EDGAR

- **E**nergy-efficient **D**ata and **G**raph **A**lgorithms **R**esearch
- Funded by Applied Math, ASCR
- Early Career Research Program (start: 2013)
- PI: Aydın Buluç (Berkeley Lab)
- Postdoctoral Fellow:
  - Ariful Azad (100%)
- Students:
  - Veronika Strnadova-Neeley (Since Oct 2015, UCSB)
  - Adam Sealfon (CSGF Fellow, Summer 2015, MIT)
  - Chaitanya Aluru (undergraduate, UC Berkeley)

EDGAR (FY15)

Awards:

- Aydın Buluç, ***IEEE TCSC Award for Excellence for Early Career Research*** by the IEEE Committee on Scalable Computing, 2015
- **HipMer team** (next slide), HPCWire's Readers' Choice Award for the ***Best Use of HPC Application in Life Sciences***

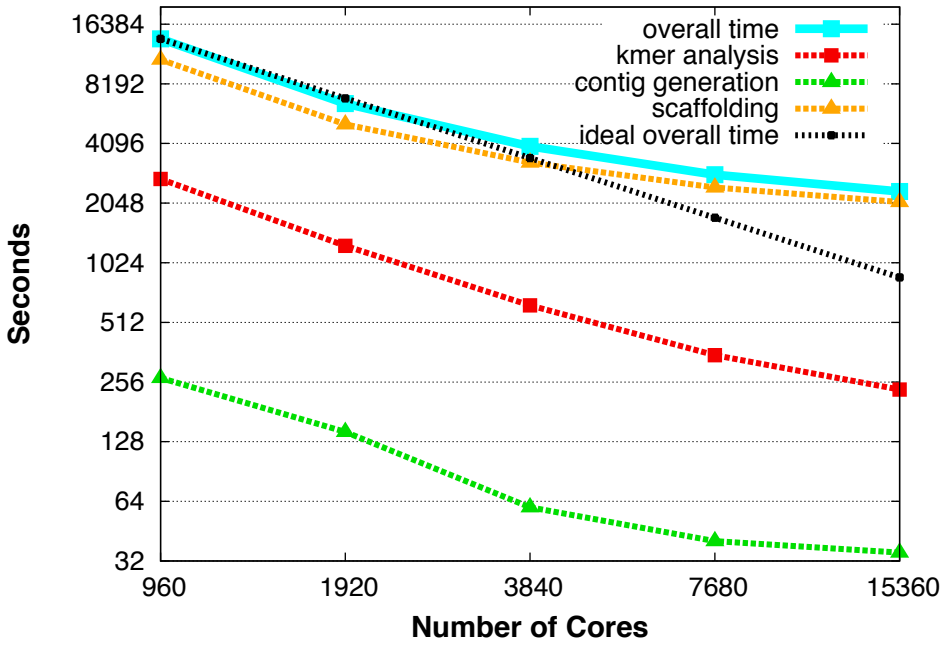
Artifacts:

- Aydın Buluç, ***Guest Editor***: Parallel Computing, special issue on "Graph Analysis for Scientific Discovery"
- Six peer-reviewed publications, one invited article
- Six invited talks (one at conference, five at universities/labs)

HipMer: An Extreme-Scale De Novo Genome Assembler

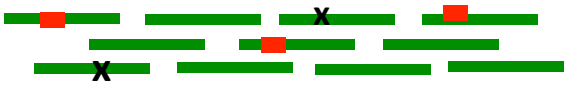
Meraculous assembler is used in production at the Joint Genome Institute

- Wheat assembly is a "grand challenge"
- Hardest part is contig generation (large in-memory *hash table* that represents graph)
- HipMer is an efficient parallelization of Meraculous



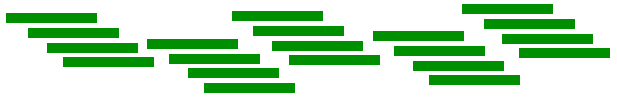
Meraculous Assembly Pipeline

reads



New fast & parallel I/O

k-mers



New k-mer analysis filters errors using probabilistic "Bloom Filter"

contigs

Graph algorithm (connected components) scales to 15K cores on NERSC's Edison

scaffolding using scalable alignment

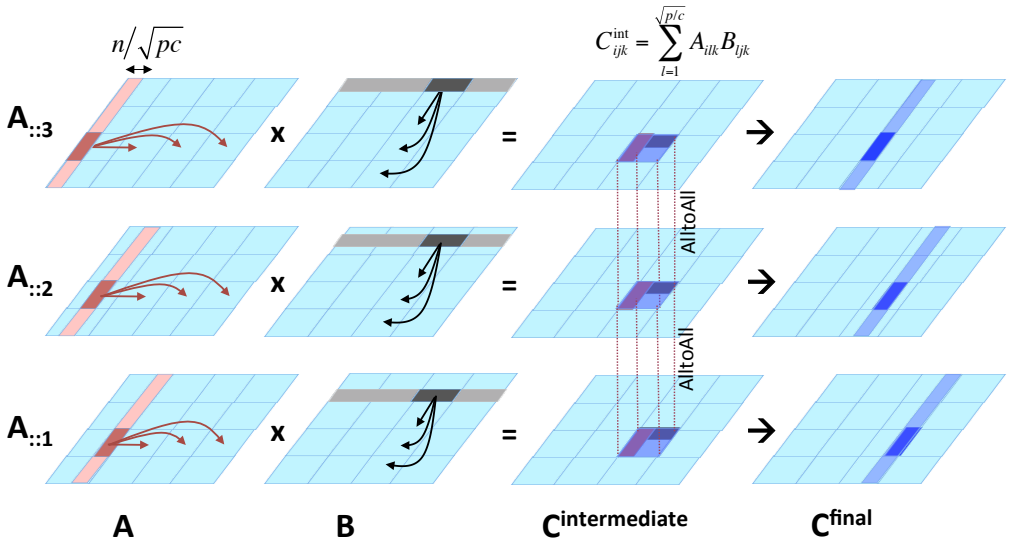
Performance improvement from ***days*** *to* ***minutes***

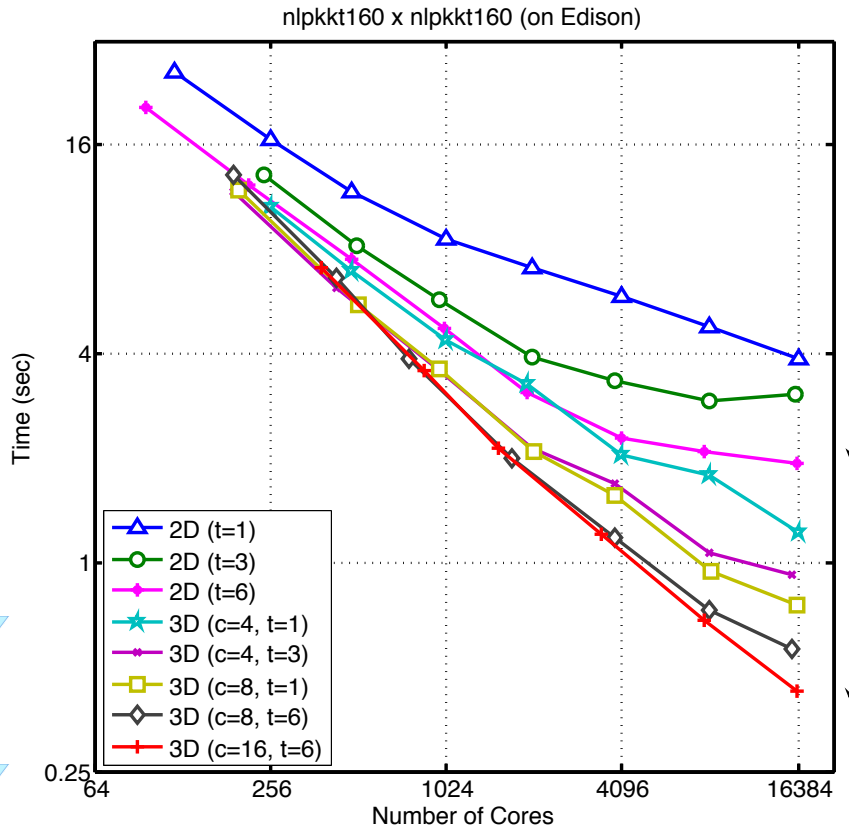
E. Georganas, **A. Buluç**, J. Chapman, S. Hofmeyr, **C. Aluru**, R. Egan, L. Oliker, D. Rokhsar, K. Yelick. HiPMer: An extreme-scale de novo genome assembler. SC'15

Communication-Avoiding Sparse Matrix-Matrix Multiply

Applications:

- Algebraic multigrid (AMG) restriction
- Graph computations
- Quantum chemistry
- Similarity computation (data mining)
- Interior-point optimization



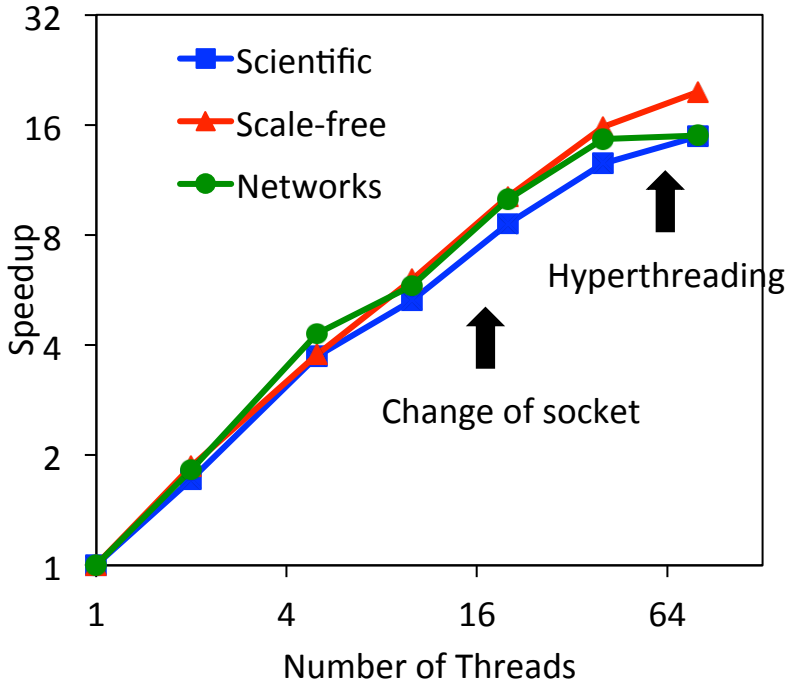


3D-threaded (red) beats the previous state-of-the-art (blue) by 8X at large concurrencies

A. Azad, G. Ballard, **A. Buluç**, J. Demmel, L. Grigori, O. Schwartz, S. Toledo, S. Williams. Exploiting multiple levels of parallelism in sparse matrix-matrix multiplication. arXiv preprint arXiv:1510.00844.

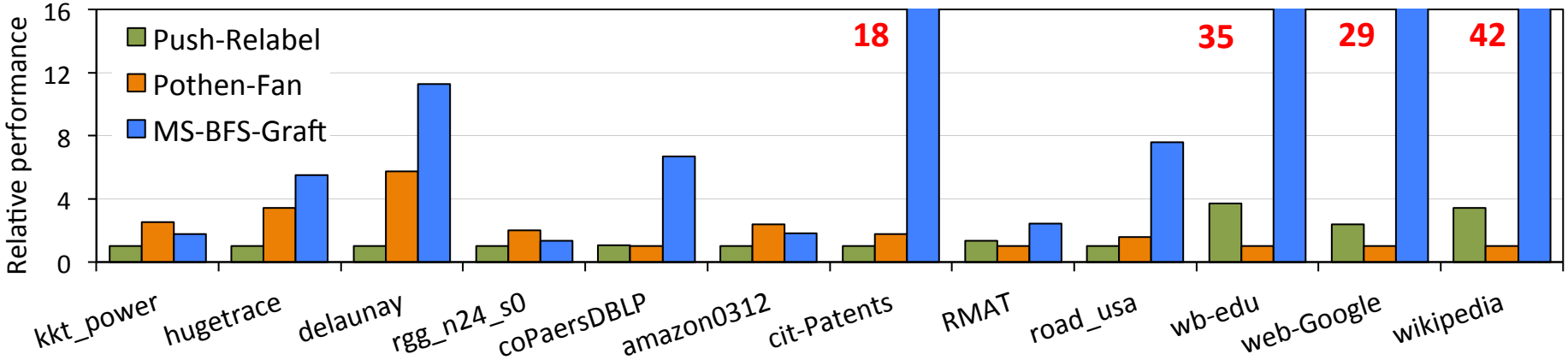
Parallel Maximum Cardinality Matching in Bipartite Graphs Using "Tree Grafting"

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



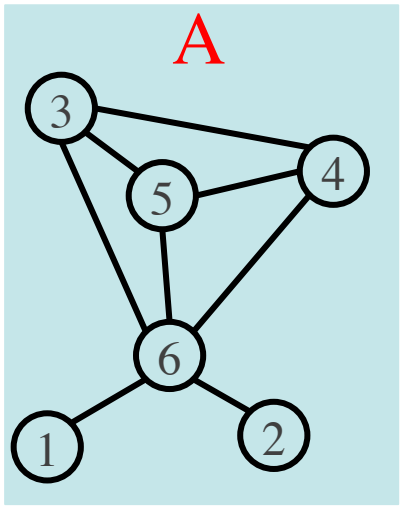
Performance: On 40-core Intel

On average **7x** faster than current best algorithm. Can be up to **42x** faster.



A. Azad, A.Pothen, **A. Buluç**. Computing Maximum Cardinality Matchings in Parallel on Bipartite Graphs via Tree-Grafting. Under revision in IEEE Transactions on Parallel and Distributed Systems (TPDS).

Counting and Enumerating Triangles using Matrix Algebra

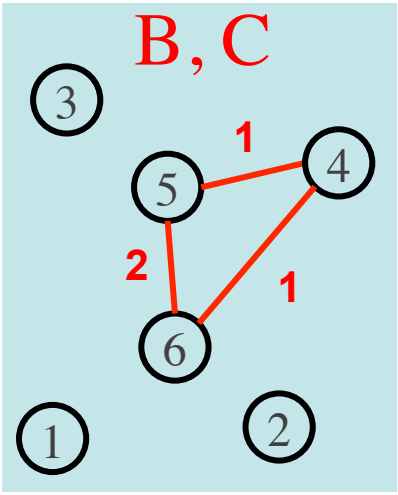


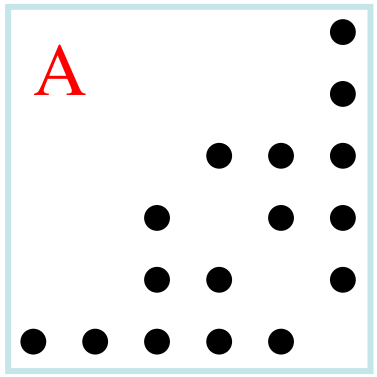
$A = L + U$ (hi->lo + lo->hi)

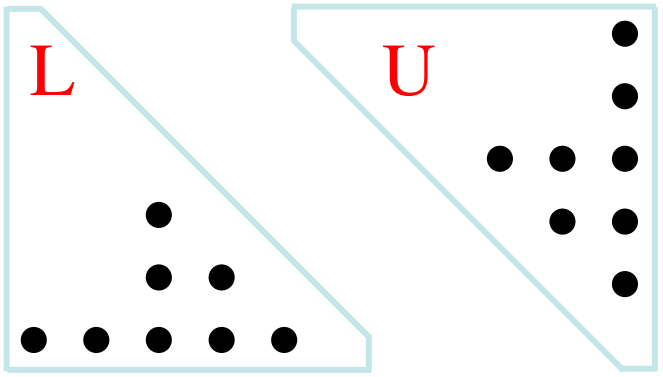
$L \times U = B$ (wedge, low hinge)

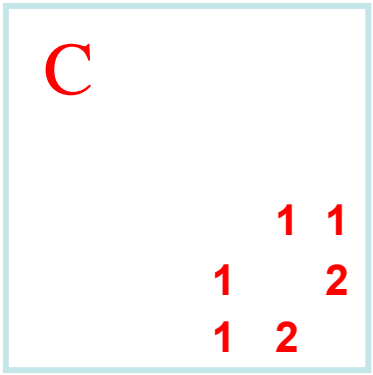
$A \wedge B = C$ (closed wedge)

sum(C)/2 = **4 triangles**





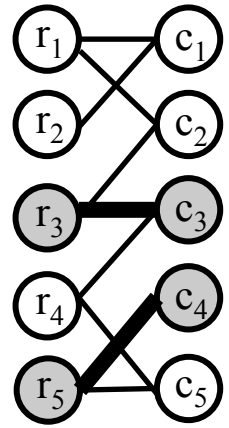




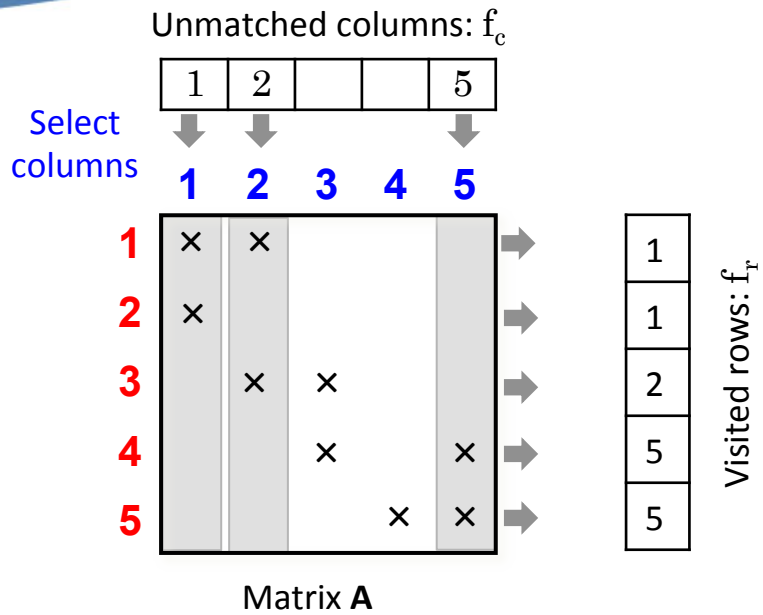
A. Azad, **A. Buluç**, and J. R Gilbert. "Parallel triangle counting and enumeration using matrix algebra", *Graph Algorithm Building Blocks IPDPS Workshops, 2015*

Maximal Cardinality Matching using Matrix Algebra

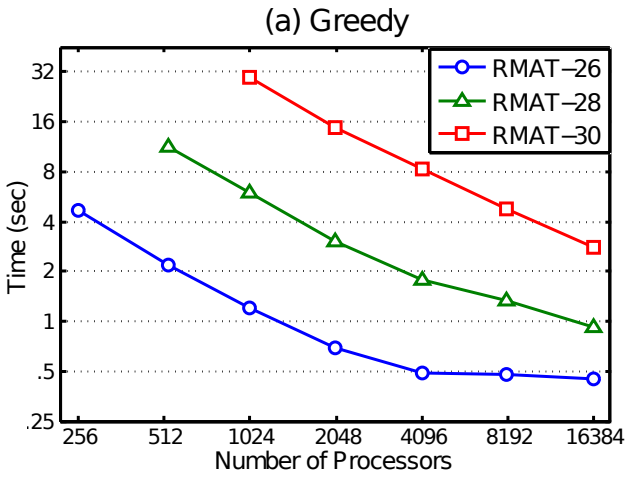
Matrix-based primitives enable efficient and scalable distributed-memory implementations of various maximal cardinality algorithms solely by minimal modifications to the underlying semiring operator.

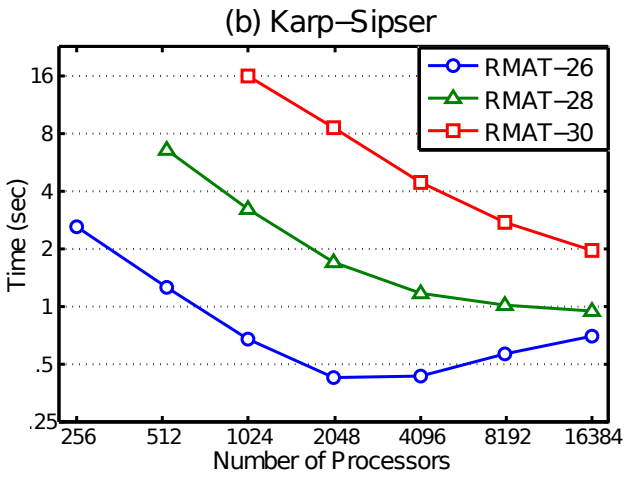


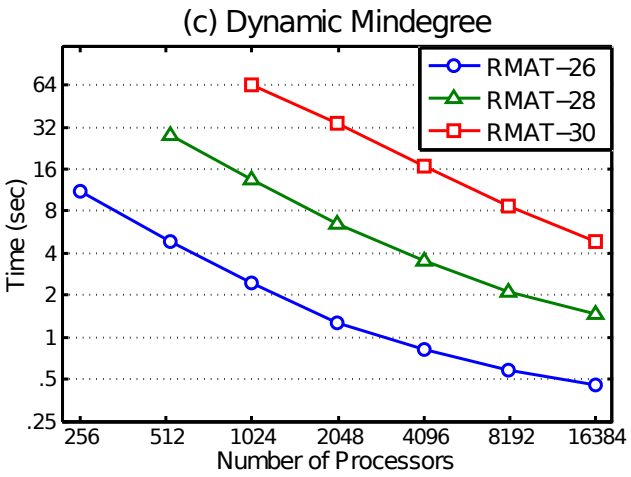
Bipartite graph
$G(R, C, E)$



Matrix **A**







A. Azad, A Buluç. Distributed-memory algorithms for maximal cardinality matching using matrix algebra. In *IEEE International Conference on Cluster Computing (CLUSTER)*, 2015 (full paper).