EDGAR

- **E**nergy-efficient **D**ata and **G**raph **A**lgorithms **R**esearch
- Funded by Applied Math, ASCR
- Program manager: Sandy Landsberg
- Early Career Research Program (start: 2013)
- PI: Aydın Buluç (Berkeley Lab)
- Postdoctoral Fellows:
  - Ariful Azad (100%, Feb 2014-now)
  - Harsha Simhadri (40%, Sep 2013- Sep 2014)
- Students (short term):
  - Chaitanya Aluru (Nov 2014-now)
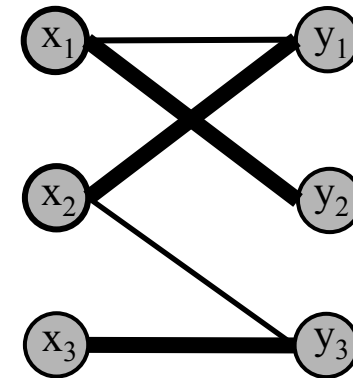 - Eric Lee (Summer 2014)

Problem: maximum cardinality matching in bipartite graph.

Application: block triangular forms of matrices, least square problems, circuit simulation, weighted matching.

Algorithm: Search for disjoint paths alternating between matched and unmatched edges.

Vertex		Edge	
⬤	Matched	▬▬	Matched
◯	Unmatched	──	Unmatched

Maximum Cardinality Matching

Innovation: Re-use search trees created in one phase in the next phase by grafting branches of trees. Significantly reduces work in the tree-traversal and exposes more parallelism.
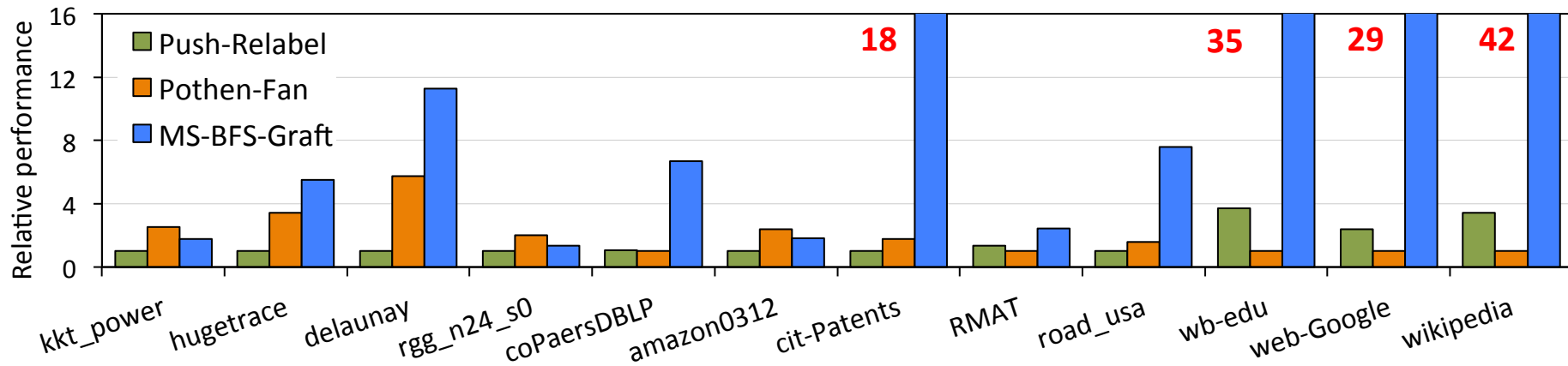
Ariful Azad, Aydın Buluç, and Alex Pothen. A parallel tree grafting algorithm for maximum cardinality matching in bipartite graphs. In Proceedings of the IPDPS, 2015.
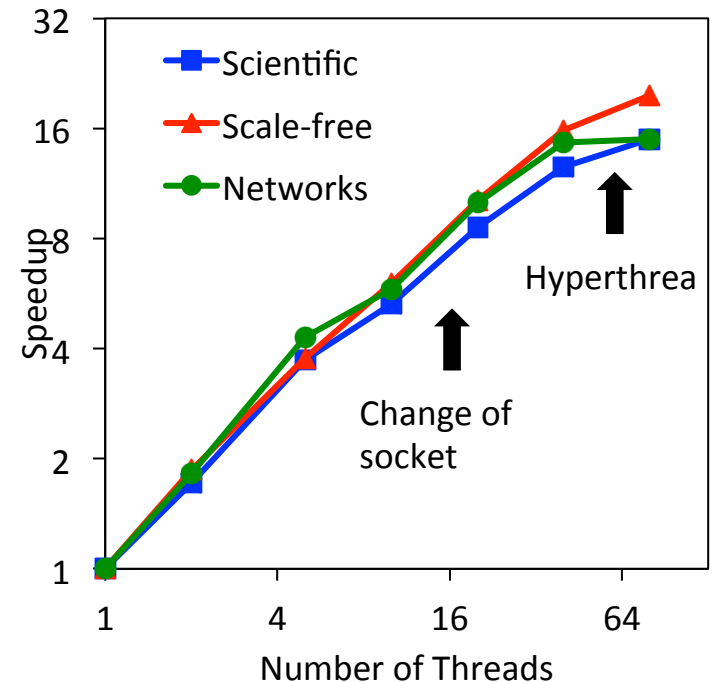
Performance: On 40-core Intel Westmere-EX
On average **7x** faster than current best algorithm. Can be up to **42x** faster.
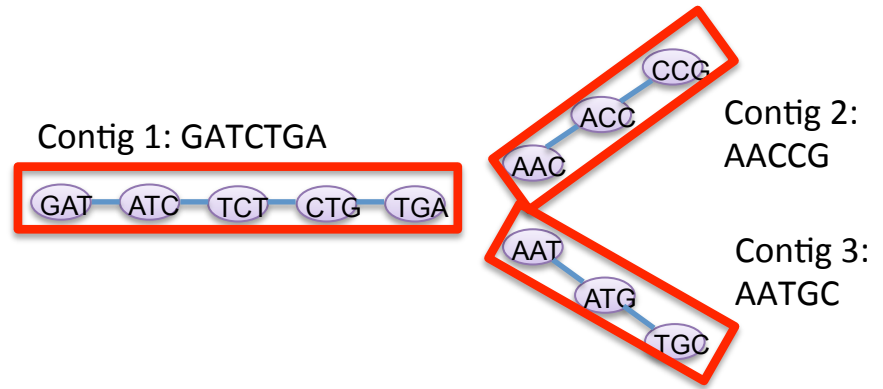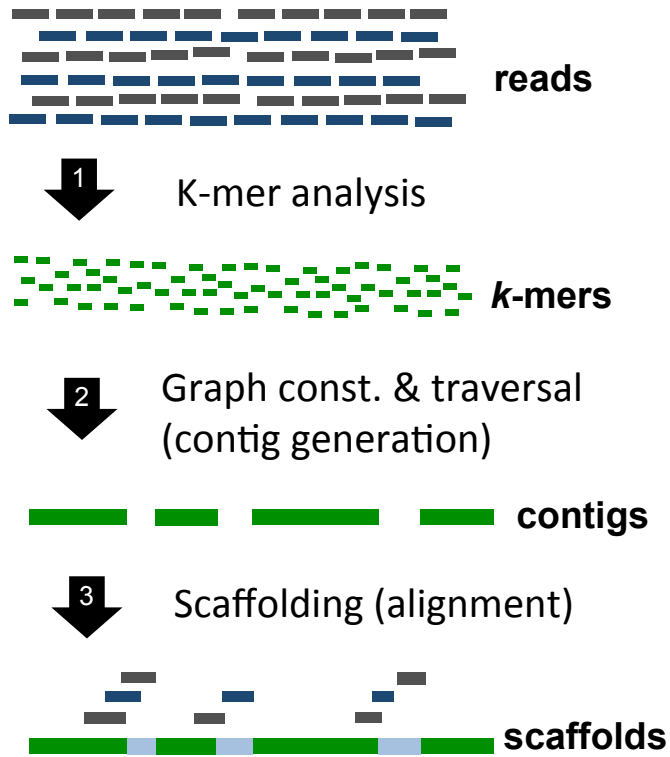

Scaling: One node of Edison (24-core Intel Ivy Bridge). On average **17x** speedups relative to serial algorithm.
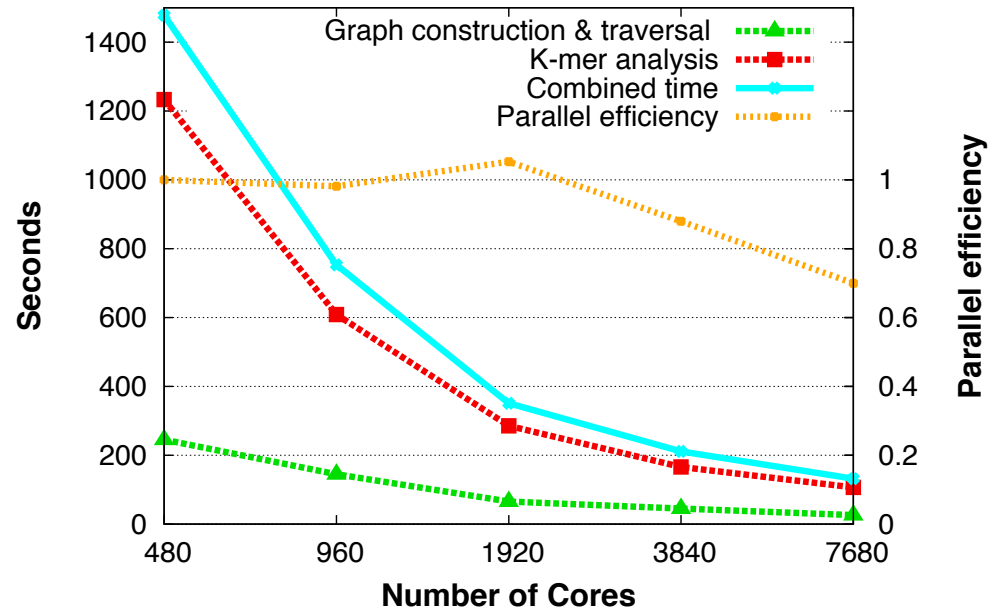

Future direction: expose parallelism suitable for distributed matching (no practical algorithm is available); apply to static pivoting for solving systems of sparse linear equations.
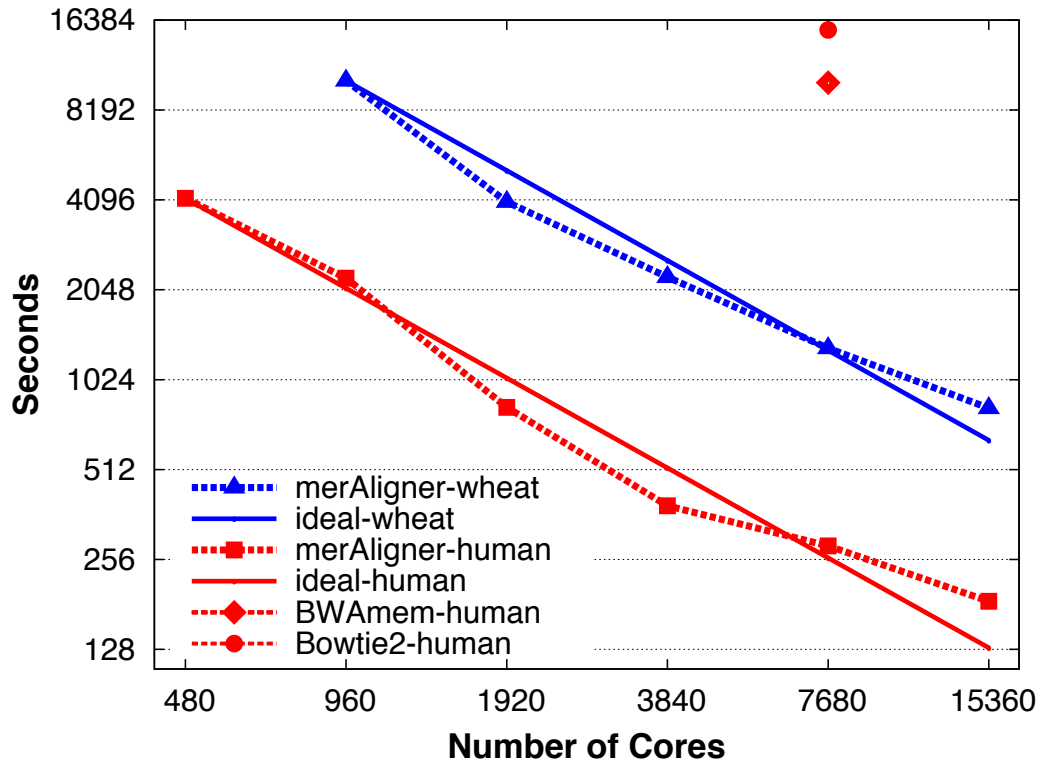
Parallel Algorithms for De Novo Genome Assembly

reads

1 K-mer analysis

k-mers

2 Graph const. & traversal (contig generation)

contigs

3 Scaffolding (alignment)

scaffolds

Contig 1: GATCTGA

GAT — ATC — TCT — CTG — TGA

AAC — ACC — CCG   Contig 2: AACCG

AAT — ATG — TGC   Contig 3: AATGC

Improvement for first two steps:
Human: 60 hours → 2 minutes
Wheat: 140 hours → 15 minutes

Scaling of Steps 1 and 2 on NERSC Edison with Human Genome

Evangelos Georganas, Aydın Buluç, Jarrod Chapman, Leonid Oliker, Daniel Rokhsar, and Katherine Yelick.
Parallel de bruijn graph construction and traversal for de novo genome assembly. In Supercomputing (SC'14).

Parallel Genome Alignment for De novo Assembly

- In de novo assembly, billions of reads must be aligned to contigs
- First aligner to parallelize the seed index construction ("fully" parallel)

Fast Parallel Block Eigensolvers

- Block iterative methods:
  - have data locality and cache re-use
  - expose more parallelism
- Sparse matrix vector block multiplication
- Motivated by nuclear structure calculations
- Applications: data analysis & spectral clustering

$n \times n$ matrix with nnz nonzeroes, in $\beta \times \beta$ blocks

SpMM

SpMM_T
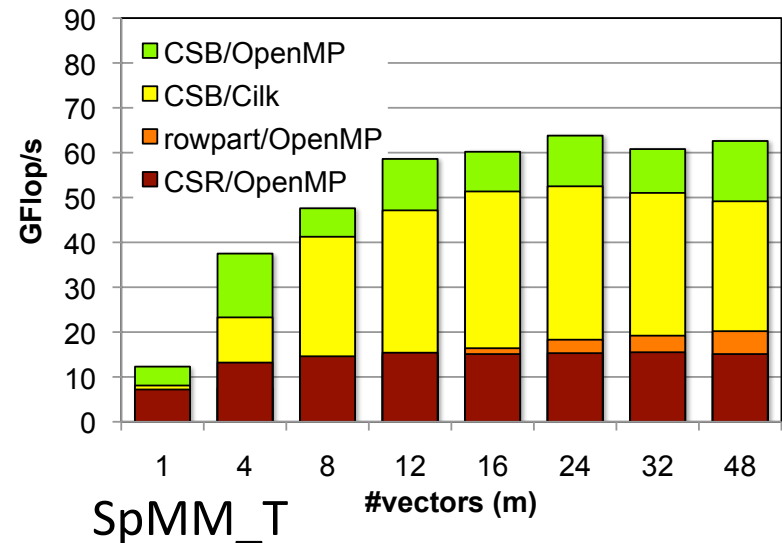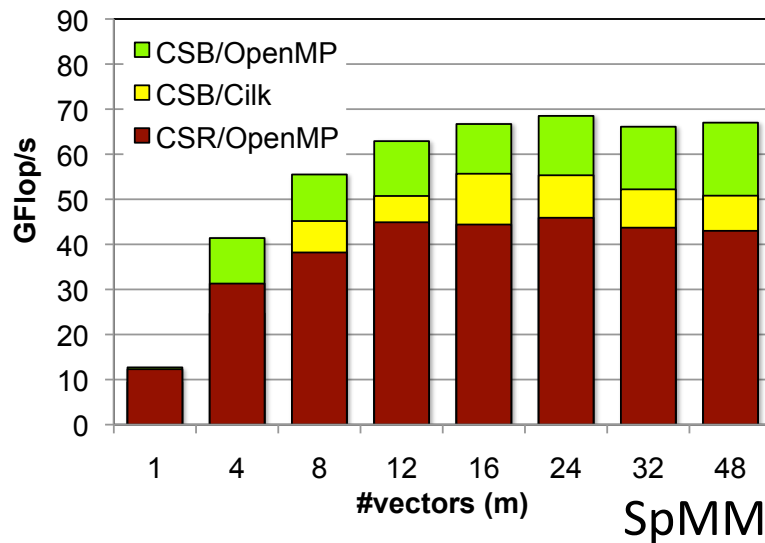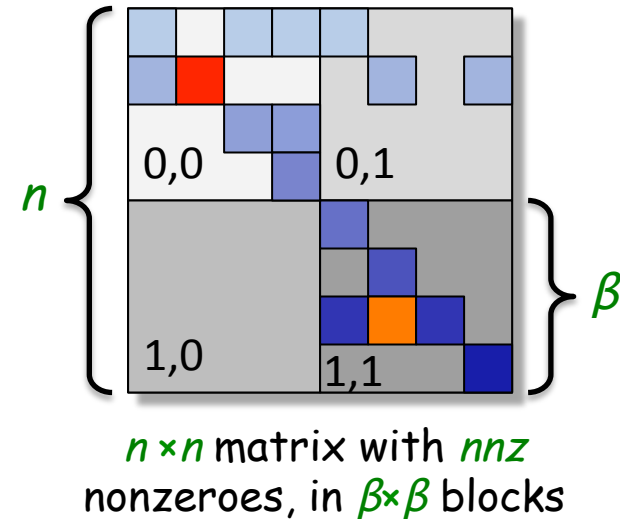
H. Metin Aktulga, Aydın Buluç, Samuel Williams, and Chao Yang. Optimizing sparse matrix-multiple vectors multiplication for nuclear configuration interaction calculations. In *Proceedings of the IPDPS, 2014*

Filtered Semantic Graph Processing
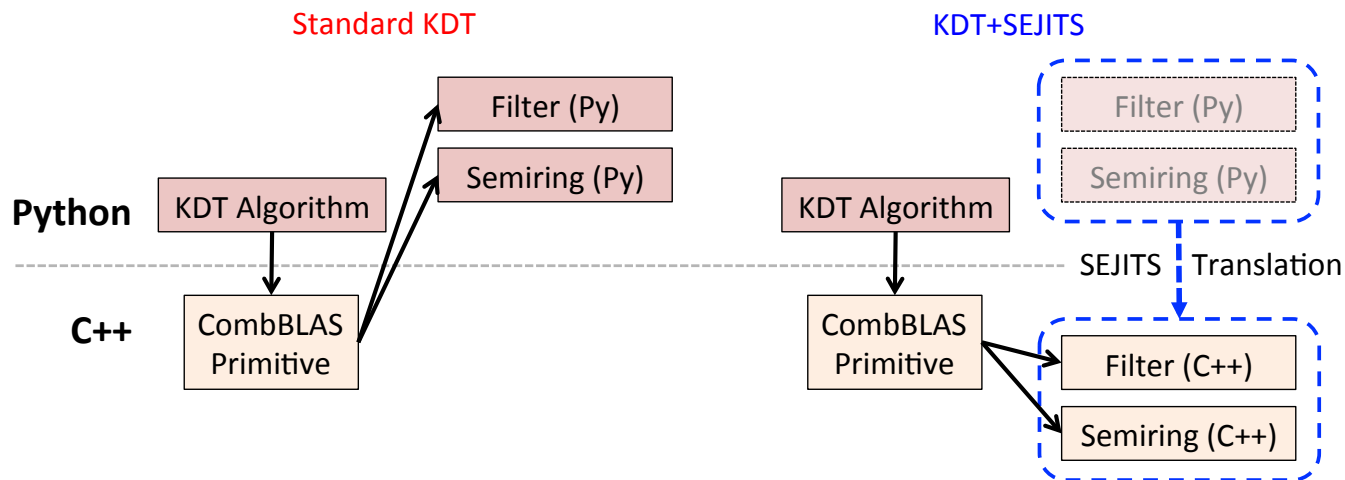
- **Filters** enable efficient processing of semantic graphs (with edge/vertex attributes)
- **Semirings** enable customization of graph algorithms using matrix primitives
- Filters & semirings can be a performance bottleneck in high-level languages
- SEJITS enable just-in-time compilation of filters/semirings into **Combinatorial BLAS,** matching performance low-level code
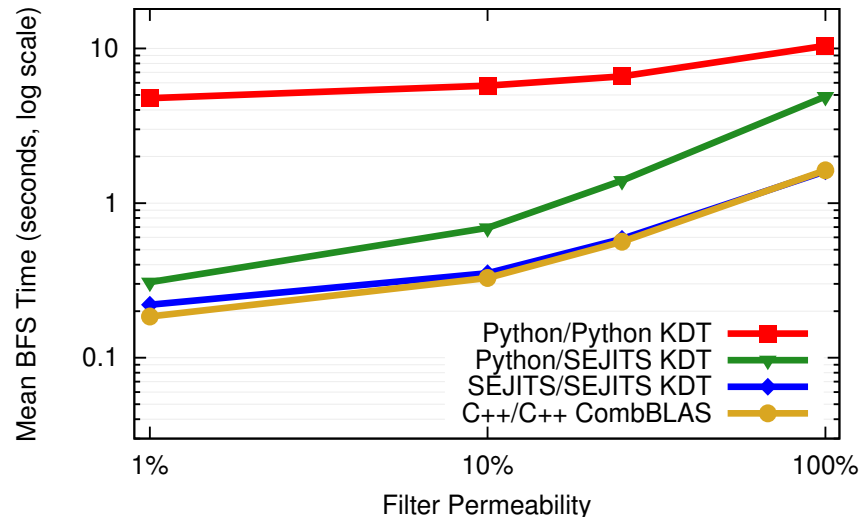
The Graph BLAS Effort

Standards for Graph Algorithm Primitives

Tim Mattson (Intel Corporation), David Bader (Georgia Institute of Technology), Jon Berry (Sandia National Laboratory), Aydin Buluc (Lawrence Berkeley National Laboratory), Jack Dongarra (University of Tennessee), Christos Faloutsos (Carnegie Melon University), John Feo (Pacific Northwest National Laboratory), John Gilbert (University of California at Santa Barbara), Joseph Gonzalez (University of California at Berkeley), Bruce Hendrickson (Sandia National Laboratory), Jeremy Kepner (Massachusetts Institute of Technology), Charles Leiserson (Massachusetts Institute of Technology), Andrew Lumsdaine (Indiana University), David Padua (University of Illinois at Urbana-Champaign), Stephen Poole (Oak Ridge National Laboratory), Steve Reinhardt (Cray Corporation), Mike Stonebraker (Massachusetts Institute of Technology), Steve Wallach (Convey Corporation), Andrew Yoo (Lawrence Livermore National Laboratory)

Abstract-- It is our view that the state of the art in constructing a large collection of graph algorithms in terms of linear algebraic operations is mature enough to support the emergence of a standard set of primitive building blocks. This paper is a position paper defining the problem and announcing our intention to launch an open effort to define this standard.

- The Graph BLAS Forum: http://istc-bigdata.org/GraphBlas/
- Graph Algorithms Building Blocks (GABB workshop at IPDPS'14 and IPDPS'15): http://www.graphanalysis.org/workshop2015.html
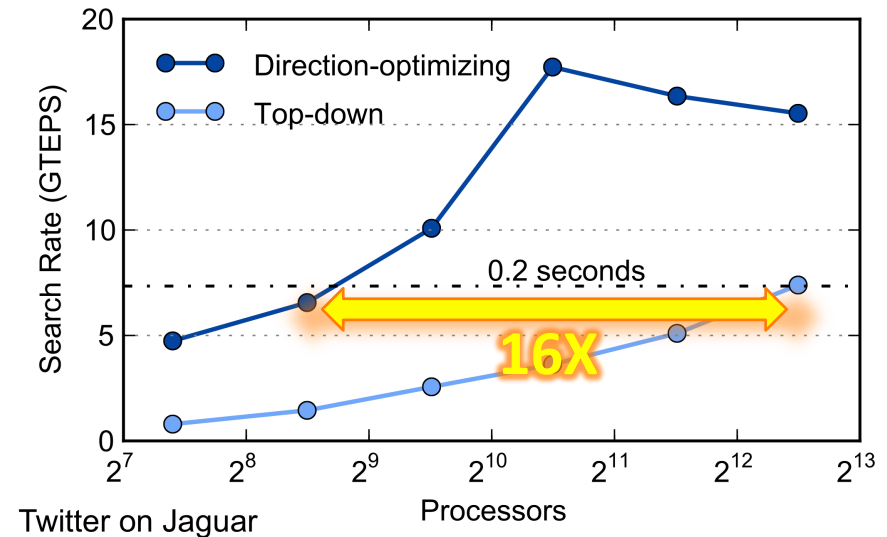
Direction-optimizing BFS on distributed memory

Up to 8X faster on Kronecker (Graph500) inputs (largest run on 16 billion vertices and 256 billion edges)
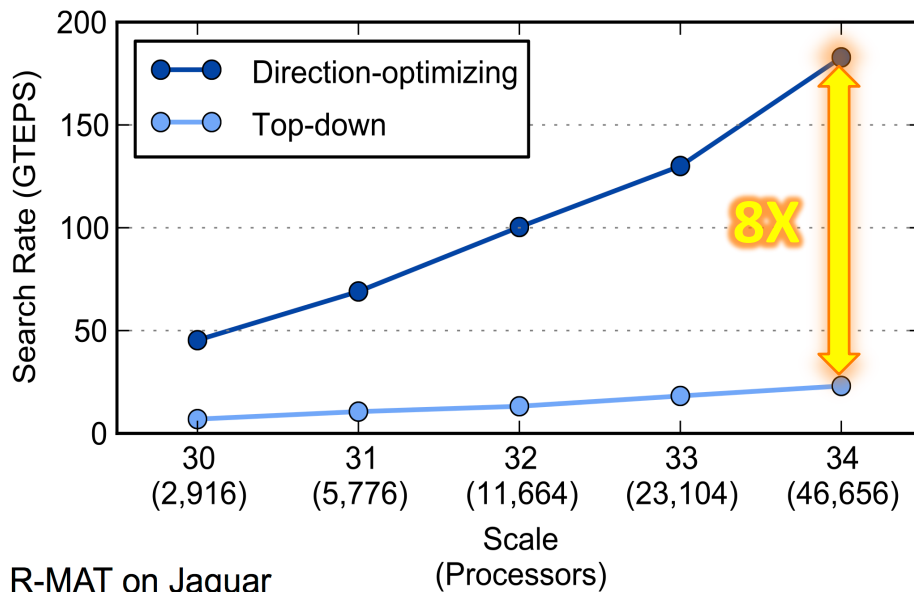
For a fixed-sized real input, direction-optimizing algorithm needs **1/16th of processors (and energy)**

R-MAT on Jaguar

Twitter on Jaguar

Implemented on top of **Combinatorial BLAS** (i.e. uses 2D decomposition)

Buluç, Beamer, Madduri, Asanović, Patterson, "Distributed-Memory Breadth-First Search on Massive Graphs", Chapter in Parallel Graph Algorithms, Bader (editor), CRC Press / Taylor & Francis (2015), to appear
Beamer, B., Asanović, Patterson, "Distributed Memory Breadth-First Search Revisited: Enabling Bottom-Up Search", IPDPSW'13

Communication-avoiding All-Pairs Shortest-Paths

V1   **V2**

2x speedup

6.2x speedup

n=8192

n=4096
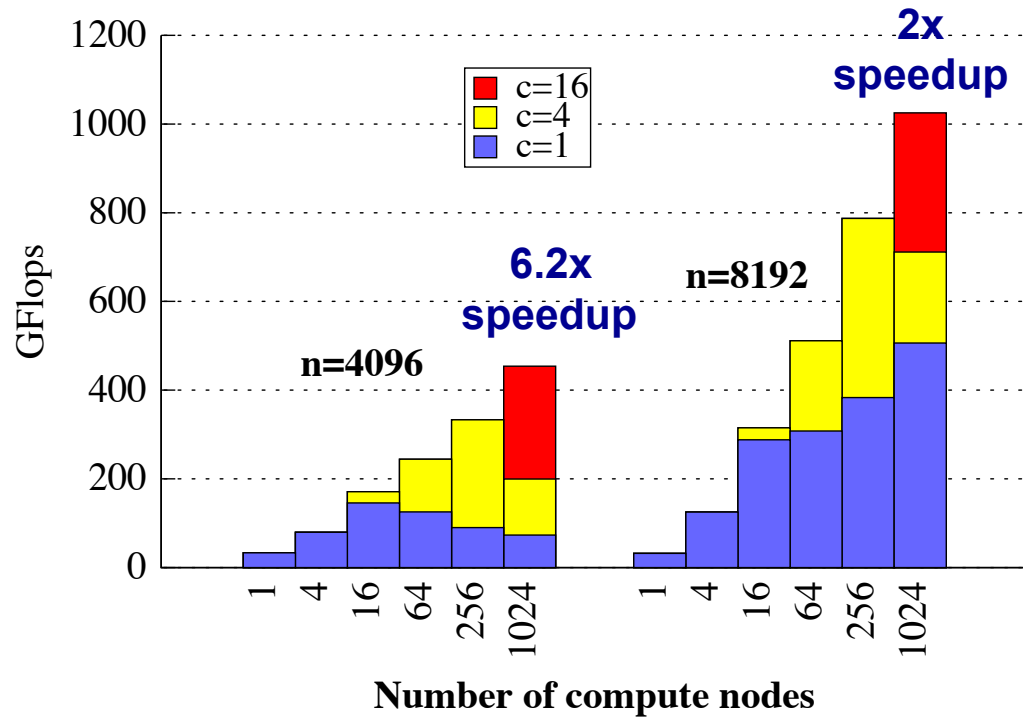
Strong Scaling on Hopper (Cray XE6 up to 1024 nodes = 24576 cores)

Algorithm based on Kleene's recursive formulation on the (min,+) semiring
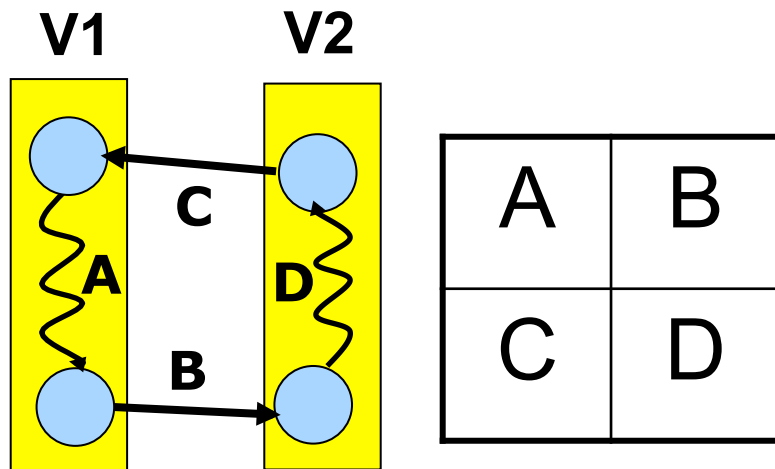
+ is "min",   **×** is "add"

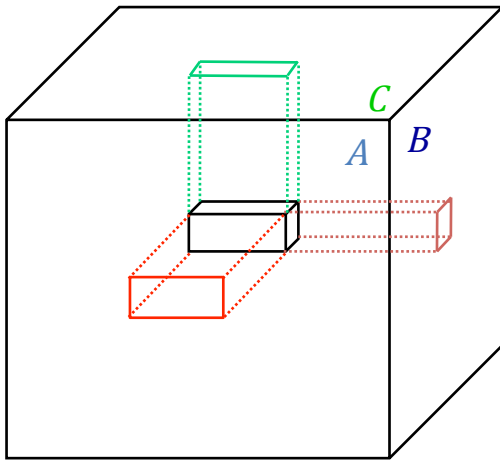|	
A	B
C	D

A = A*;   % recursive call

B = AB; C = CA;

D = D + CB;

D = D*;   % recursive call

B = BD; C = DC;

A = A + BC;

Solomonik, Buluç, Demmel. "Minimizing communication in all-pairs shortest paths", IPDPS. 2013.

Communication Optimal Sparse Matrix Multiplication

Matrix multiplication:
$$\forall (i,j) \in n \times n,$$
$$C(i,j) = \Sigma_k A(i,k)B(k,j),$$

Assumption: Assignment of data and work to processors is sparsity-pattern-independent

Previous Sparse Classical Lower Bound:

$$\Omega\left(\frac{\#FLOPs}{\left(\sqrt{M}\right)^3} \cdot \frac{M}{P} \right) = \Omega\left(\frac{d^2 n}{P\sqrt{M}} \right)$$

[Ballard, et al. SIMAX'11]

New Lower bound for Erdős-Rényi(n,d) :

$$\Omega\left(\min\left\{ \frac{dn}{\sqrt{P}}, \frac{d^2 n}{P} \right\} \right) \checkmark$$

- Expected (Under some technical assumptions)
- *Two new algorithms (3D iterative & recursive)* that attain the new lower bound
- No previous algorithm attain these.

Ballard, Buluç, Demmel, Grigori, Lipshitz, Schwartz, and Toledo. Communication optimal parallel multiplication of sparse random matrices. In SPAA 2013.