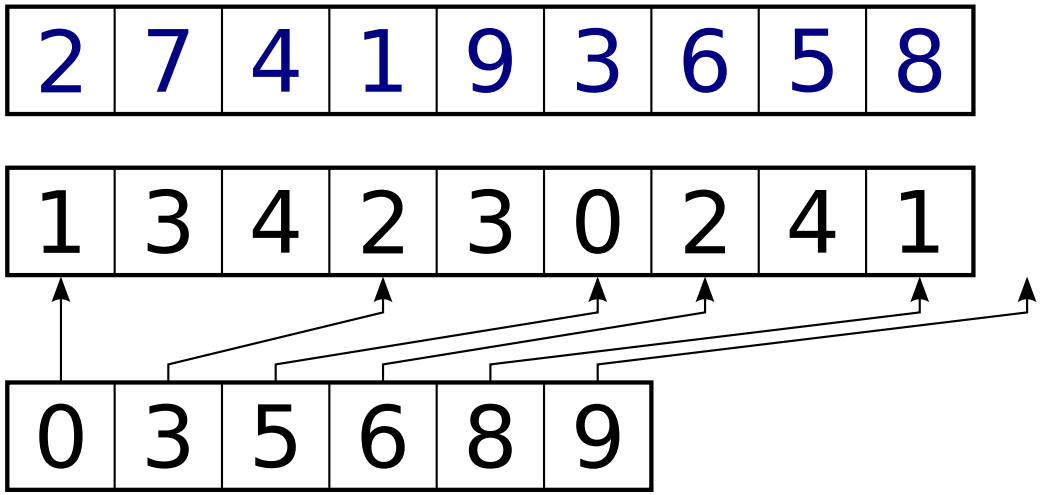
https://en.wikipedia.org/wiki/Matrix_multiplication#/media/File:Matrix_multiplication_diagram_2.svg

Compressed Sparse Row Format

Sparse Matrix Storage

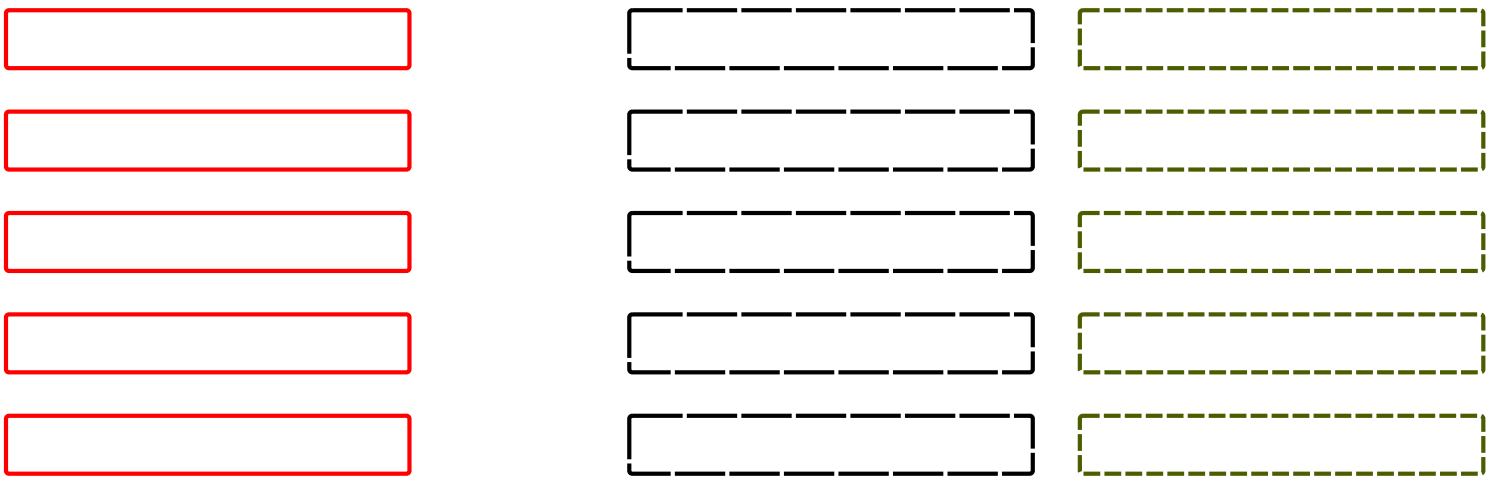


Matrix



CSR

Sparse Matrix Products

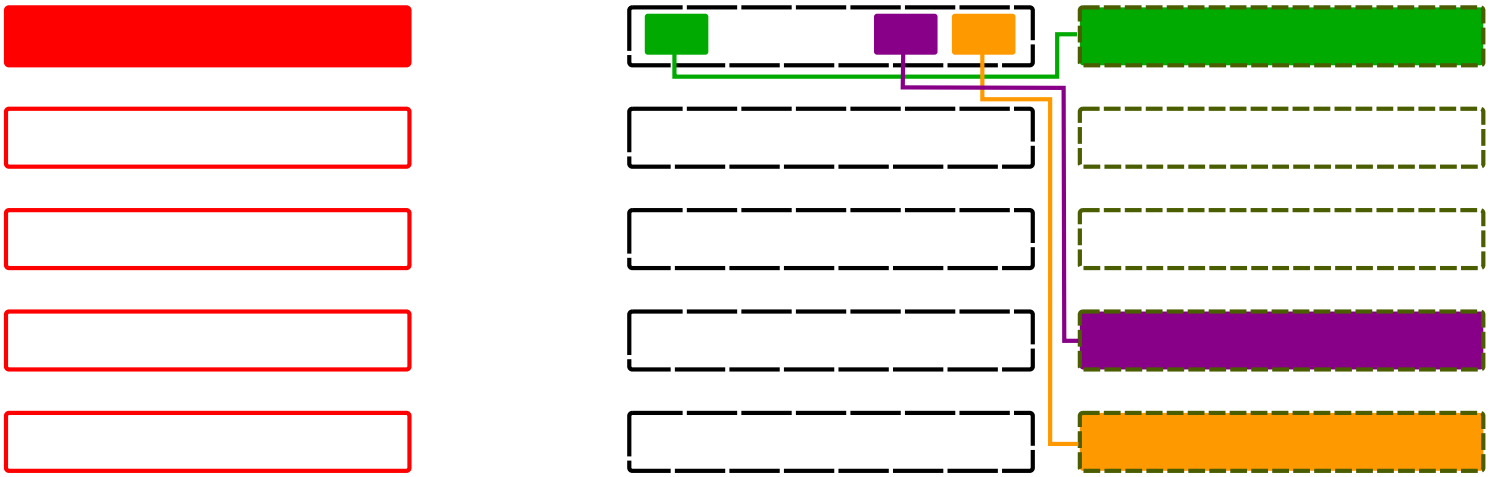


C = **A** **B**

Sparse Matrix Products

Row i : $\mathbf{c}_i = \sum_j a_{ij} \mathbf{b}_j$

Sparse Matrix Products



Row i : $\mathbf{c}_i = \sum_j a_{ij} \mathbf{b}_j$

Sparse Matrix-Matrix Multiplications

Sequential MatMatMult in PETSc

- Sorted
- Scalable
- Scalable, fast
- Heap
- BTHeap
- Linked-list condensed
- RowMerge – NEW!
- RowMerge2 – NEW!

Sparse Matrix-Matrix Multiplications

Sequential MatMatMult in PETSc

- Sorted
- Scalable
- Scalable, fast
- Heap
- BTHeap
- Linked-list condensed
- RowMerge – NEW!
- RowMerge2 – NEW!

Two Stages

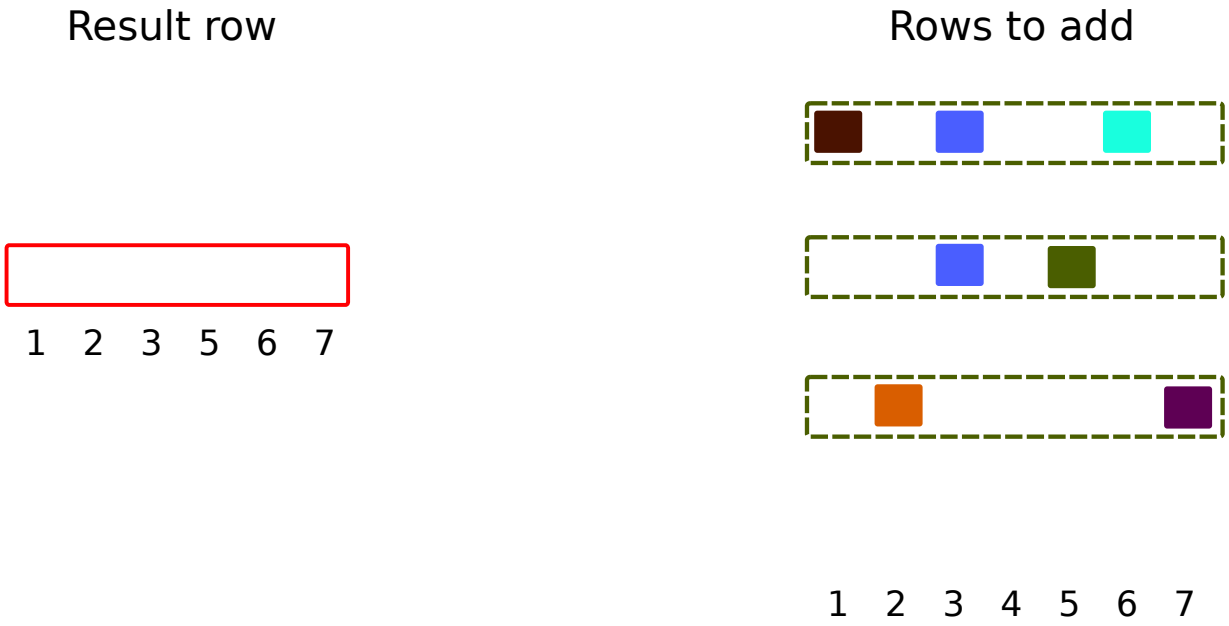
- Symbolic phase: Determine sparsity pattern
- Numeric phase: Compute numerical values, sparsity pattern known

Sparse Matrix-Matrix Multiplications

Good News:

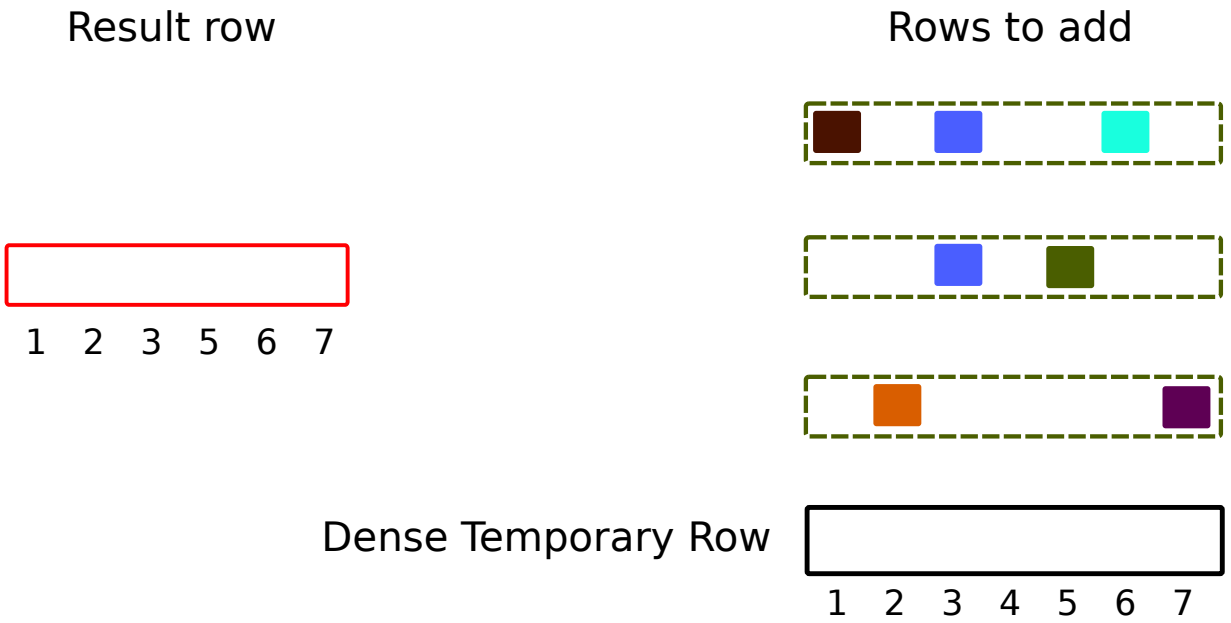
Numeric phase is easy!

Sparse Matrix-Matrix Multiplications



Numeric Phase

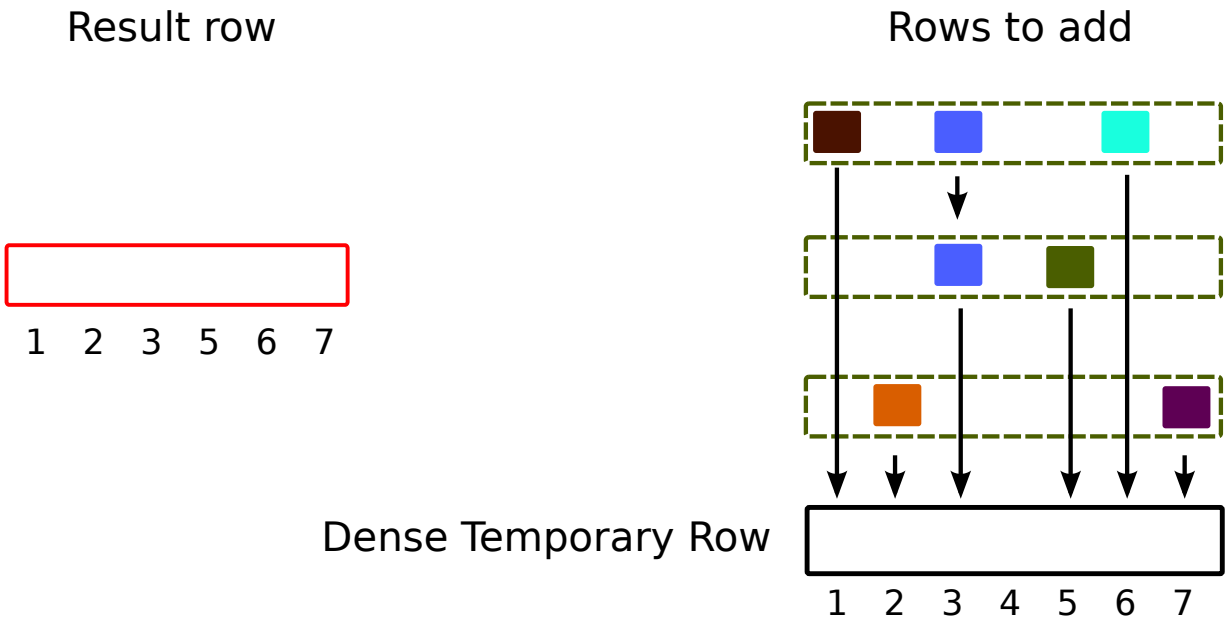
Sparse Matrix-Matrix Multiplications



Numeric Phase

Merge directly to dense array

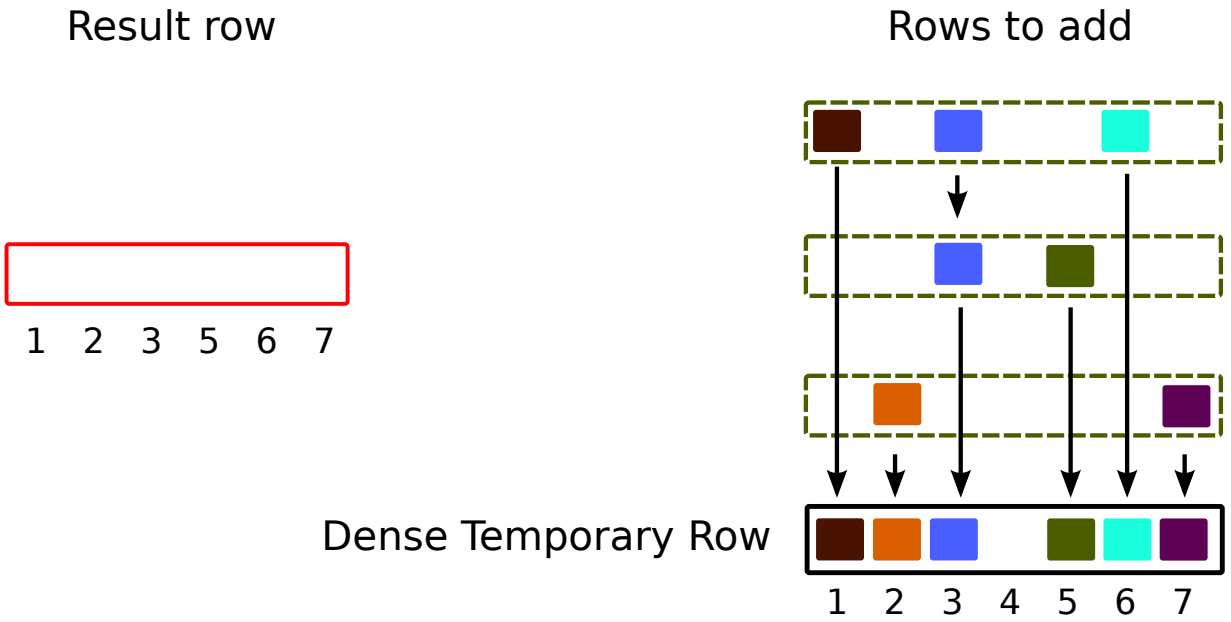
Sparse Matrix-Matrix Multiplications



Numeric Phase

Merge directly to dense array

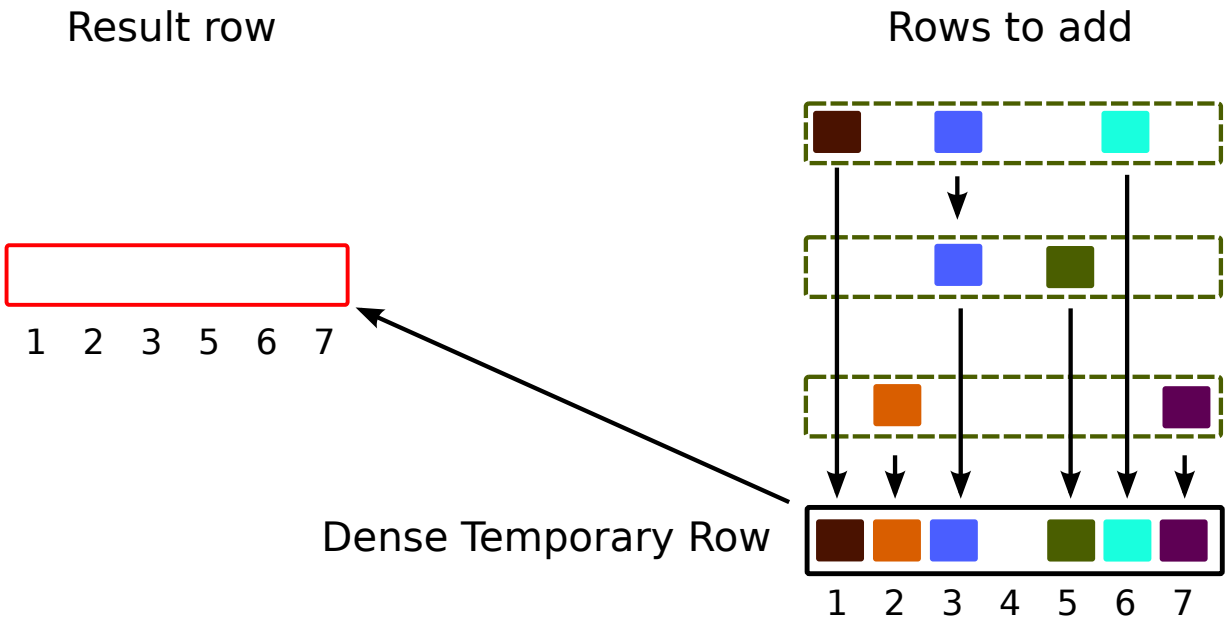
Sparse Matrix-Matrix Multiplications



Numeric Phase

Merge directly to dense array

Sparse Matrix-Matrix Multiplications

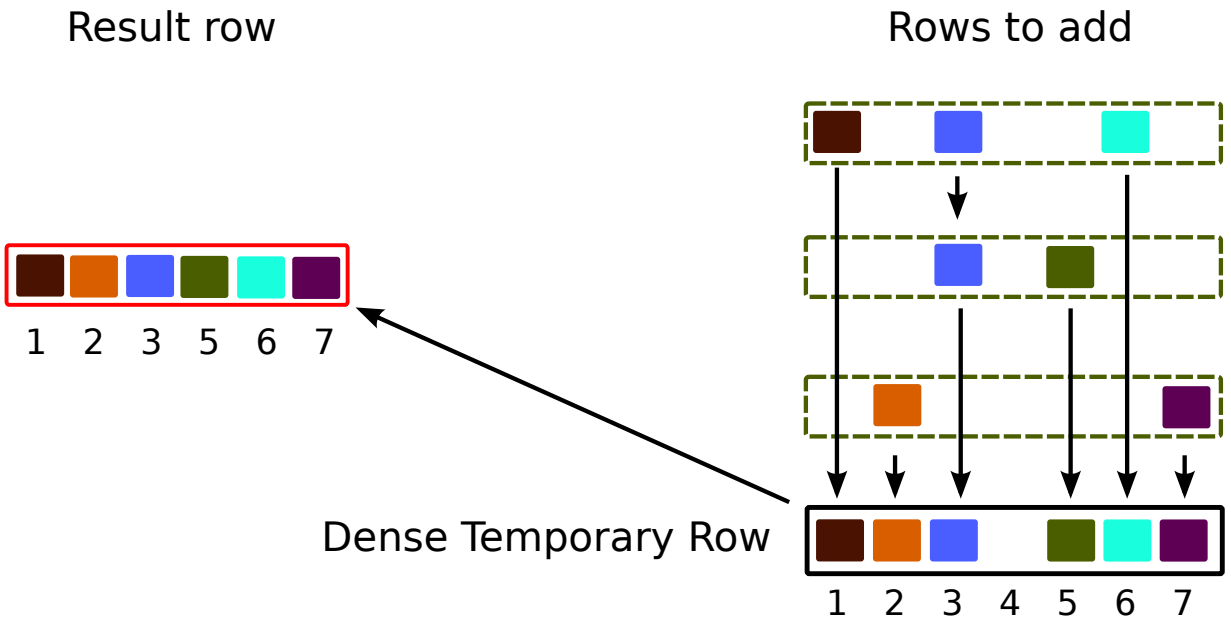


Numeric Phase

Merge directly to dense array

Pick up nonzeros to form $\mathbf{C}$

Sparse Matrix-Matrix Multiplications



Numeric Phase

Merge directly to dense array

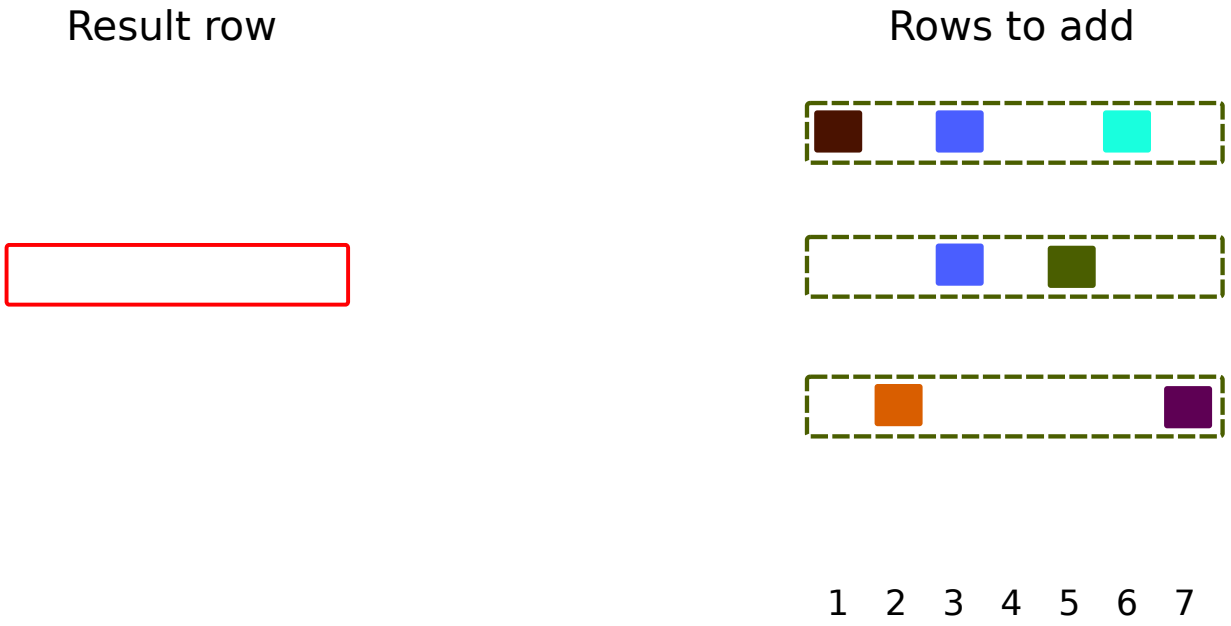
Pick up nonzeros to form $\mathbf{C}$

Sparse Matrix-Matrix Multiplications

Bad News:

Symbolic phase is tricky!

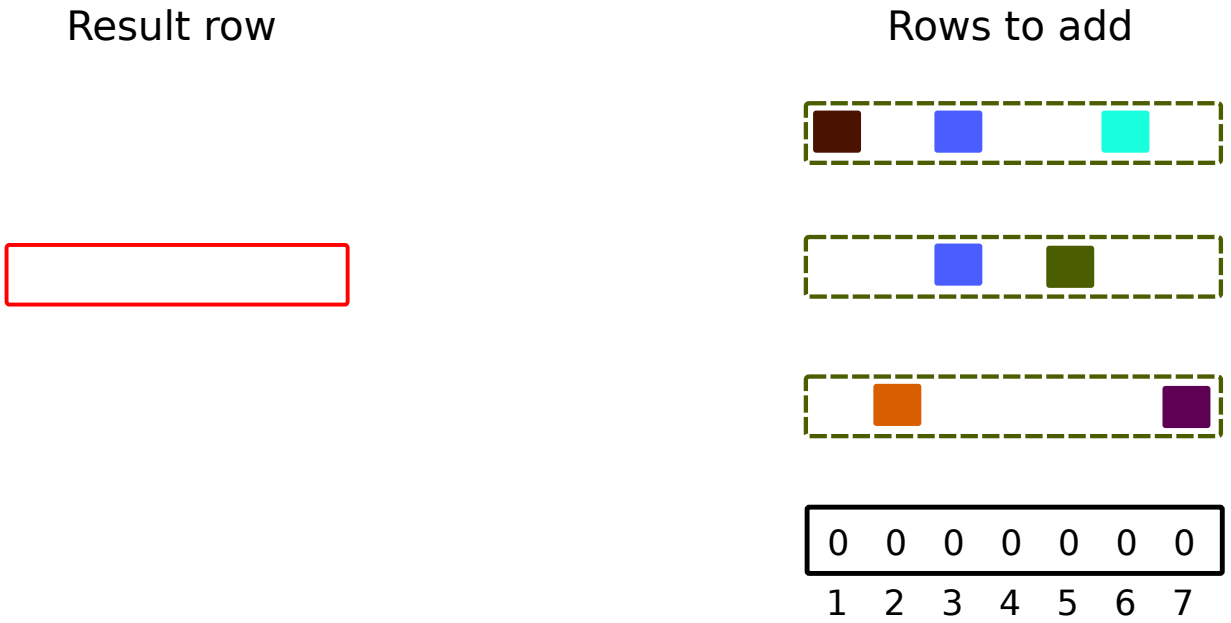
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

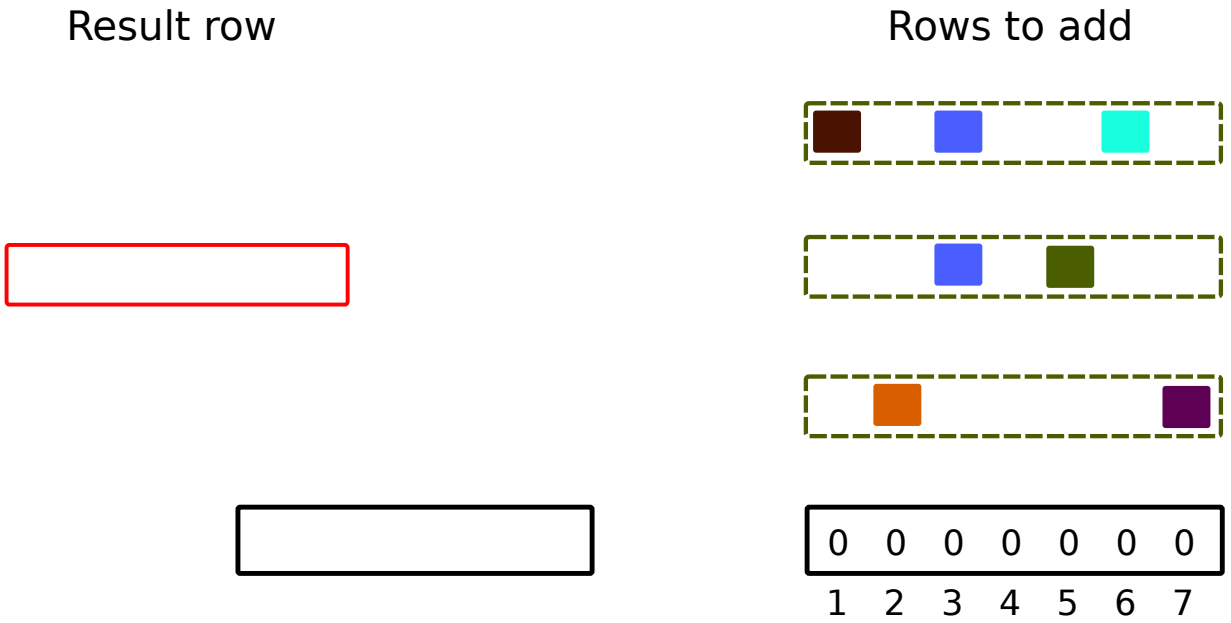
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

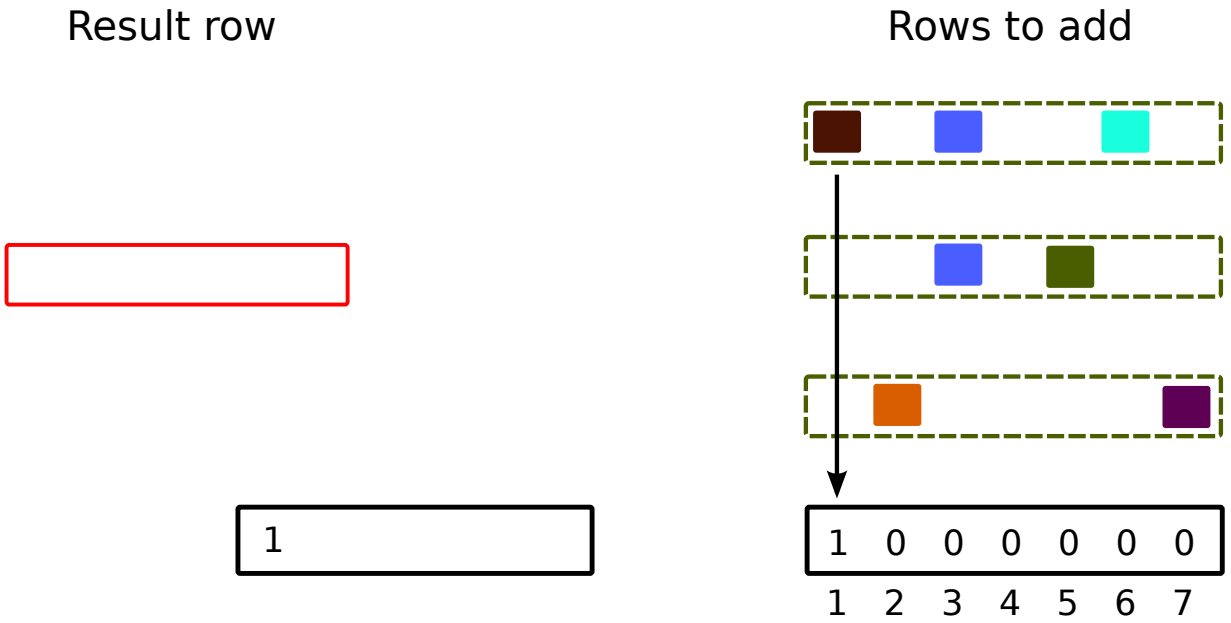
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

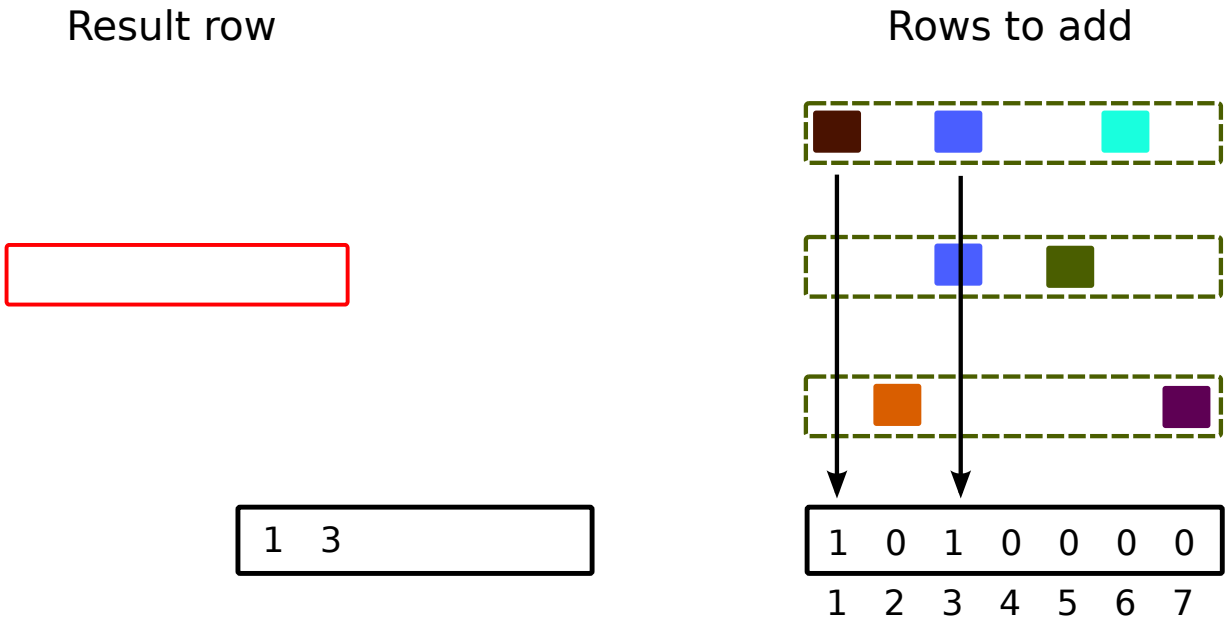
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

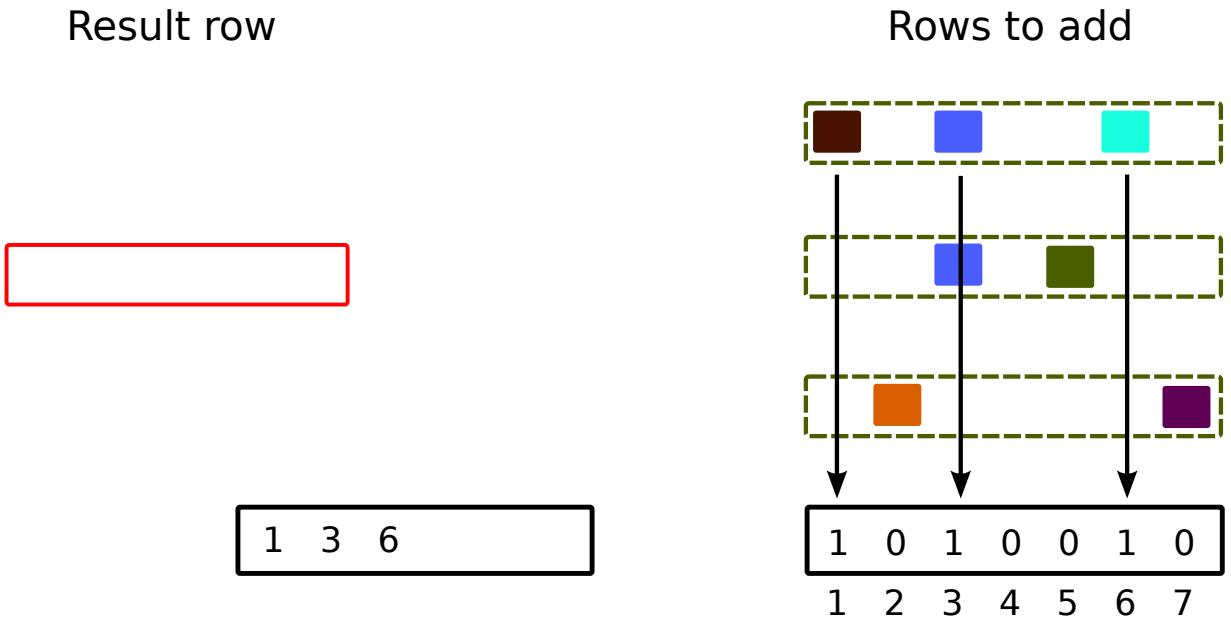
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

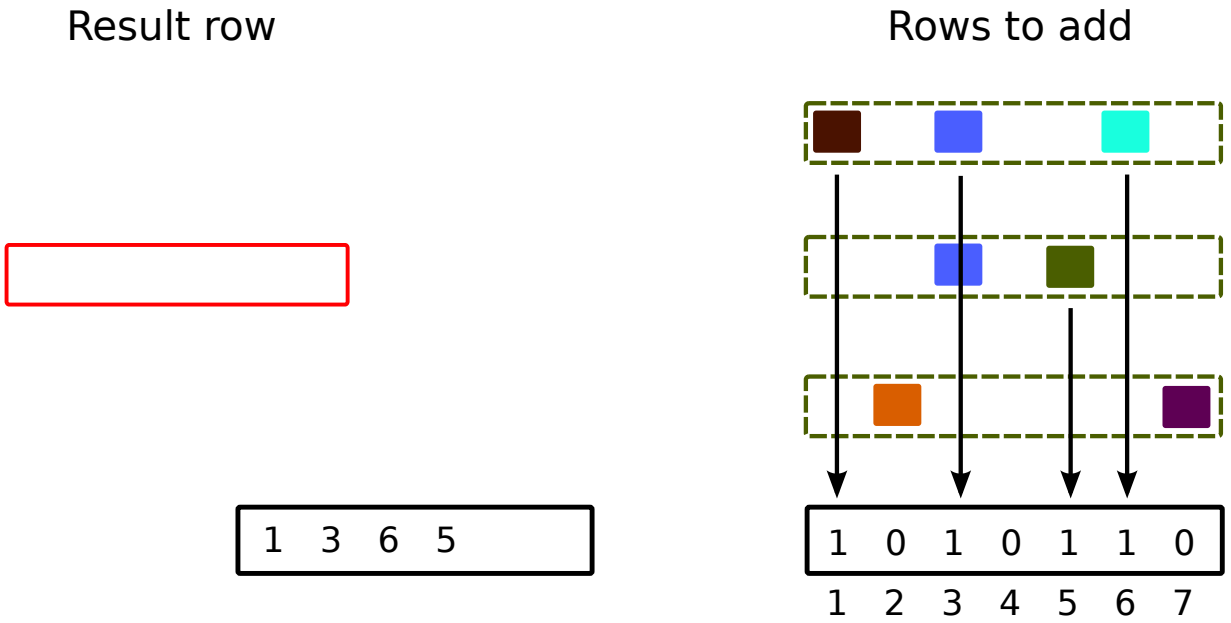
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

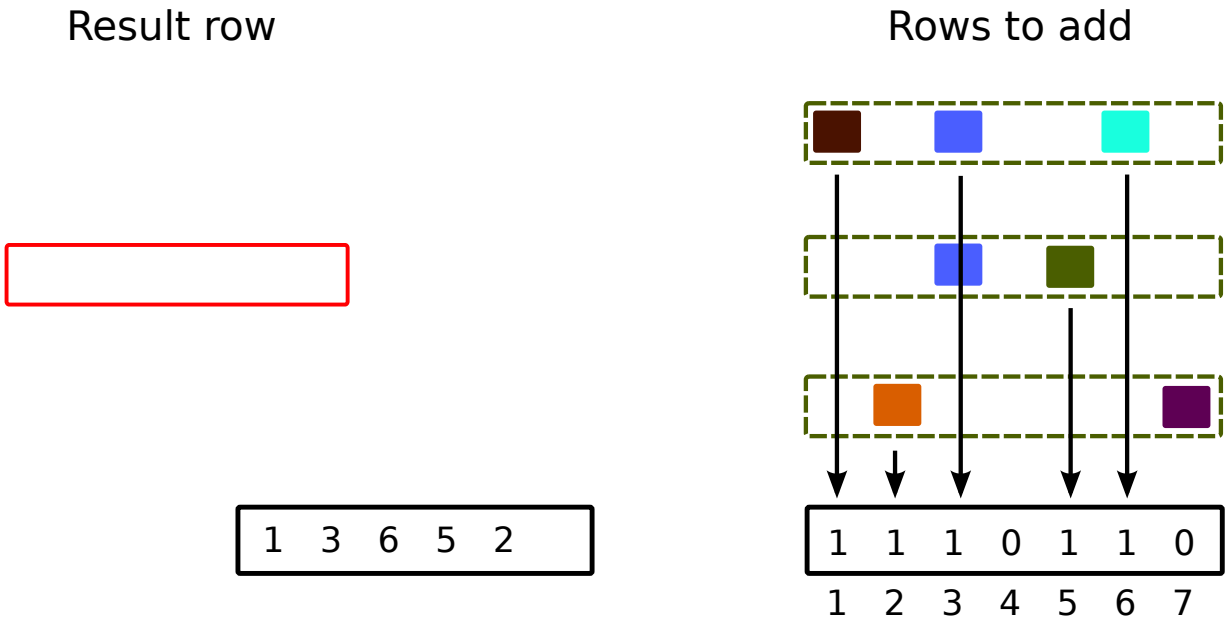
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

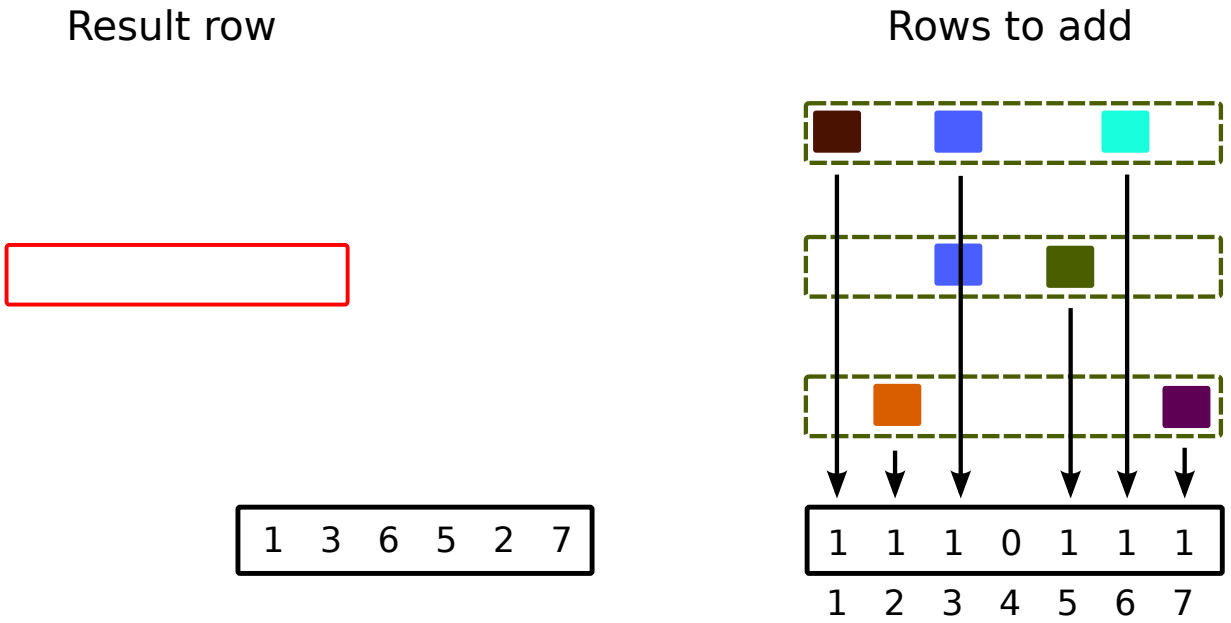
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

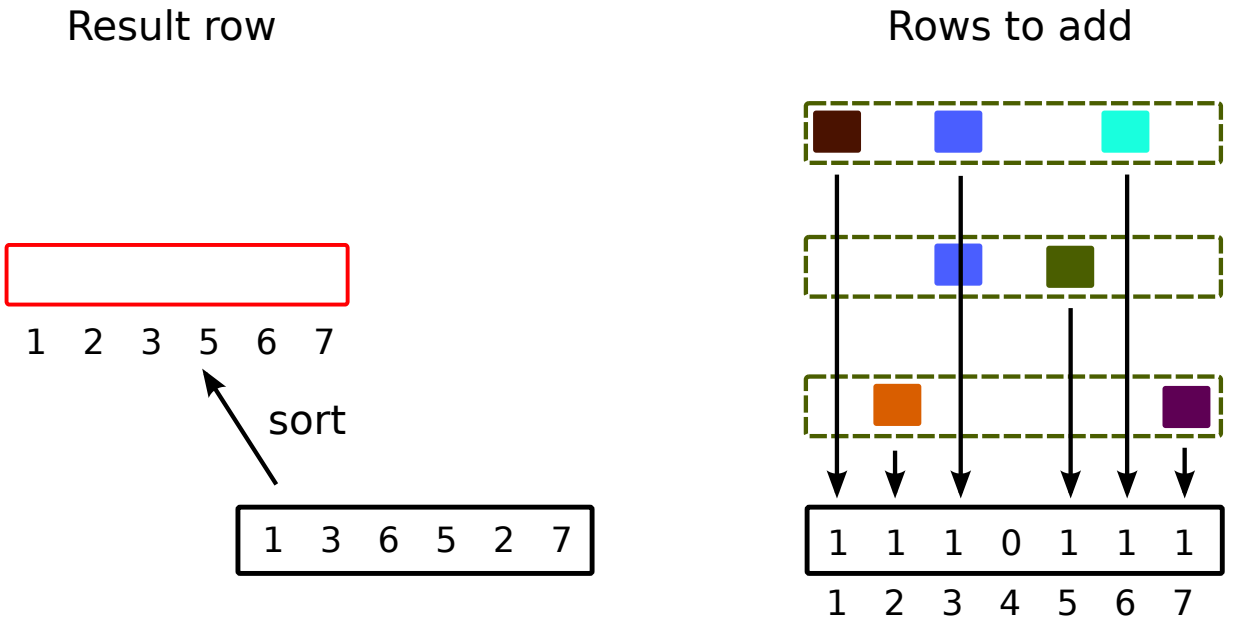
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

- Dense flag array
- Segmented buffer for nonzero indices

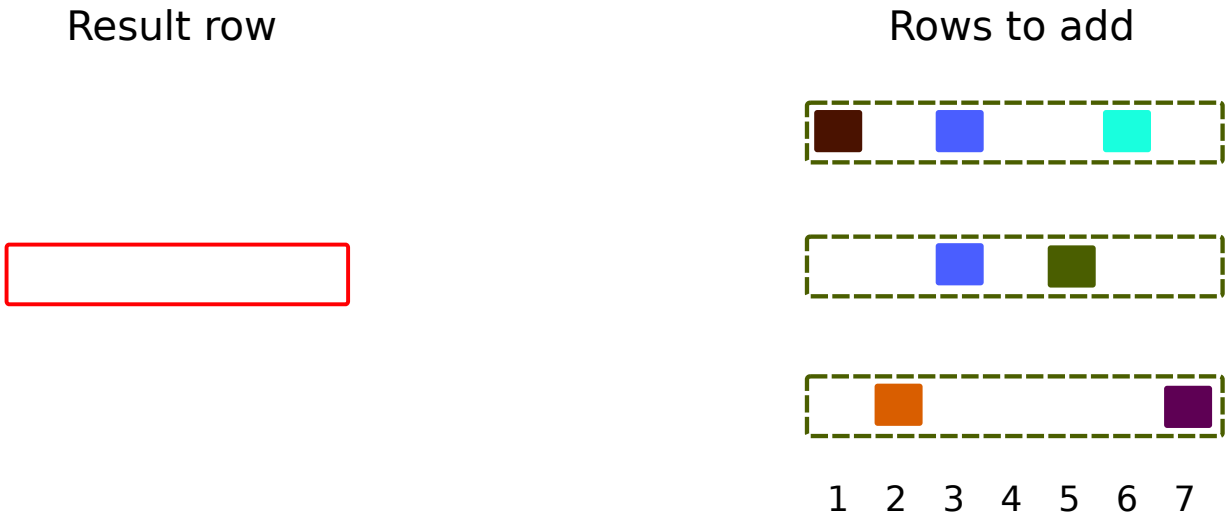
Sparse Matrix-Matrix Multiplications



Sorted MatMatMult in PETSc

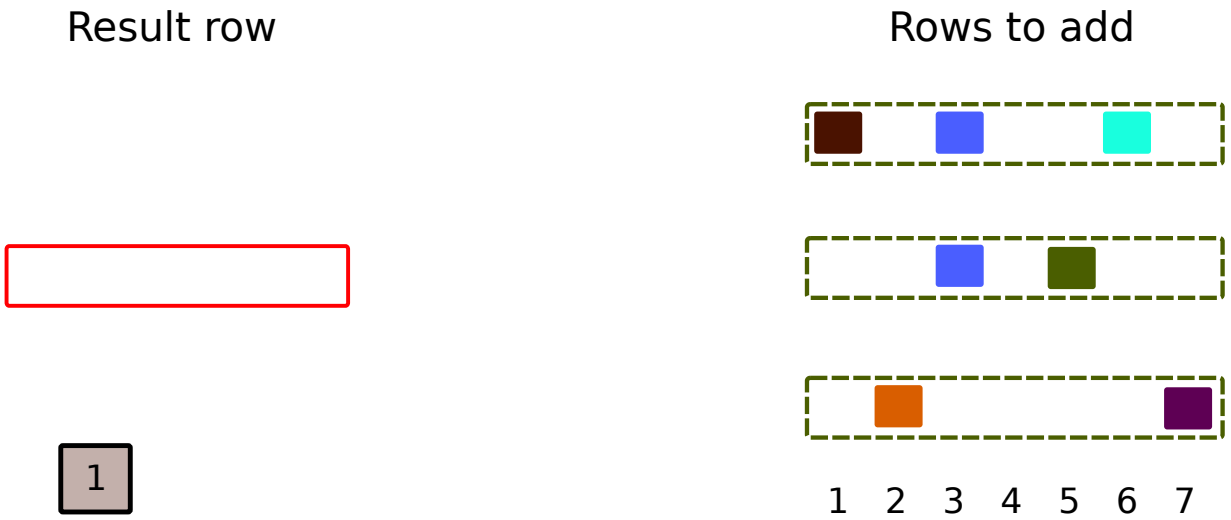
- Dense flag array
- Segmented buffer for nonzero indices

Sparse Matrix-Matrix Multiplications



Scalable MatMatMult in PETSc

Sparse Matrix-Matrix Multiplications

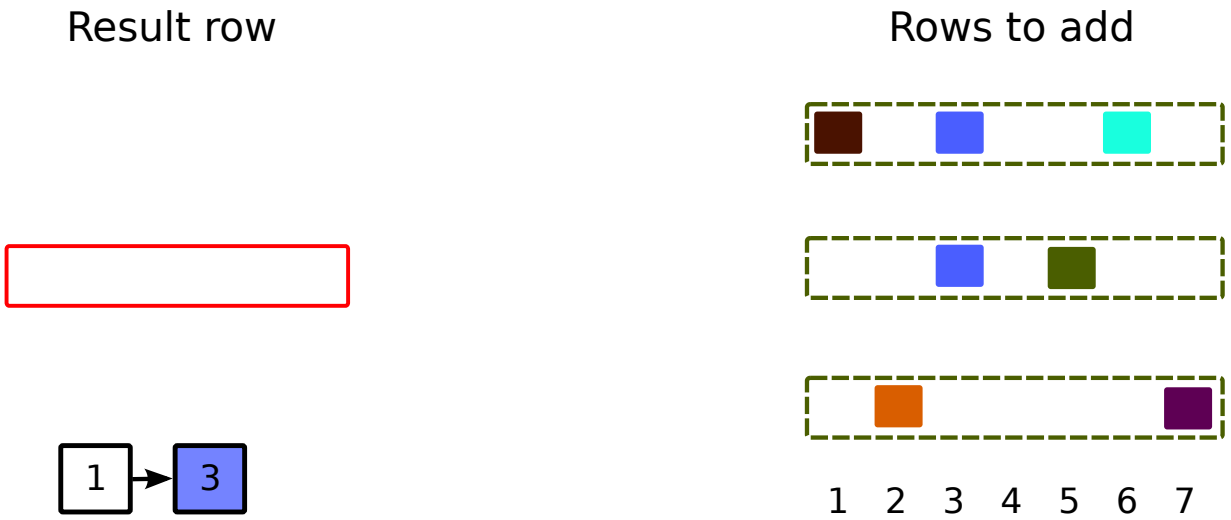


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

Sparse Matrix-Matrix Multiplications

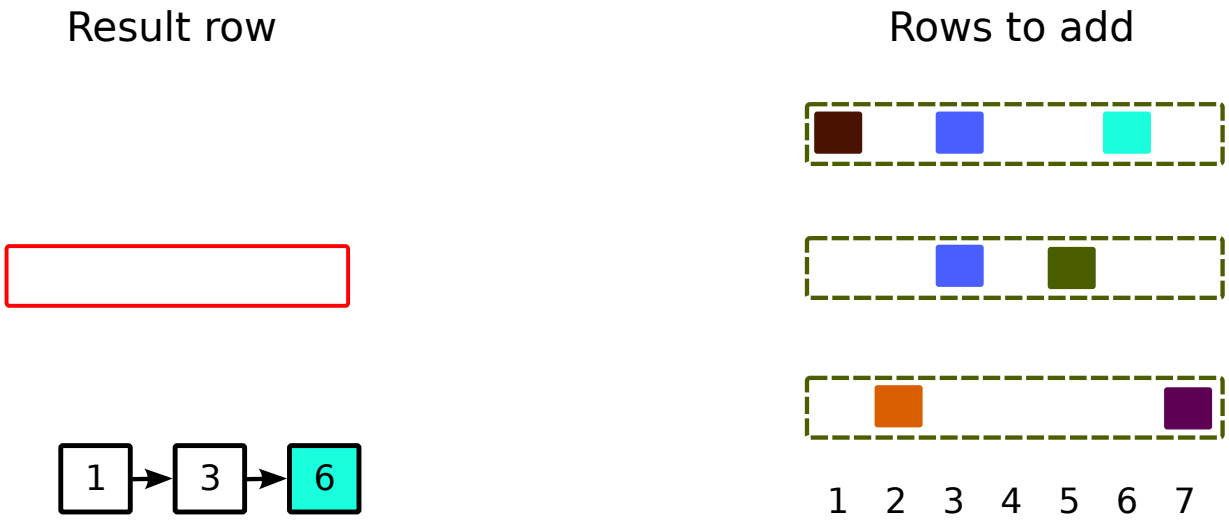


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

Sparse Matrix-Matrix Multiplications

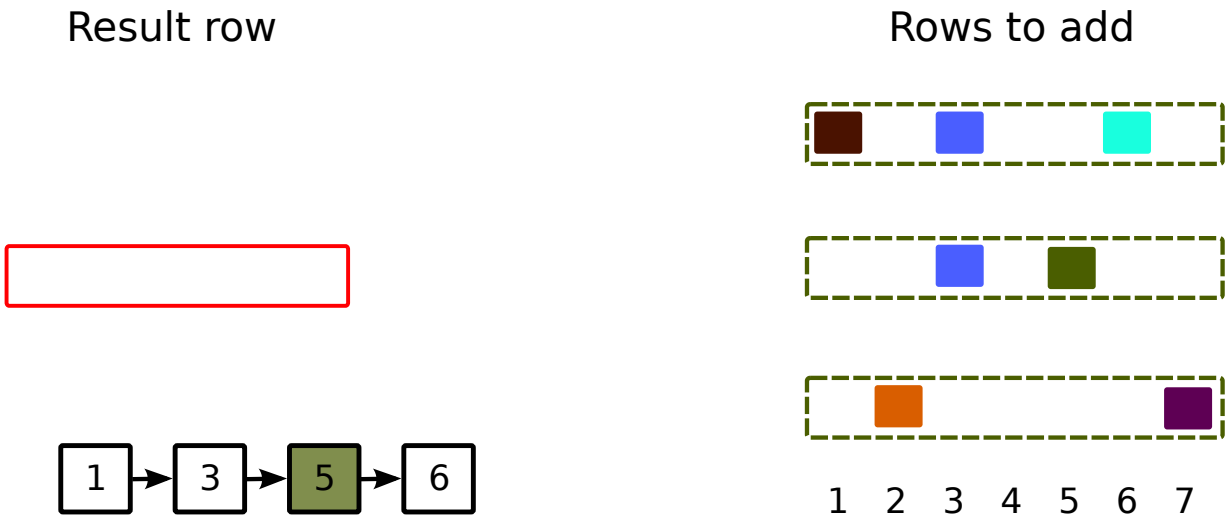


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

Sparse Matrix-Matrix Multiplications

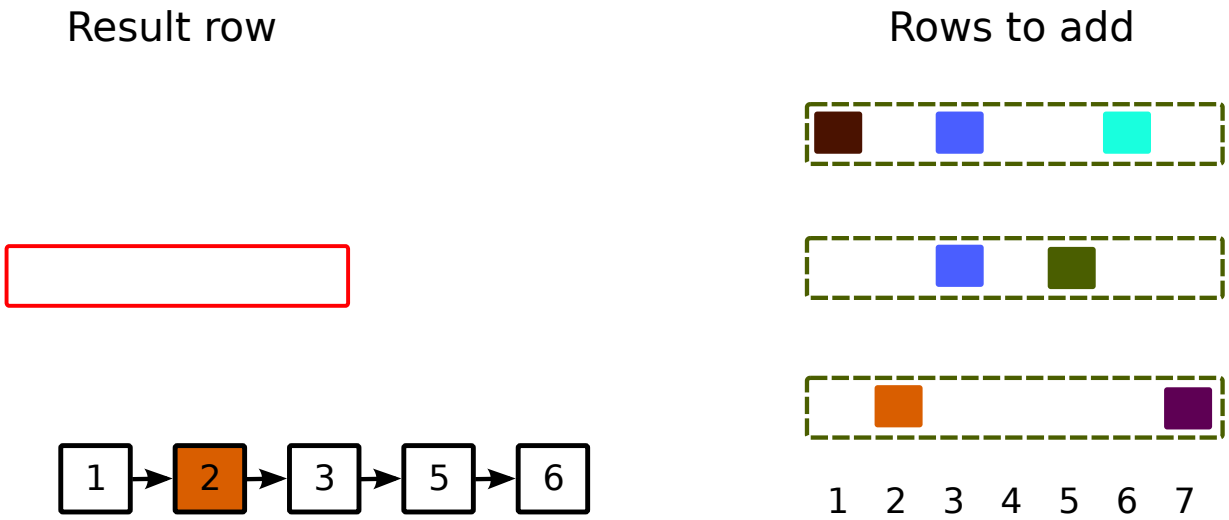


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

Sparse Matrix-Matrix Multiplications

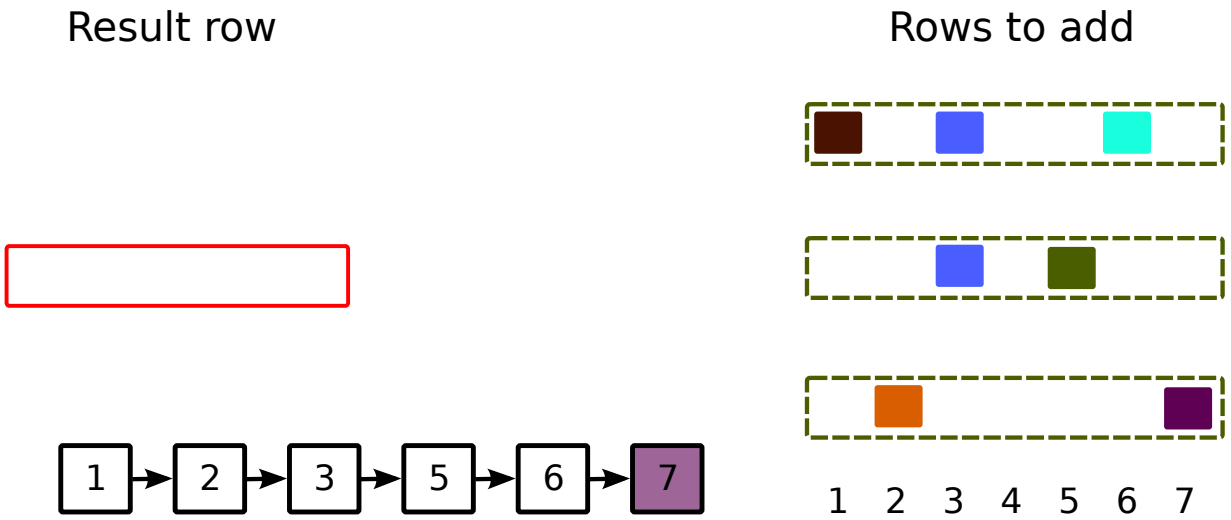


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

Sparse Matrix-Matrix Multiplications

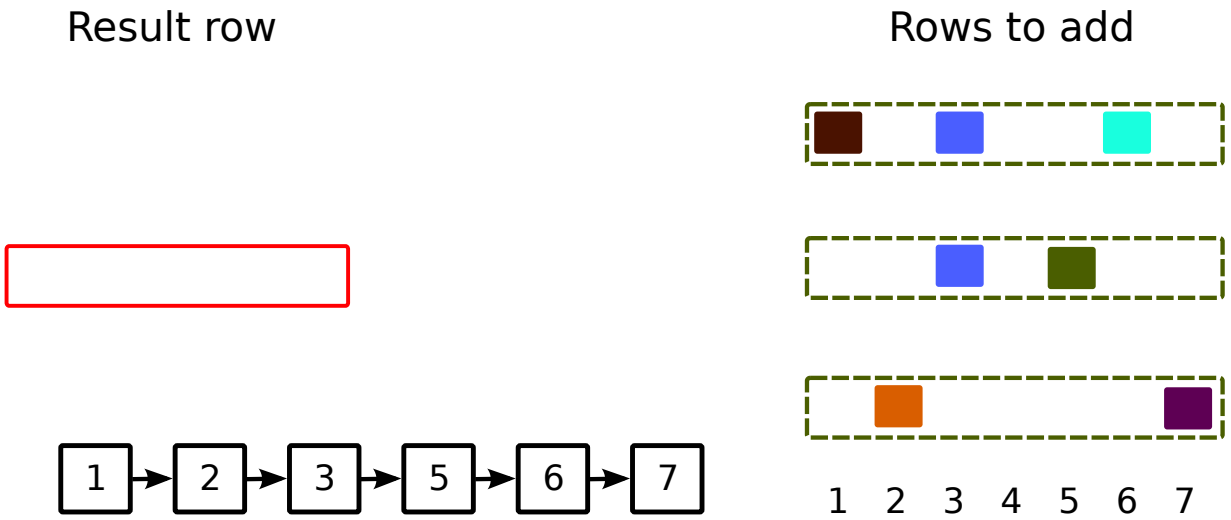


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

Sparse Matrix-Matrix Multiplications

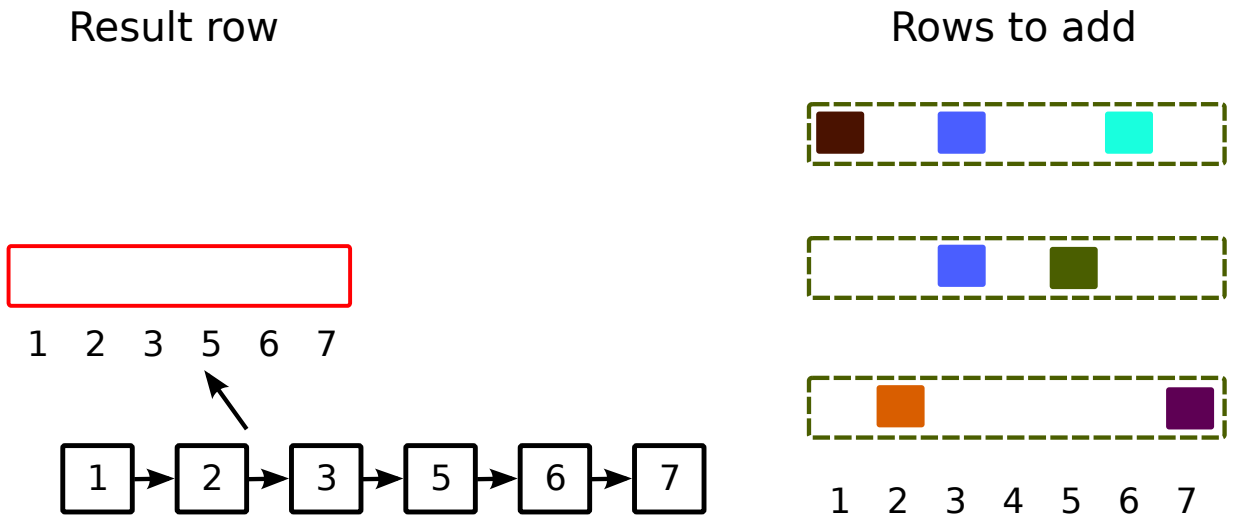


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

Sparse Matrix-Matrix Multiplications

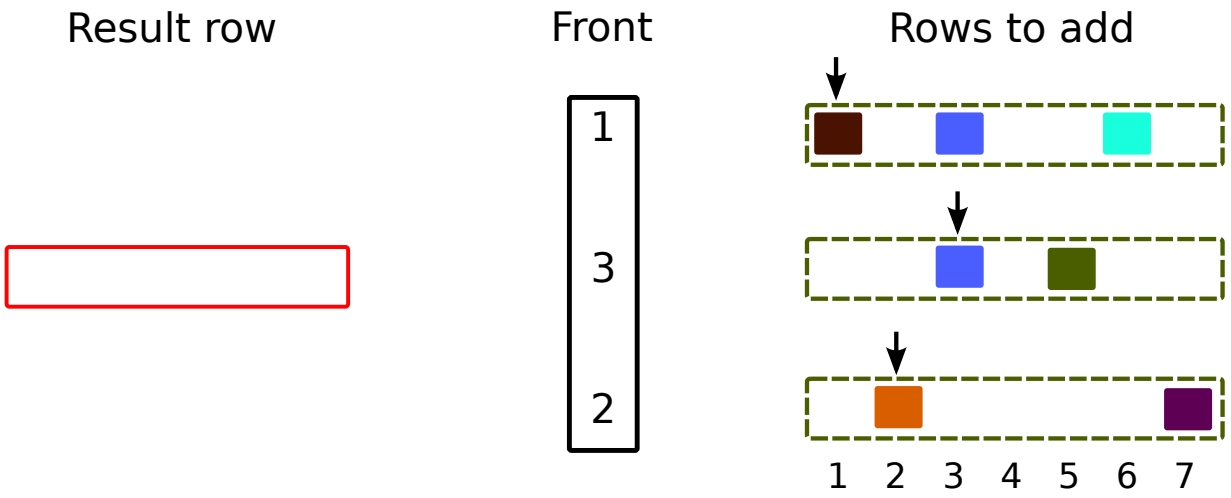


Scalable MatMatMult in PETSc

Use {linked list, heap, binary tree} to merge nonzero column indices

No dense array

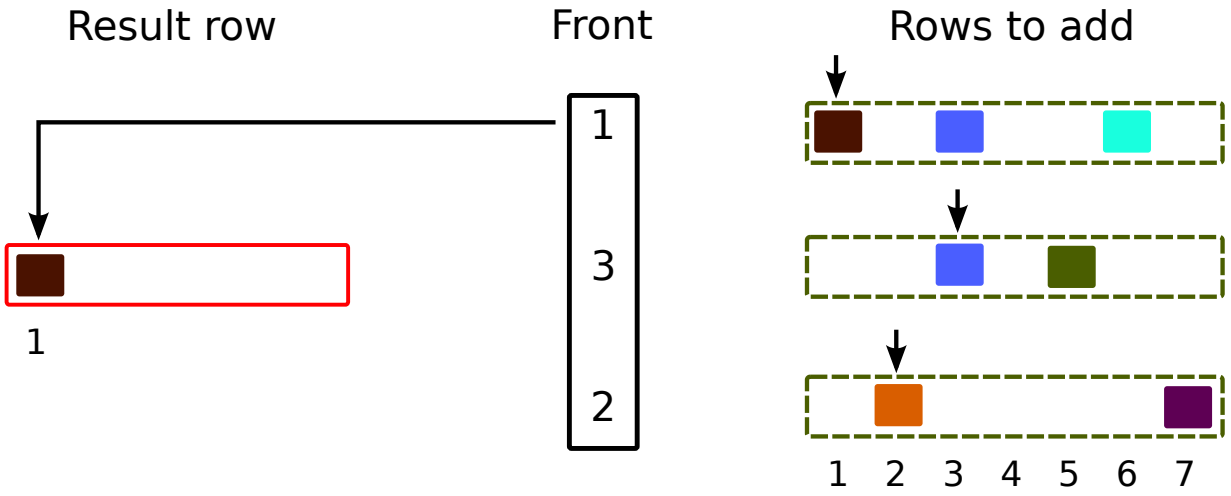
Row Merge Algorithm



Result Row Computation

1. Determine minimum index in front

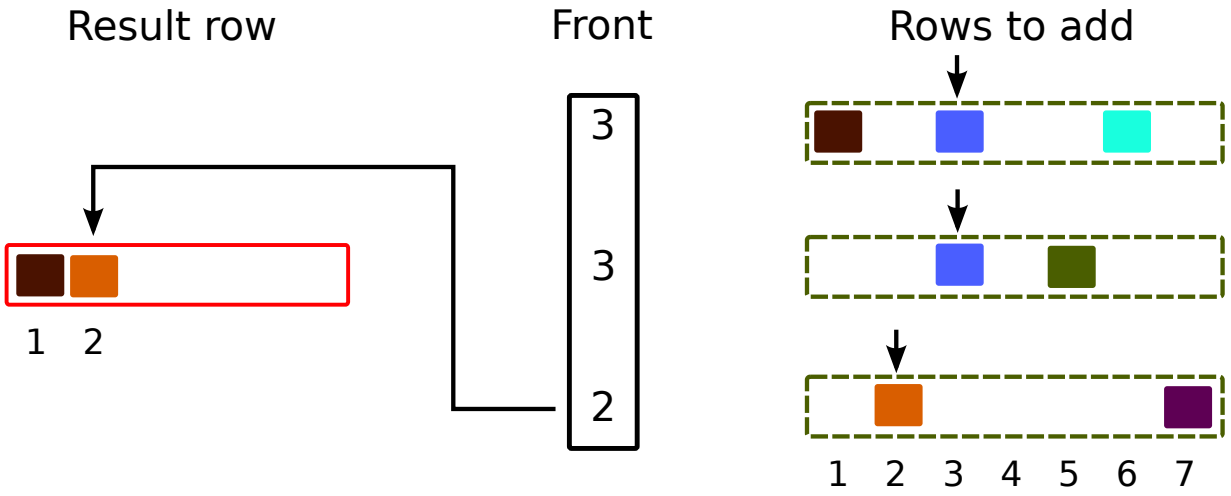
Row Merge Algorithm



Result Row Computation

1. Determine minimum index in front
2. Write minimum index

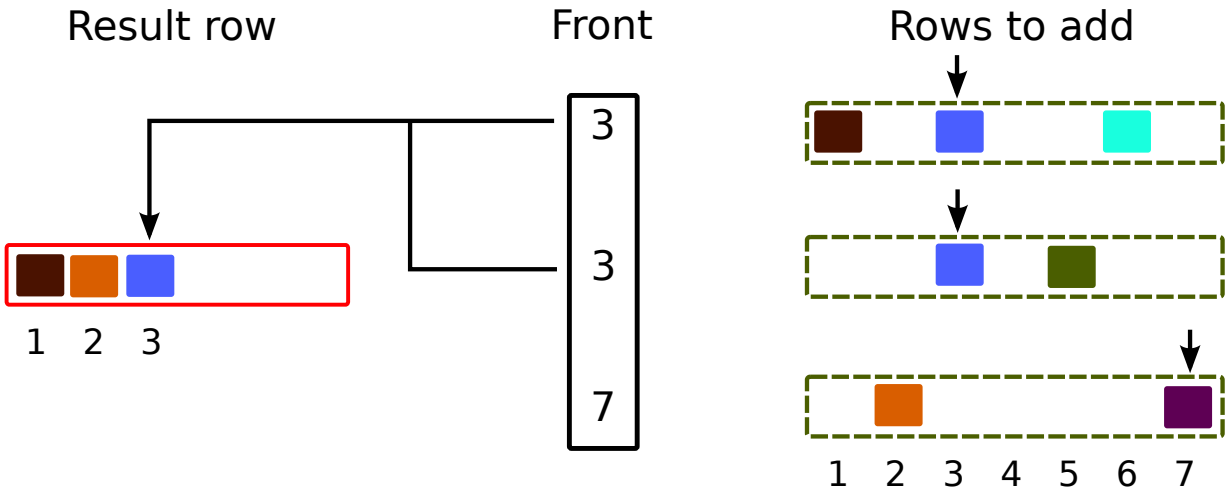
Row Merge Algorithm



Result Row Computation

1. Determine minimum index in front
2. Write minimum index
3. Advance front where minimum index occurred

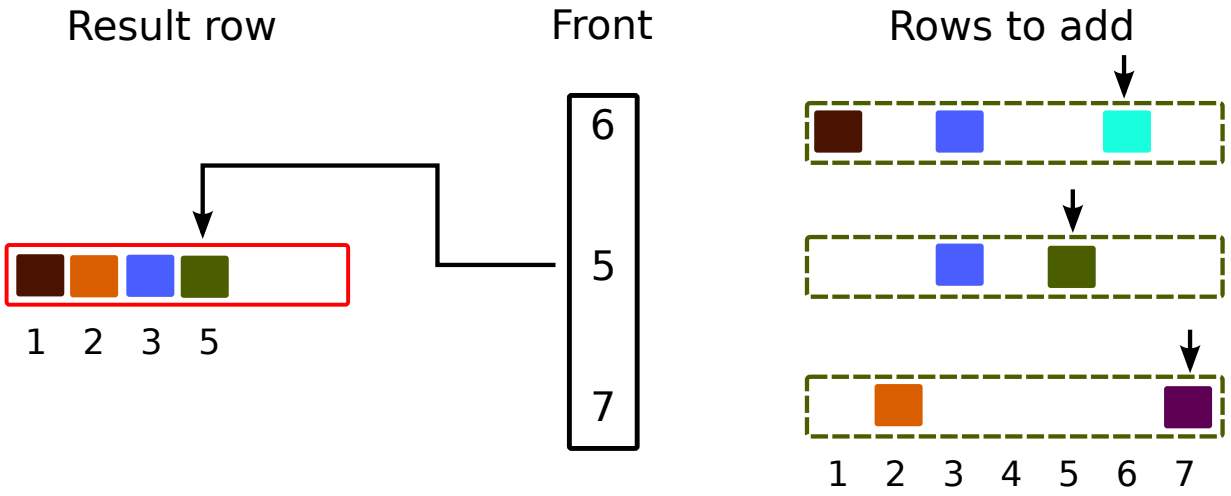
Row Merge Algorithm



Result Row Computation

1. Determine minimum index in front
2. Write minimum index
3. Advance front where minimum index occurred

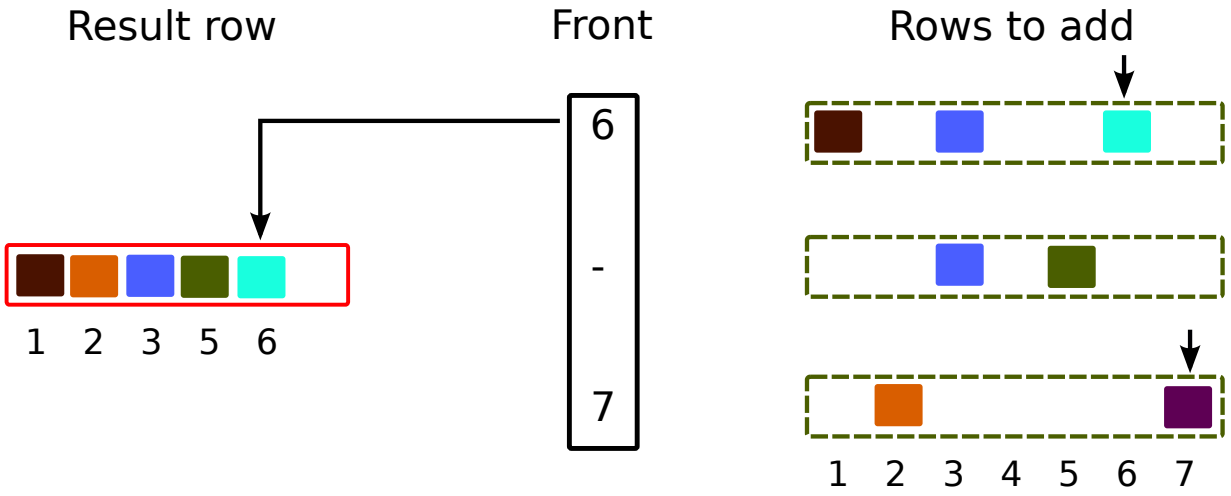
Row Merge Algorithm



Result Row Computation

1. Determine minimum index in front
2. Write minimum index
3. Advance front where minimum index occurred

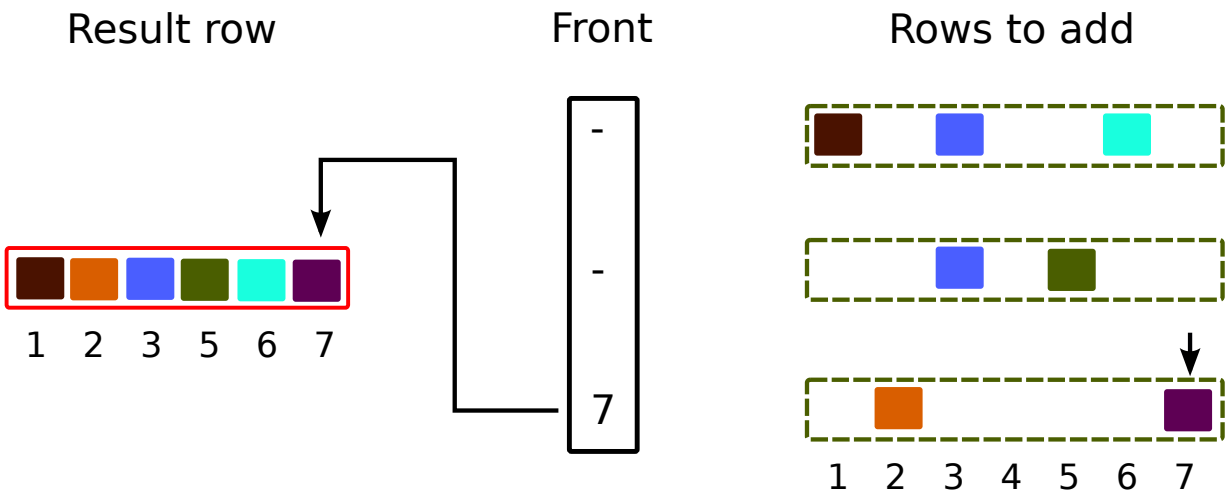
Row Merge Algorithm



Result Row Computation

1. Determine minimum index in front
2. Write minimum index
3. Advance front where minimum index occurred

Row Merge Algorithm



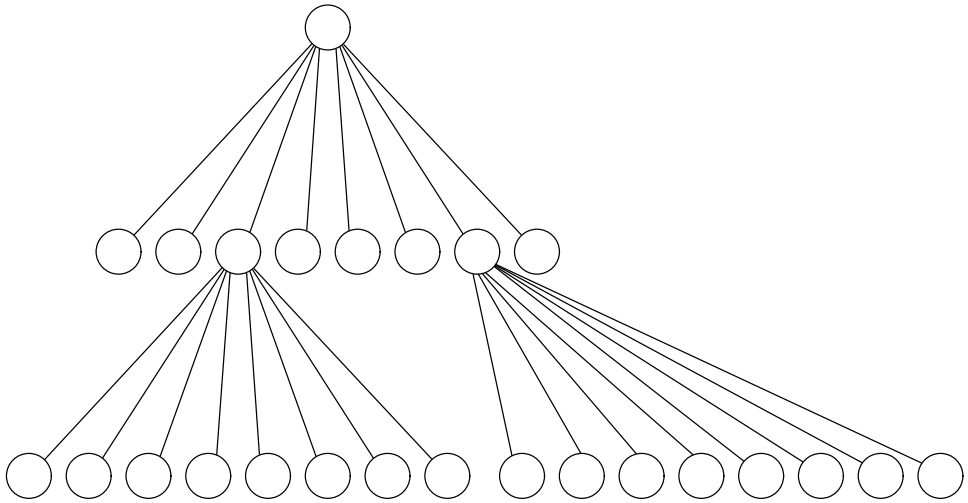
Result Row Computation

1. Determine minimum index in front
2. Write minimum index
3. Advance front where minimum index occurred

Sparse Matrix Products

Algorithm Details

- Split matrix if rows too large
- Recursively merge 8 rows
- (Re-)use scratchpad memory



https://en.wikipedia.org/wiki/Octree\#/media/File:Octree2.svg

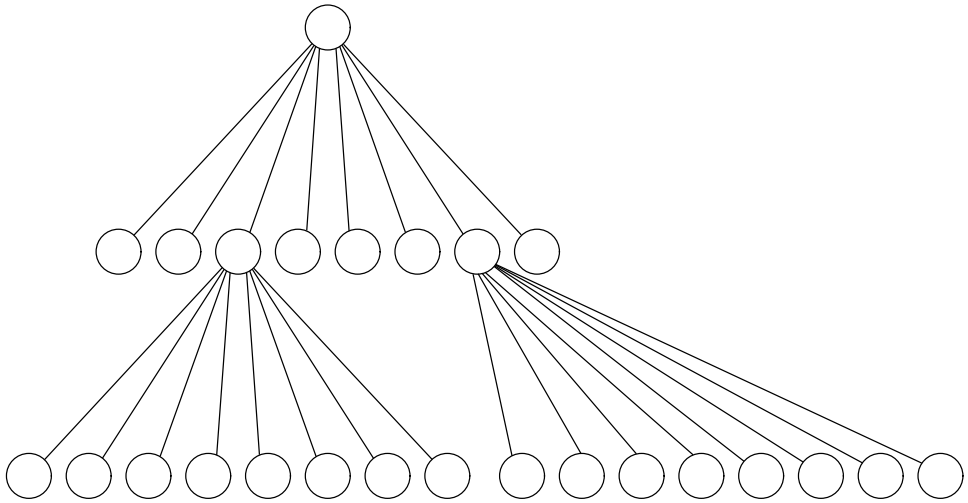
Sparse Matrix Products

Algorithm Details

Split matrix if rows too large

Recursively merge 8 rows

(Re-)use scratchpad memory



https://en.wikipedia.org/wiki/Octree\#/media/File:Octree2.svg

Hardware Details

CPU: Use AVX2 to merge 8 rows simultaneously

GPU: Merge up to 32/64 rows simultaneously with one warp/wavefront

MIC: Use AVX-KNC to merge 16 rows simultaneously

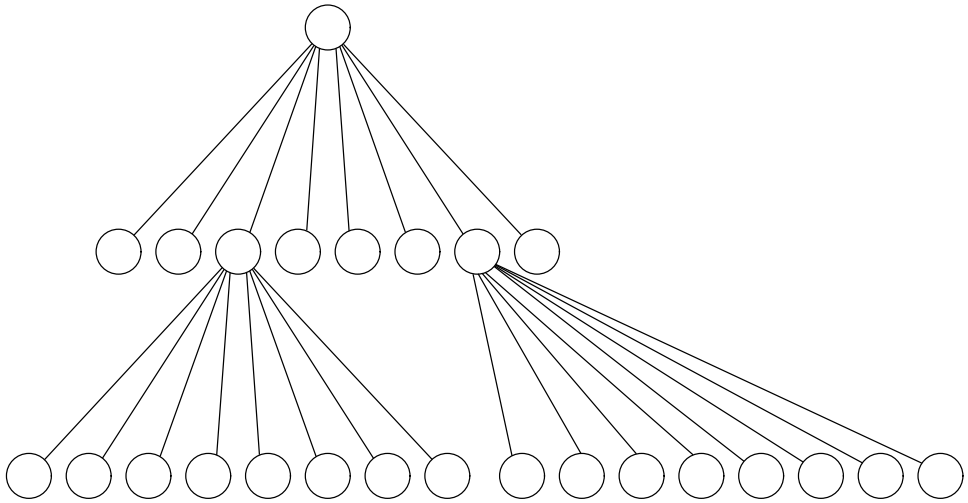
Sparse Matrix Products

Algorithm Details

Split matrix if rows too large

Recursively merge 8 rows

(Re-)use scratchpad memory



https://en.wikipedia.org/wiki/Octree\#/media/File:Octree2.svg

Hardware Details

CPU: Use AVX2 to merge 8 rows simultaneously

GPU: Merge up to 32/64 rows simultaneously with one warp/wavefront

MIC: Use AVX-KNC to merge 16 rows simultaneously

Parallelization and Load Balancing

CPU, MIC: Use MPI!

CPU, MIC, alternative: OpenMP dynamic scheduling, chunk size 1024

GPU: thread scheduling in hardware

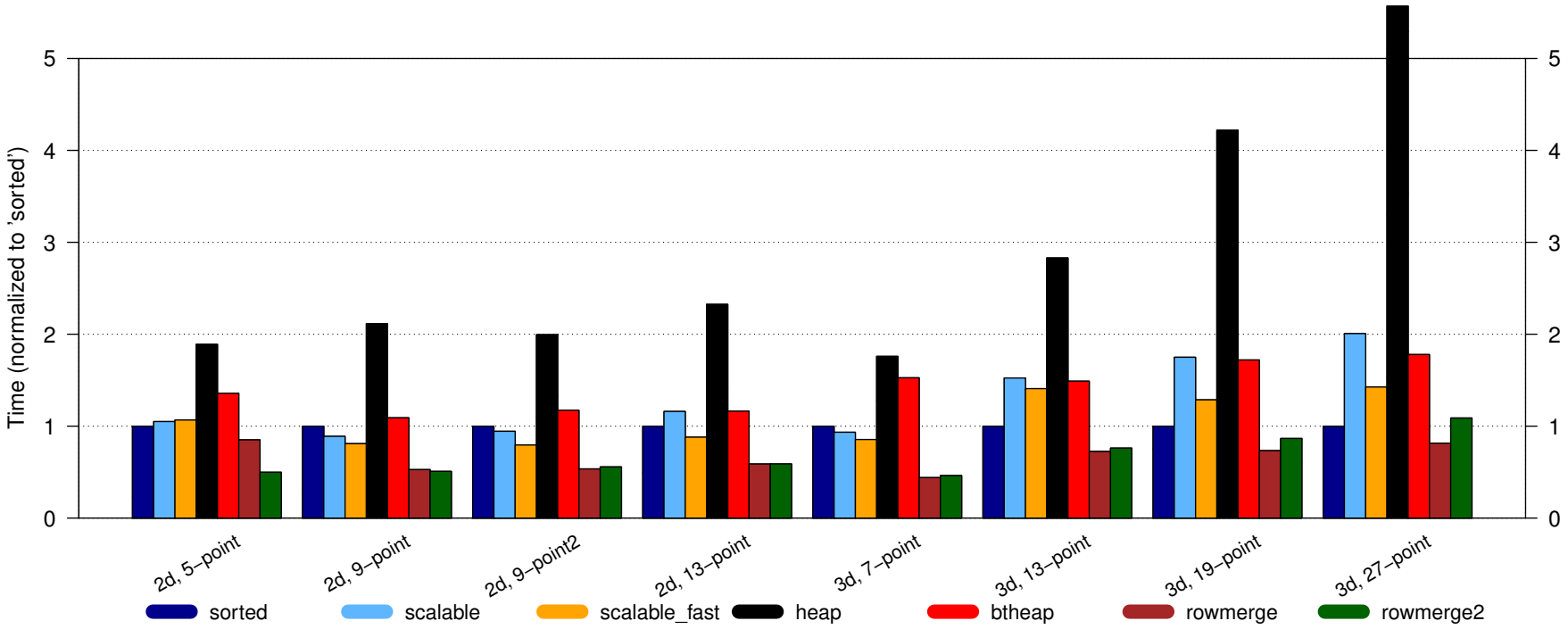
Benchmarks

Which approach is fastest?

Sorted vs. Scalable vs. Scalable, fast vs. Heap vs. BTHeap vs. Linked-list condensed vs. RowMerge vs. RowMerge2

Benchmark Results

Comparison of Sequential MatMatMult in PETSc



(Tests run on an Intel Core i3-3217U)

Row Merge up to 2x faster than PETSc's default

Benchmark Setup

Hardware for Comparison

AMD FirePro W9100 GPU

Intel Xeon E5-2670v3 (Haswell, dual socket)

NVIDIA Tesla K20

Software for Comparison

Intel MKL 11.2.1

NVIDIA cuSPARSE 7.0

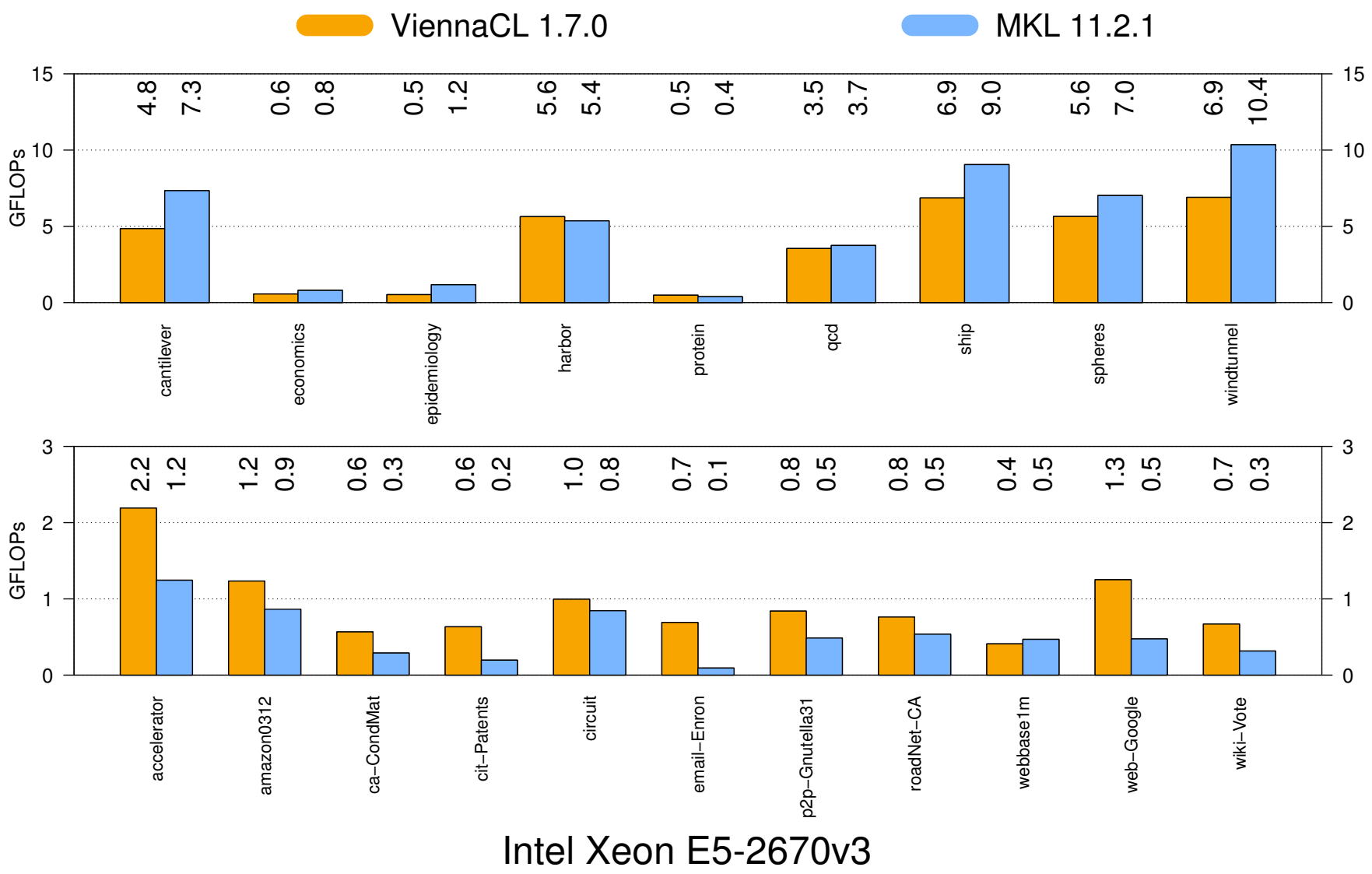
CUSP 0.5.1

Matrices

20 matrices from Florida Sparse Matrix Collection

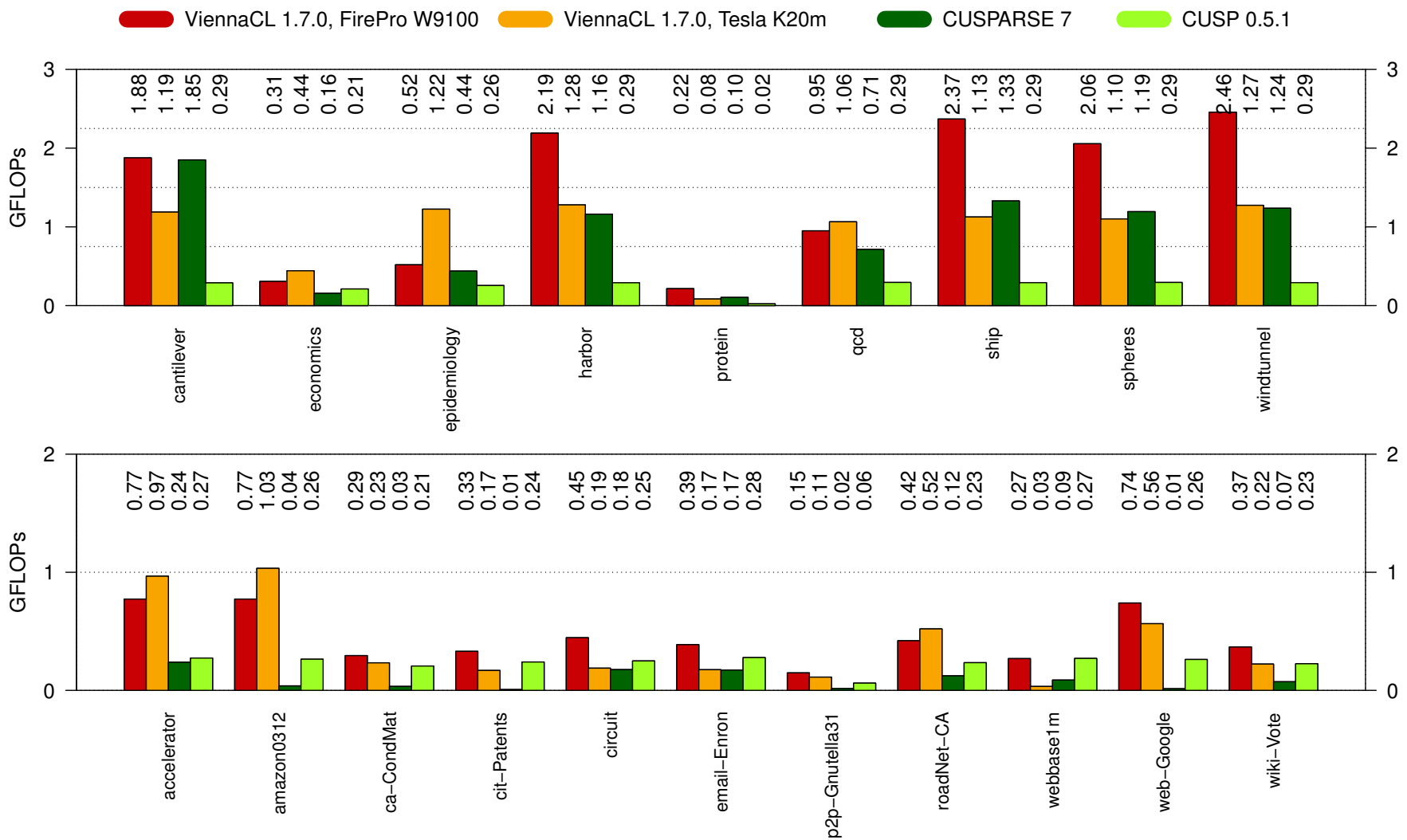
Operation: $\mathbf{B} = \mathbf{AA}$

Benchmark Results



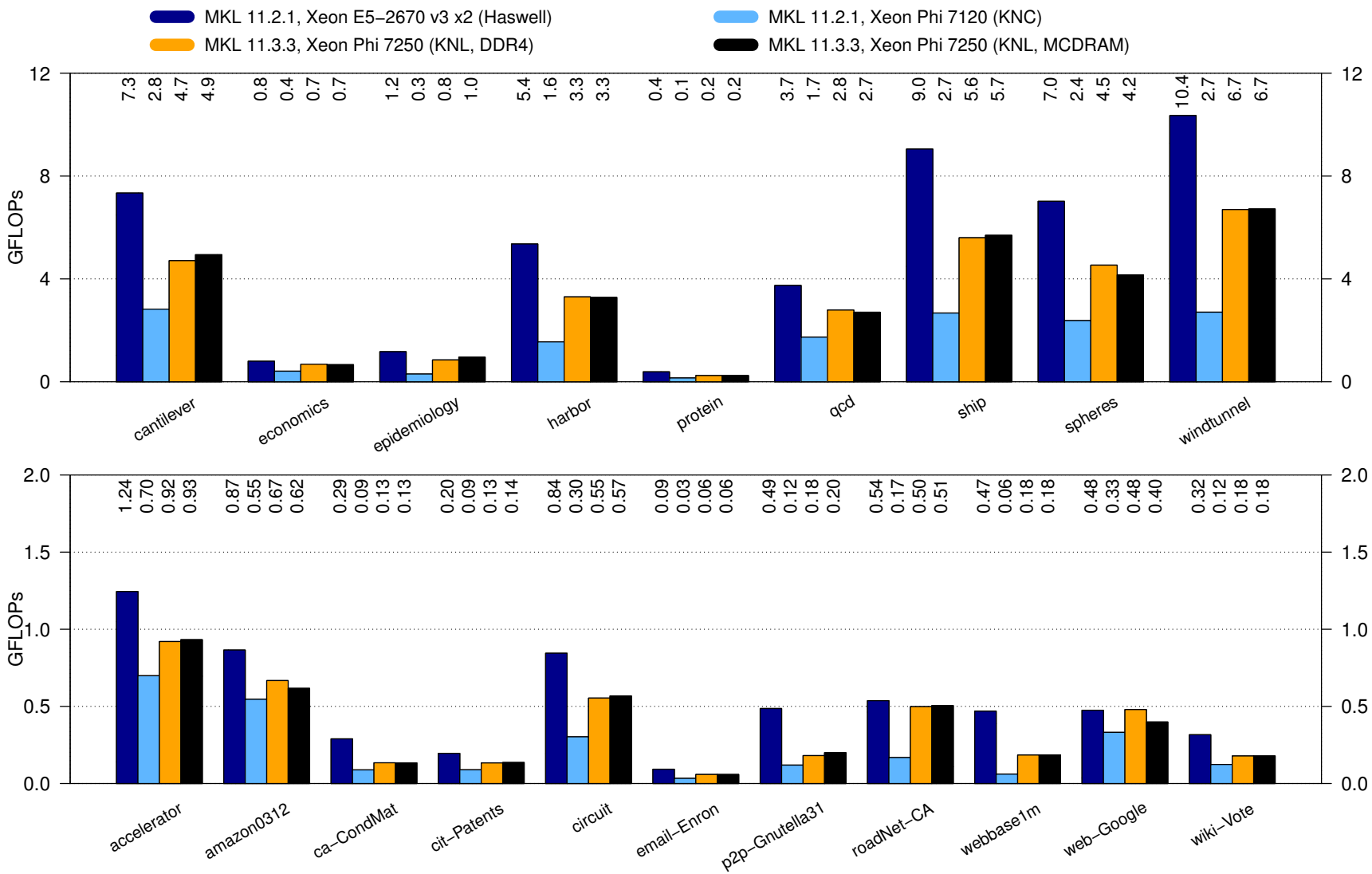
Intel Xeon E5-2670v3

Benchmark Results



AMD FirePro W9100, NVIDIA Tesla K20m

Benchmark Results



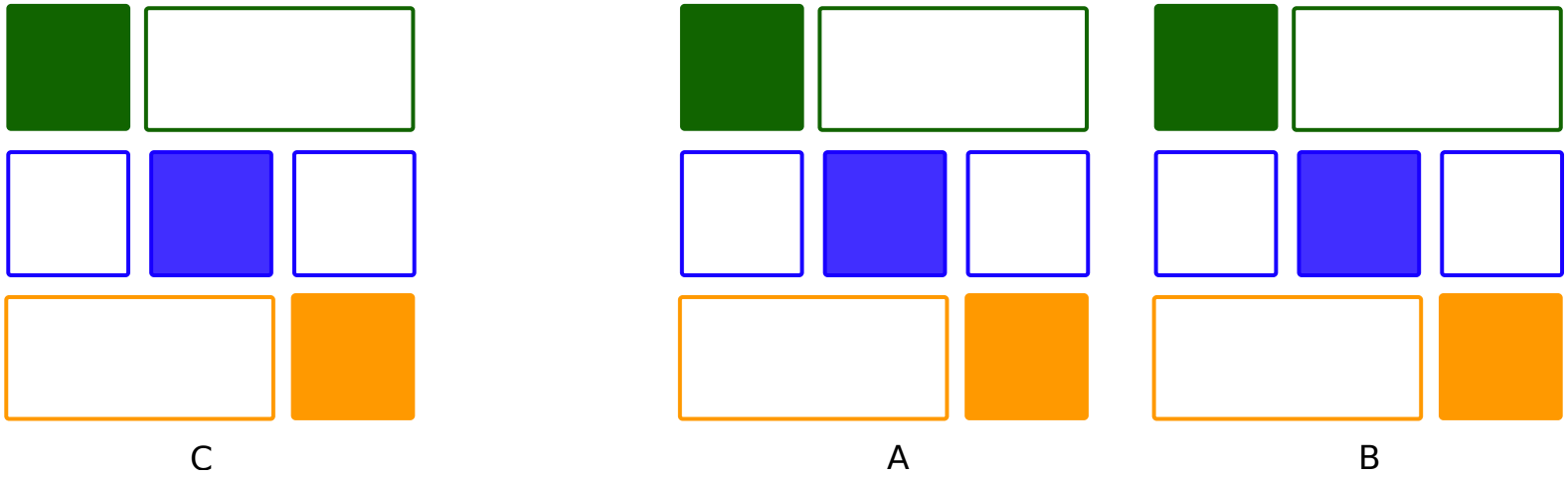
Comparison on Intel Xeon E5-2670v3 (Haswell) vs. Xeon Phi (KNL)

Sparse Matrix-Matrix Multiplications

Parallel MatMatMult in PETSc

Scalable

Non-Scalable

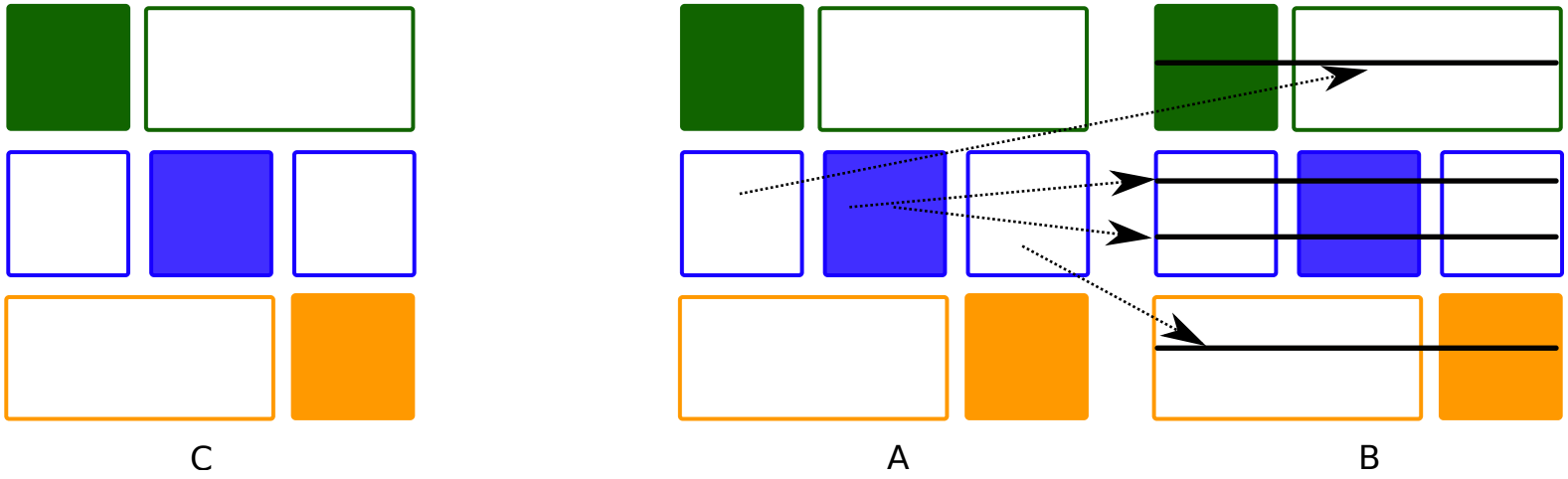


Sparse Matrix-Matrix Multiplications

Parallel MatMatMult in PETSc

Scalable

Non-Scalable

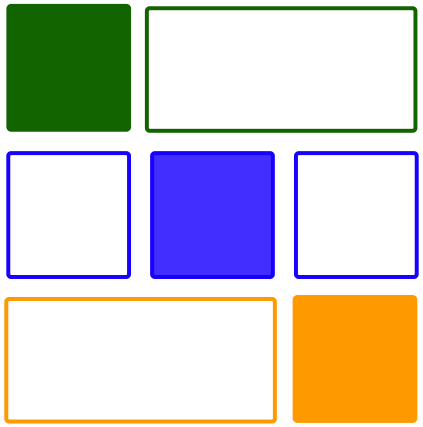


Sparse Matrix-Matrix Multiplications

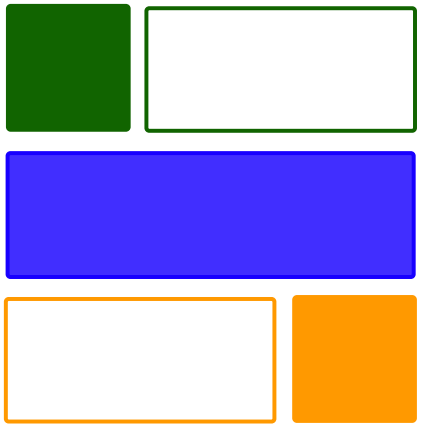
Parallel MatMatMult in PETSc

Scalable

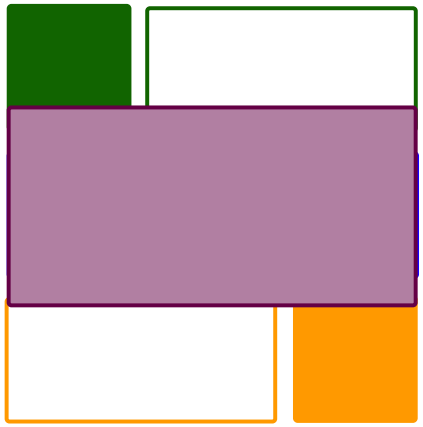
Non-Scalable



C



A



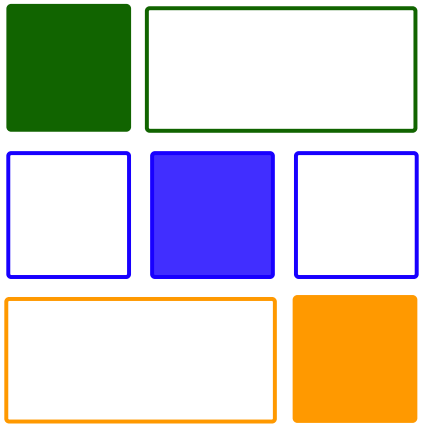
B

Sparse Matrix-Matrix Multiplications

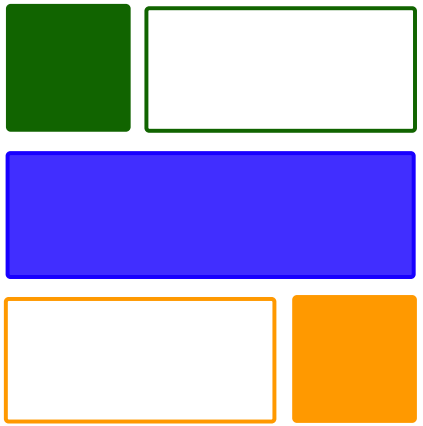
Parallel MatMatMult in PETSc

Scalable

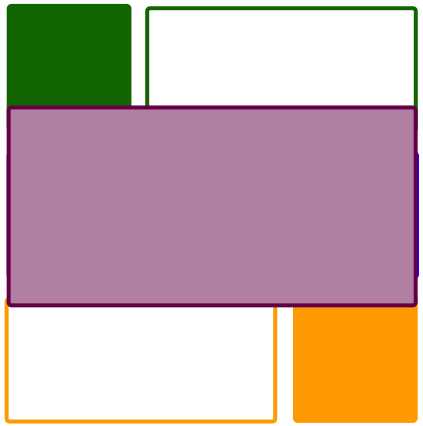
Non-Scalable



C



A



B

Lots of bookkeeping...

Summary

Sequential Sparse Matrix-Matrix Products

- Row Merge faster than MKL on Xeon CPUs on average
- Row Merge faster than cuSPARSE and CUSP on Tesla K20m (and others)

Summary

Sequential Sparse Matrix-Matrix Products

- Row Merge faster than MKL on Xeon CPUs on average
- Row Merge faster than cuSPARSE and CUSP on Tesla K20m (and others)

Parallel Sparse Matrix-Matrix Products

- Potential for an approach similar to Row Merge
- Fine-tune use of scalable vs. non-scalable routines

Summary

Sequential Sparse Matrix-Matrix Products

- Row Merge faster than MKL on Xeon CPUs on average
- Row Merge faster than cuSPARSE and CUSP on Tesla K20m (and others)

Parallel Sparse Matrix-Matrix Products

- Potential for an approach similar to Row Merge
- Fine-tune use of scalable vs. non-scalable routines

Implications and Outlook

- CPUs beat accelerators for sparse matrix-matrix products (caches!)
- Full integration into PETSc's GAMG (algebraic multigrid)
- Find CPU/GPU balance for hybrid clusters