Sparse Matrix Partitioning for Parallel Eigenanalysis of Large Static and Dynamic Graphs

Michael Wolf, Sandia National Laboratories

Ben Miller, MIT Lincoln Laboratory

IEEE HPEC 2014

September 10, 2014

Sandia National Laboratories

Exceptional service in the national interest

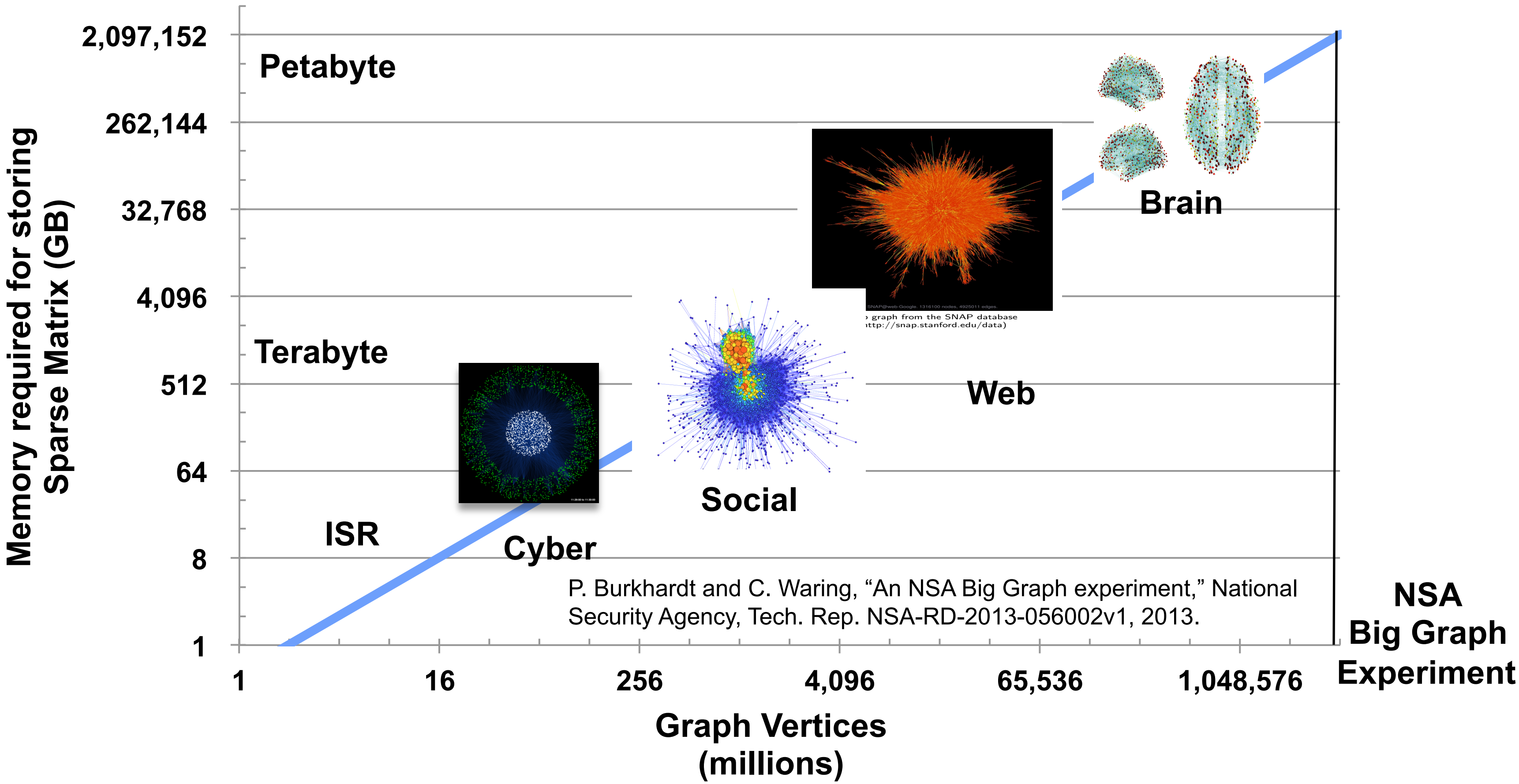
U.S. DEPARTMENT OF ENERGY

National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2014-17387 PE

The Lincoln Laboratory portion of this work is sponsored by the Intelligence Advanced Research Projects Activity (IARPA) under Air Force Contract FA8721-05-C-0002. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA or the U.S. Government.

Big Data and High Performance Computing

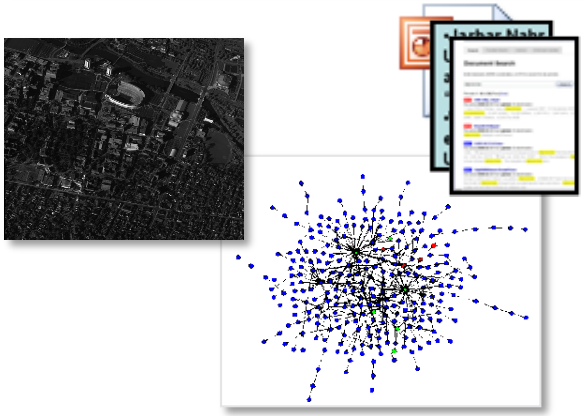


Use high performance computing to address compute challenges posed by problem scales of interest to DoD/IC

Motivating Graph Analytics Applications

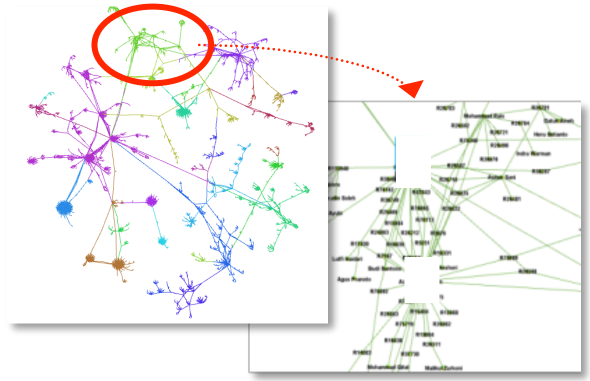
Sandia National Laboratories

ISR



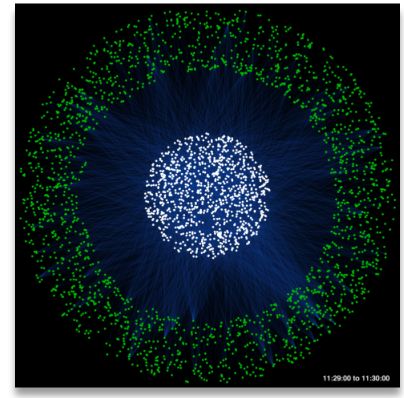
- Graphs represent entities and relationships detected through multiple sources
- 1,000s – 1,000,000s tracks and locations
- GOAL: Identify anomalous patterns of life

Social



- Graphs represent relationships between individuals or documents
- 10,000s – 10,000,000s individual and interactions
- GOAL: Identify hidden social networks

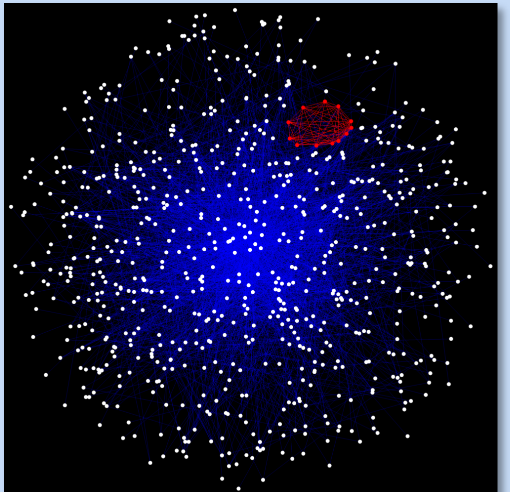
Cyber



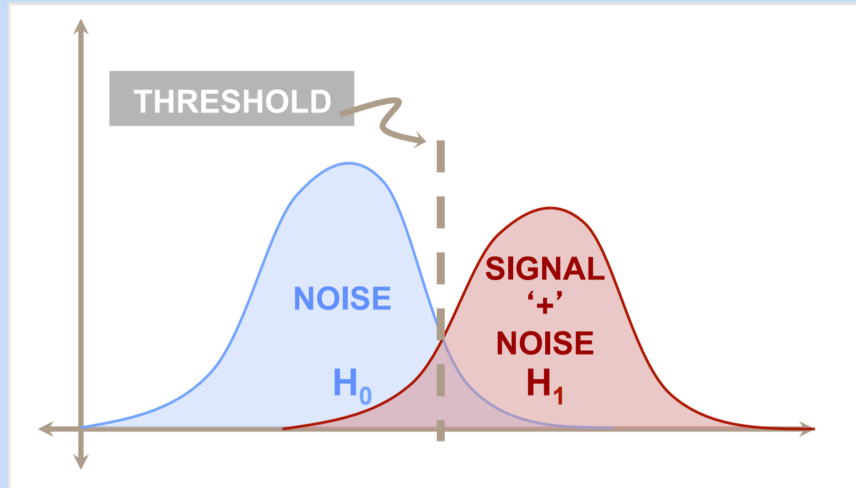
- Graphs represent communication patterns of computers on a network
- 1,000,000s – 1,000,000,000s network events
- GOAL: Detect cyber attacks or malicious software

Detection of anomalies in massive datasets (very large graphs)

Statistical Detection Framework for Graphs



Graph Theory



Detection Theory

Signal Processing for Graphs (SPG)

Develop fundamental graph signal processing concepts

Demonstrate in simulation

Apply to real data

Computational Focus: Dimensionality Reduction

Temporal Integration → Graph Model Construction → [Dimensionality Reduction: Residual Decomposition → Component Selection] → Anomaly Detection → Identification

Eigensystem

$$B = (A - E[A])$$

Solve:

$$Bx_i = \lambda_i x_i, i = 1, \ldots, m$$

Example: Modularity Matrix

$$E[A_s] = k\,k^T / (2|e|)$$

$|e|$ – Number of edges in graph $G(A)$

k – degree vector

k_i = degree(v_i), $v_i \in G(A)$

- Dimensionality reduction dominates SPG computation
- Eigen decomposition is key computational kernel
- Parallel implementation required for very large graph problems
  - Fit into memory, minimize runtime

Need fast parallel eigensolvers

Outline

- Anomaly Detection in Very Large Graphs
- Eigenanalysis and Performance Challenges
- Improving Sparse Matrix-Vector Multiplication (SpMV) Performance through Data Partitioning
- Partitioning: Dynamic Graphs and Sampling
- Summary

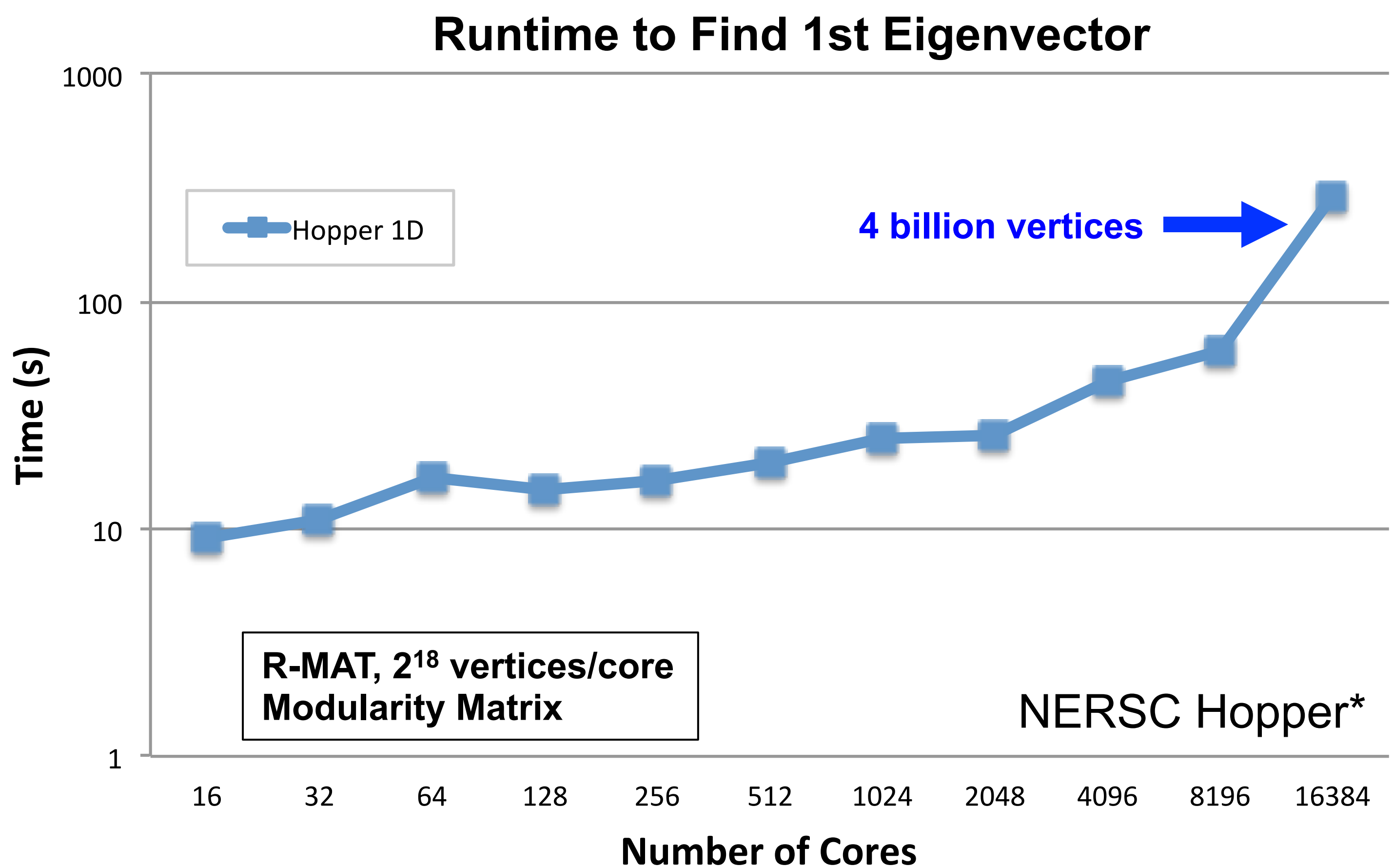
Dimensionality Reduction: Parallel Implementation

- Using Anasazi (Trilinos) Eigensolver
  - Block Krylov-Schur
  - Eigenpairs corresponding to eigenvalues with largest real component
  - User defined operators (don't form matrix explicitly)

- Initial Numerical Experiments
  - R-Mat (a=0.5, b=0.125, c=0.125, d=0.25)
    - Average nonzeros per row: 8
    - Number of rows: 2^{22} to 2^{32}
  - Two systems
    - Hopper* (NERSC) -- Cray XE6 supercomputer
    - LLGrid (MIT LL) – compute cluster (10 GB ethernet)
  - Initially: 1D random row distribution (good load balance)

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Weak Scaling Eigensolver

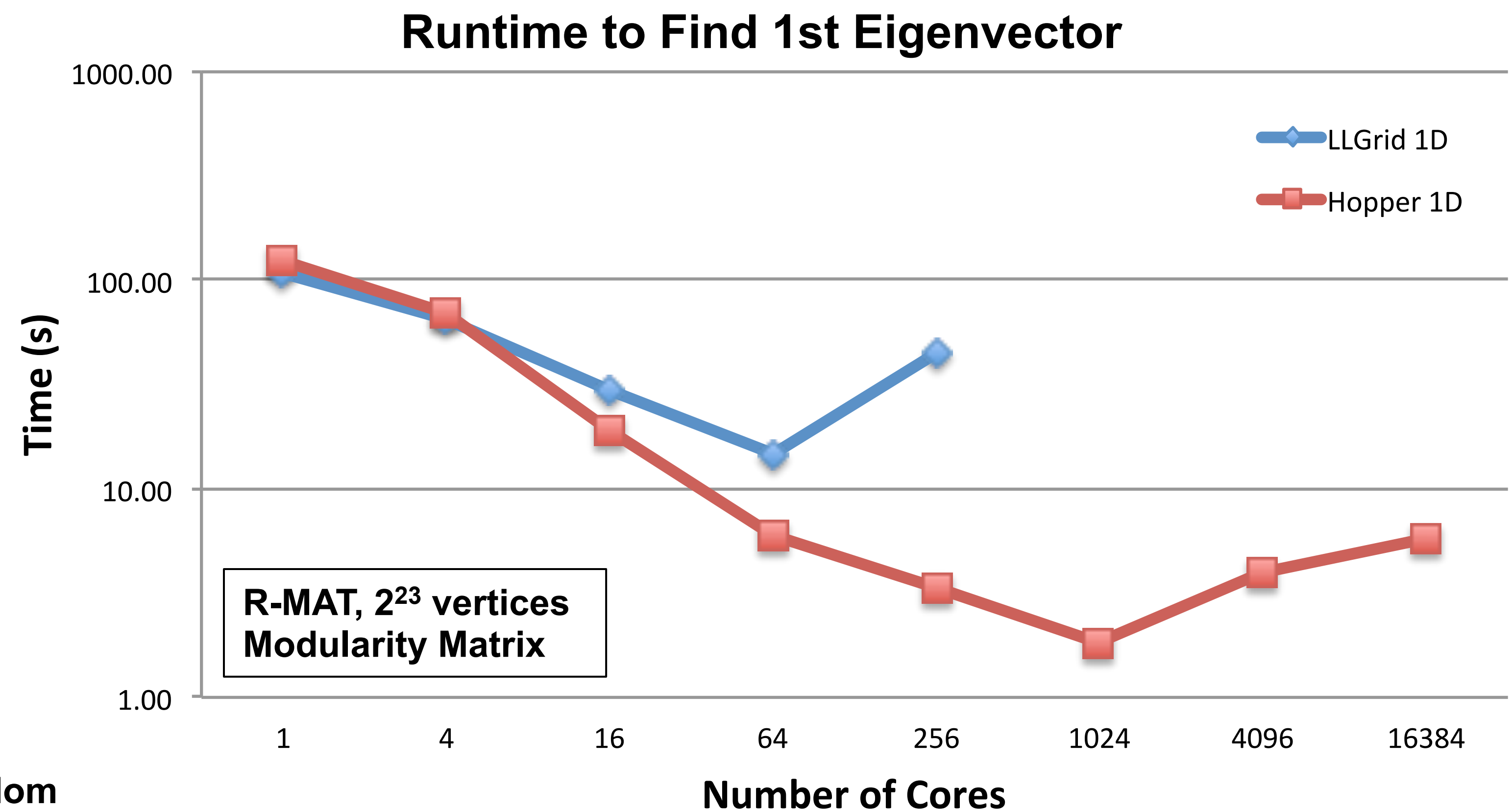


1D random partitioning

Solved system for up to 4 billion vertex graph

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Strong Scaling: Eigensolver

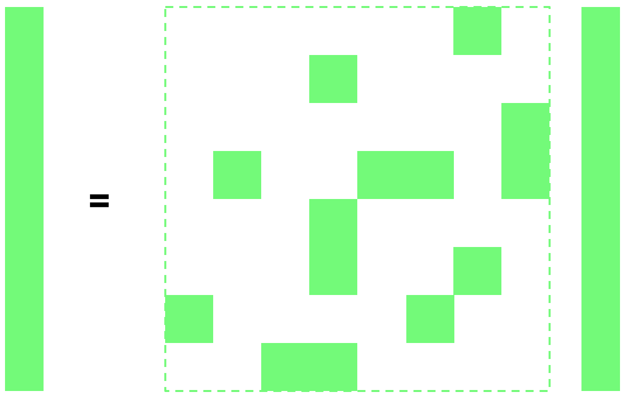


1D random partitioning

Scalability limited and runtime increases for large numbers of cores

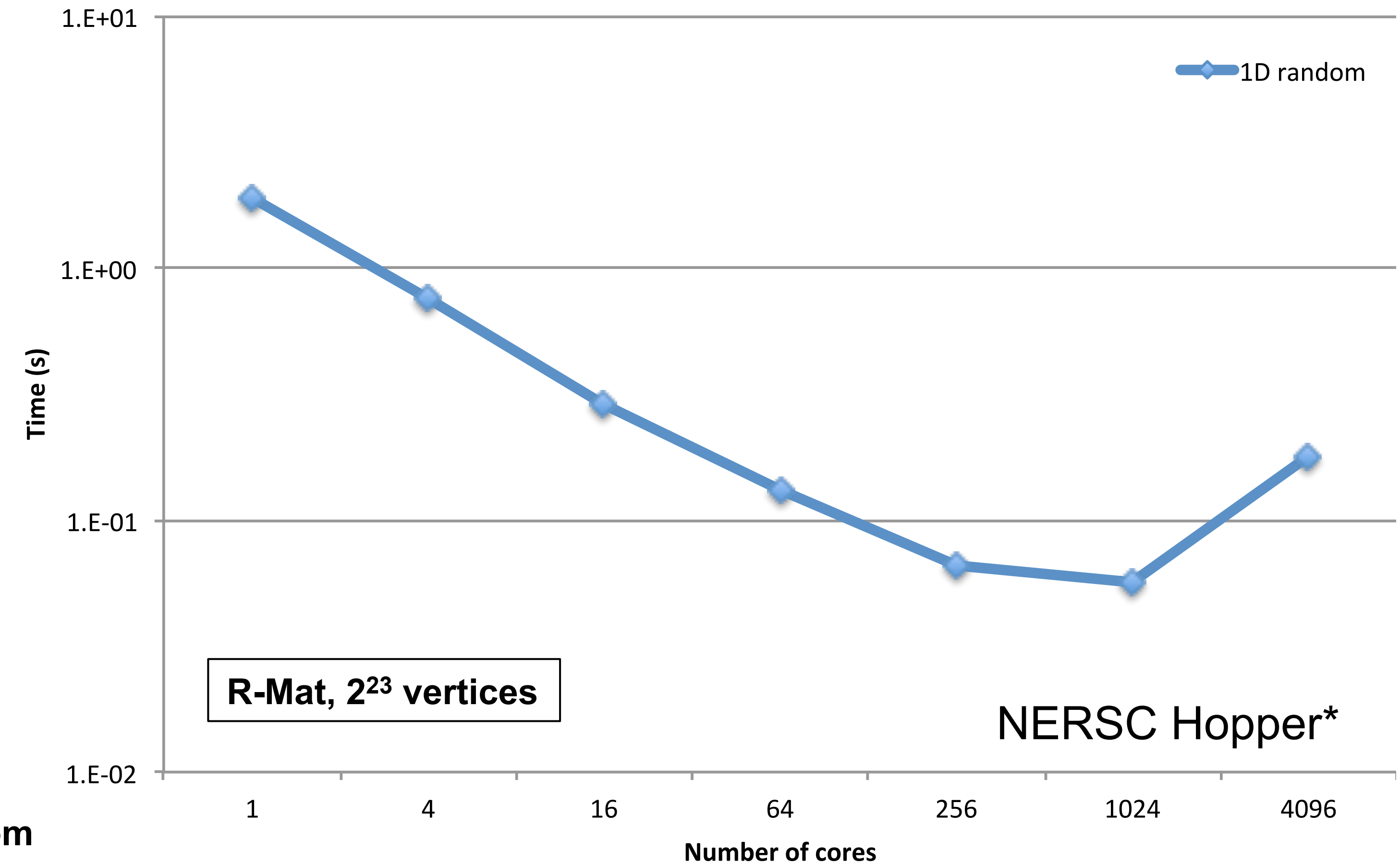
* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Sparse Matrix-Vector Multiplication



- Sparse matrix-dense vector multiplication (SpMV) key computational kernel in eigensolver
- Performance of SpMV challenging for matrices resulting from power-law graphs
  - Load imbalance
  - Irregular communication
  - Little data locality
- Important to improve performance of SpMV

Strong Scaling: SpMV



1D random partitioning

Scalability limited and runtime increases for large numbers of cores

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Outline

- Anomaly Detection in Very Large Graphs
- Eigenanalysis and Performance Challenges
- Improving Sparse Matrix-Vector Multiplication (SpMV) Performance through Data Partitioning
- Partitioning: Dynamic Graphs and Sampling
- Summary

Data Partitioning to Improve SpMV

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \end{bmatrix} = \begin{bmatrix} 1 & 6 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 1 & 9 & 0 & 5 & 0 & 0 & 0 \\ 0 & 8 & 1 & 7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 & 0 & 0 & 0 & 7 \\ 0 & 0 & 0 & 0 & 1 & 8 & 0 & 0 \\ 4 & 0 & 0 & 0 & 3 & 1 & 3 & 0 \\ 0 & 0 & 0 & 6 & 0 & 9 & 1 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 4 \\ 3 \\ 1 \\ 4 \\ 2 \\ 1 \end{bmatrix}$$

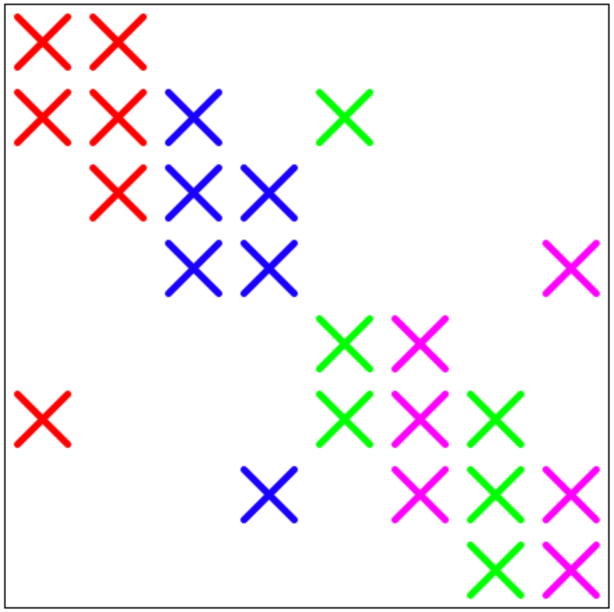
$$\mathbf{y} = \mathbf{A}\mathbf{x}$$

- Partition matrix nonzeros
- Partition vector elements

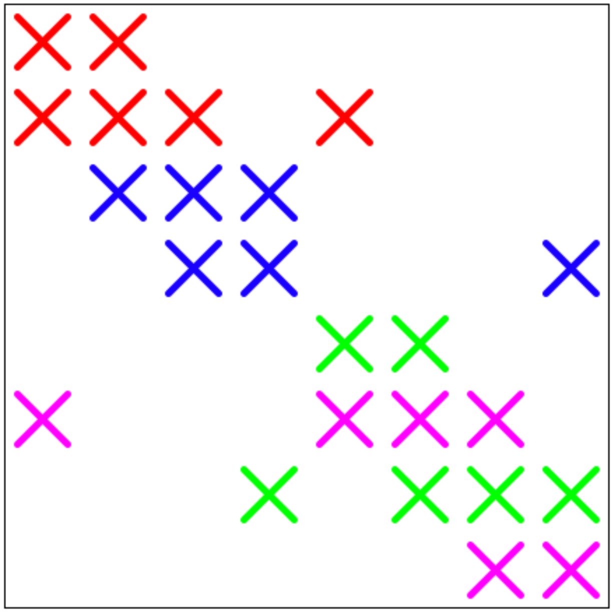
Partitioning Objective

- Ideally we minimize total execution time of SpMV
- Settle for easier objectives
  - Balance computational work
  - Minimize communication metric
    - Total communication volume
    - Number of messages
- Can Partition matrices in different ways
  - 1D
  - 2D
- Can model problem in different ways
  - Graph
  - Bipartite graph
  - Hypergraph

1D Partitioning



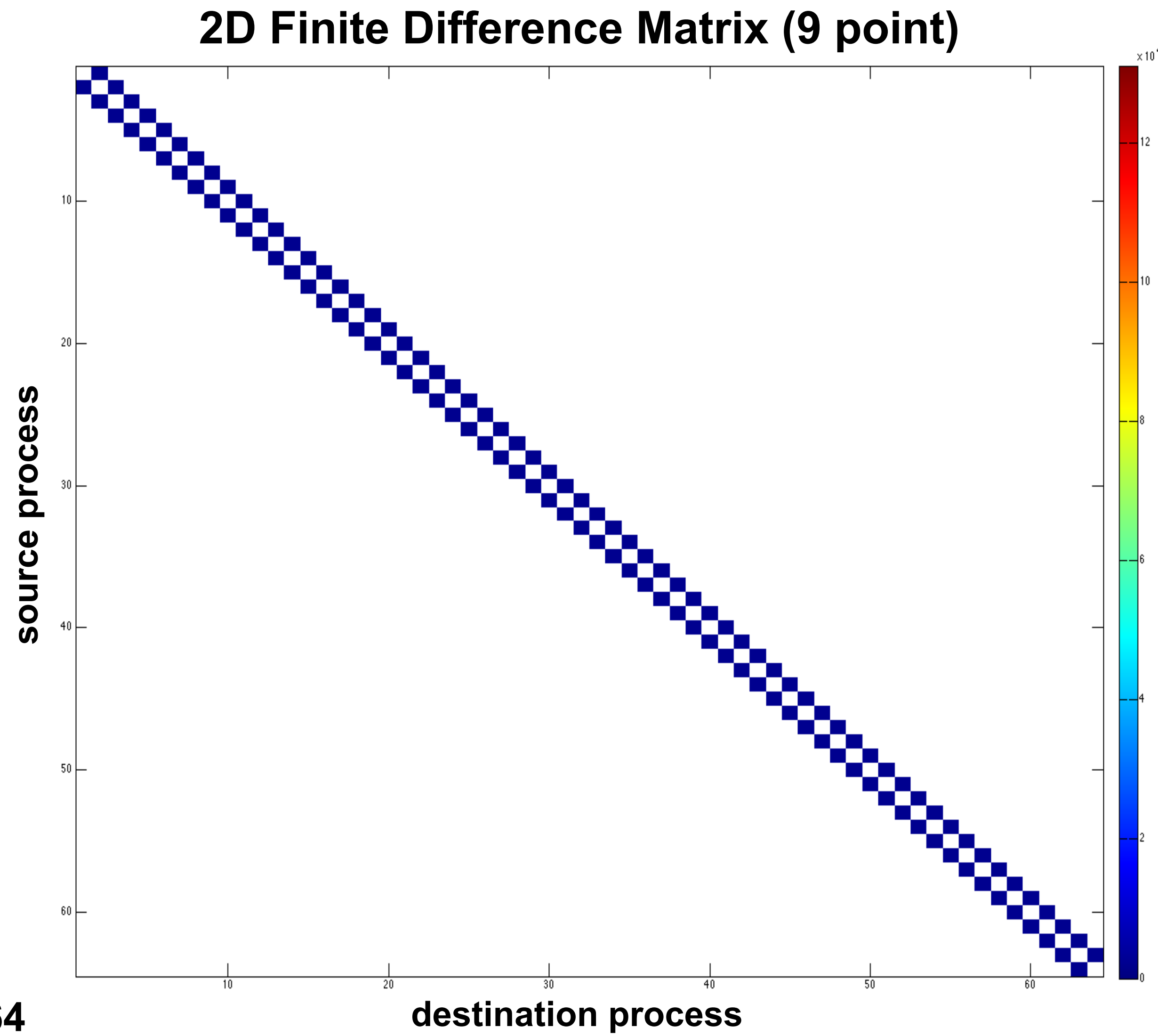
1D Column



1D Row

- Each process assigned nonzeros for set of columns
- Each process assigned nonzeros for set of rows

Communication Pattern: 1D Block Partitioning



P=64

Number of Rows: 2^{23}
Nonzeros/Row: 9

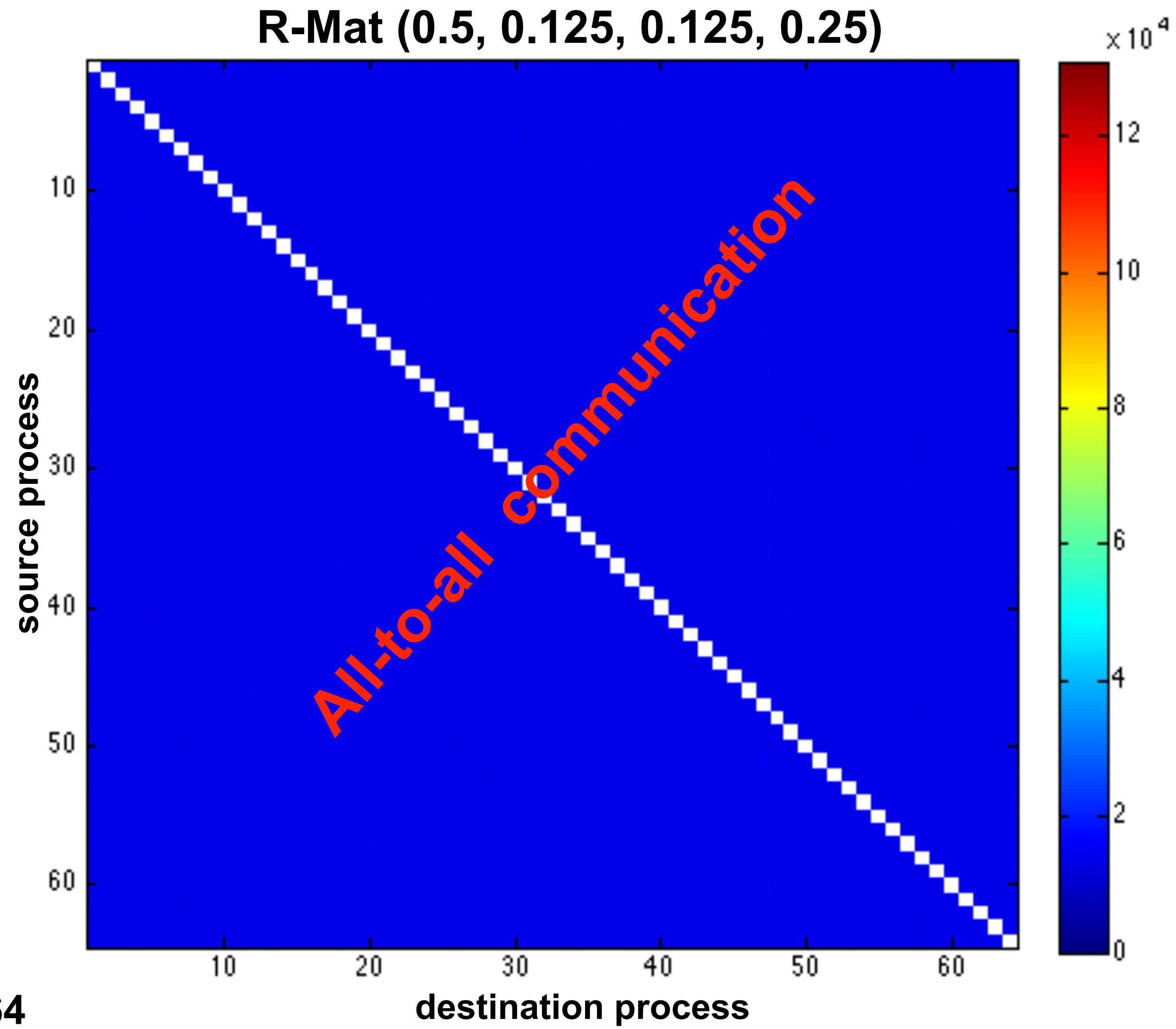
NNZ/process
min: 1.17E+06
max: 1.18E+06
avg: 1.18E+06
max/avg: 1.00

 # Messages (Phase 1)
total: 126
max: 2

Volume (Phase 1)
total: 2.58E+05
max: 4.10E+03

Nice properties:
Great load balance
Small number of messages
Low communication volume

Communication Pattern: 1D Random Partitioning



P=64

Number of Rows: 2^{23}
Nonzeros/Row: 8

NNZ/process
min: 1.05E+06
max: 1.07E+06
avg: 1.06E+06
max/avg: 1.01

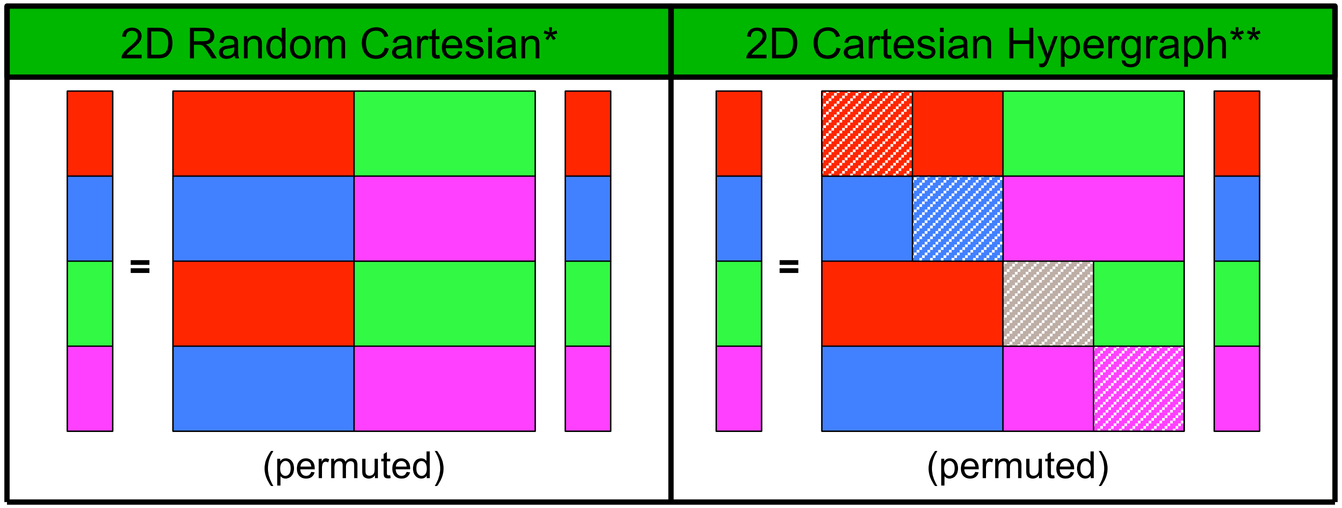
 # Messages (Phase 1)
total: 4032
max: 63

Volume (Phase 1)
total: 5.48E+07
max: 8.62E+05

Nice properties:
Great load balance

Challenges:
All-to-all communication

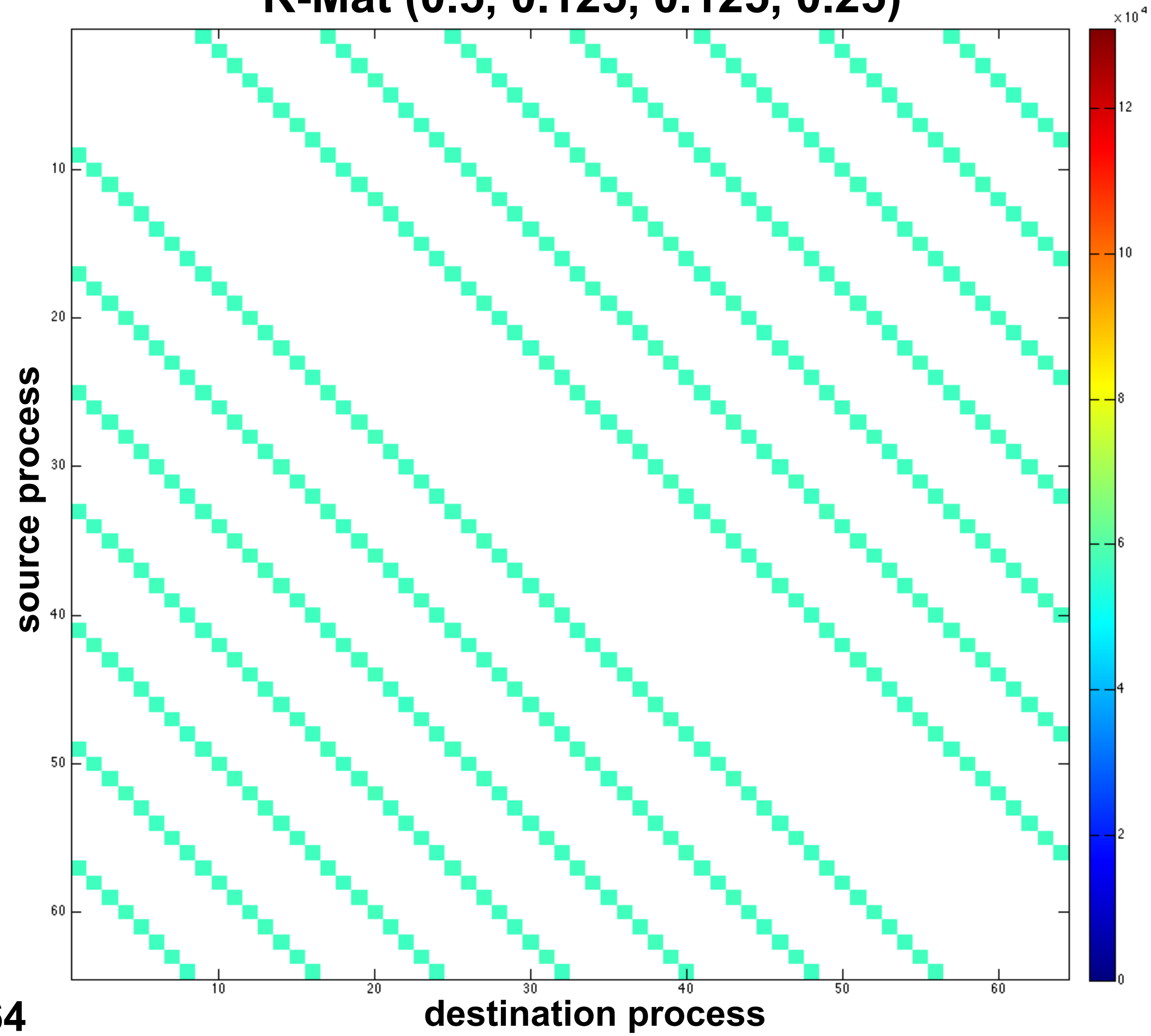
2D Partitioning



- 2D Partitioning
  - More flexibility: no particular part for entire row/column, more general sets of nonzeros
- Use flexibility of 2D partitioning to bound number of messages
- 2D Random Cartesian*
  - Block Cartesian with rows/columns randomly distributed
  - Cyclic striping to minimize number of messages
- 2D Cartesian Hypergraph**
  - Use hypergraph partitioning to minimize communication volume
  - Con: more costly to partition than random

*Hendrickson, et al.; Bisseling; Yoo, et al.
**Boman, Devine, Rajamanickam, "Scalable Matrix Computations on Large Scale-Free Graphs Using 2D Partitioning, SC2013.

Communication Pattern: 2D Random Partitioning Cartesian Blocks (2DR)



P=64

Number of Rows: 2^{23}
Nonzeros/Row: 8

NNZ/process
min: 1.04E+06
max: 1.05E+06
avg: 1.05E+06
max/avg: 1.01

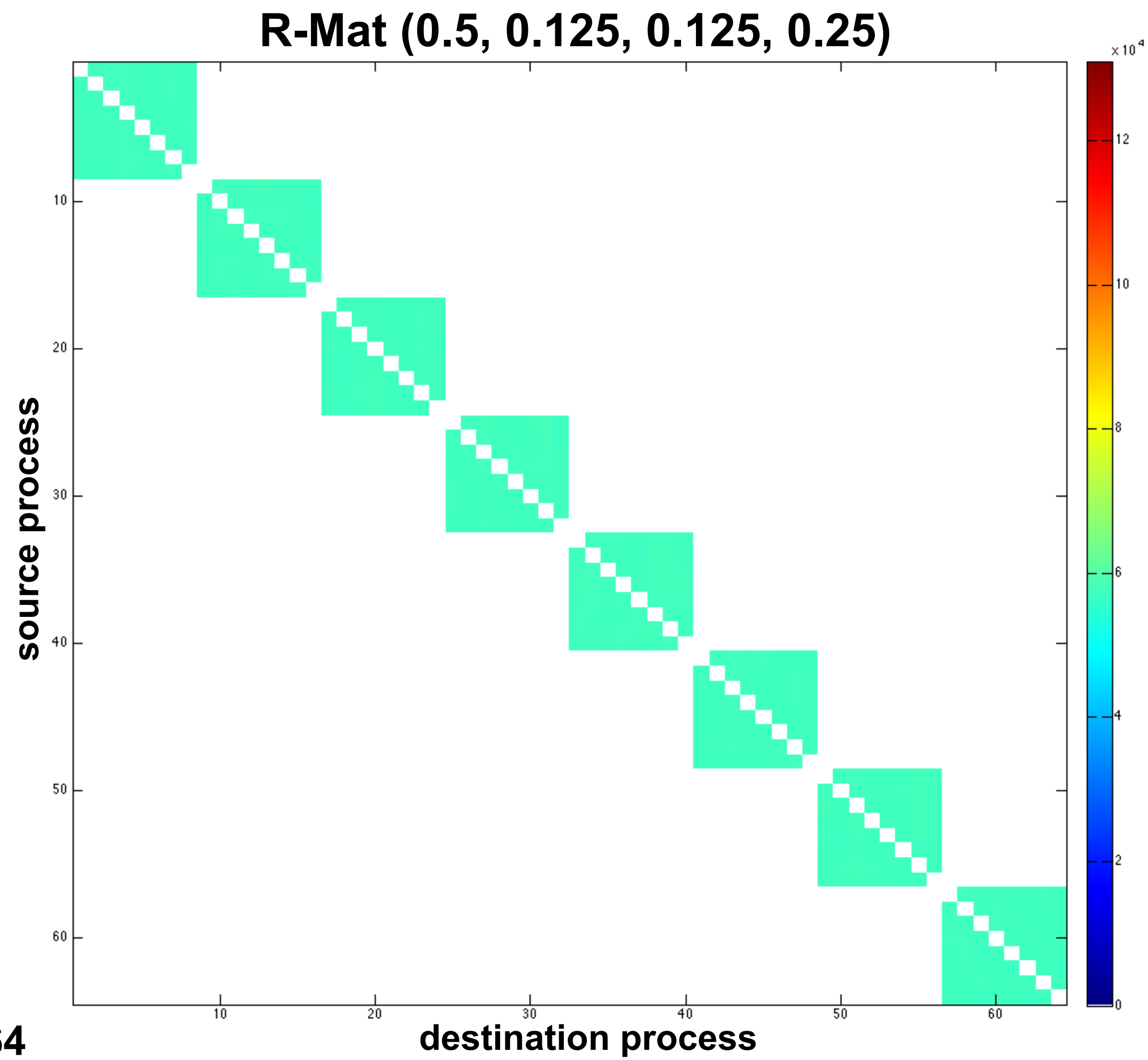
 # Messages (Phase 1)
total: 448
max: 7

Volume (Phase 1)
total: 2.57E+07
max: 4.03E+05

Nice properties:
No all-to-all communication
Total volume lower than 1DR

1DR = 1D Random

Communication Pattern: 2D Random Partitioning Cartesian Blocks (2DR)



Number of Rows: 2^{23}
Nonzeros/Row: 8

NNZ/process
min: 1.04E+06
max: 1.05E+06
avg: 1.05E+06
max/avg: 1.01

 # Messages (Phase 2)
total: 448
max: 7

Volume (Phase 2)
total: 2.57E+07
max: 4.03E+05

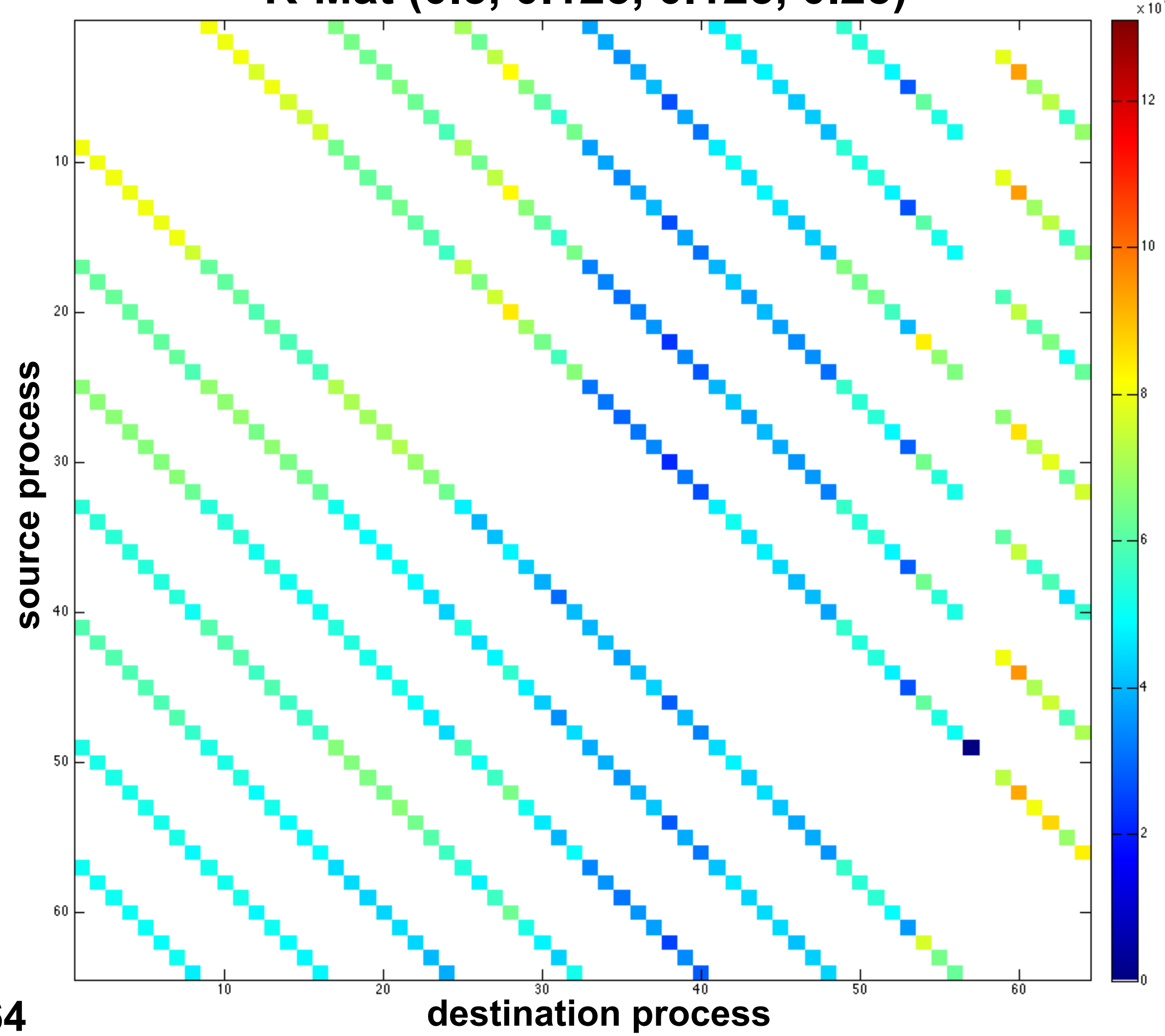
Nice properties:
No all-to-all communication
Total volume lower than 1DR

P=64

1DR = 1D Random

Communication Pattern: 2D Cartesian Hypergraph Partitioning

R-Mat (0.5, 0.125, 0.125, 0.25)



P=64

Number of Rows: 2^{23}
Nonzeros/Row: 8

NNZ/process
min: 5.88E+05
max: 1.29E+06
avg: 1.05E+06
max/avg: 1.23

 # Messages (Phase 1)
total: 448
max: 7

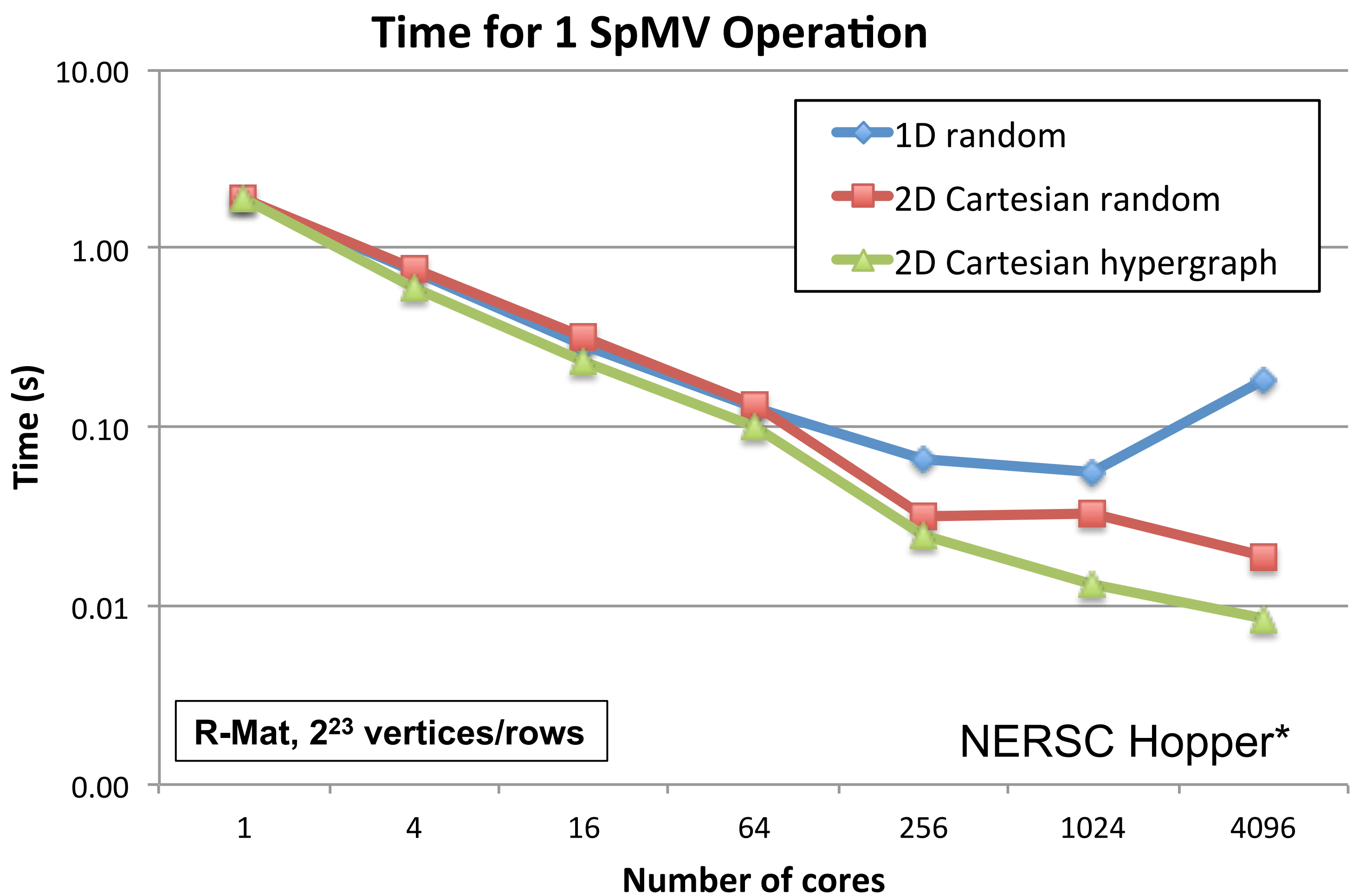
Volume (Phase 1)
total: 2.33E+07
max: 4.52E+05

Nice properties:
No all-to-all communication
Total volume lower than 2DR

Challenges:
Imbalance worse than 2DR

2DR = 2D Random Cartesian

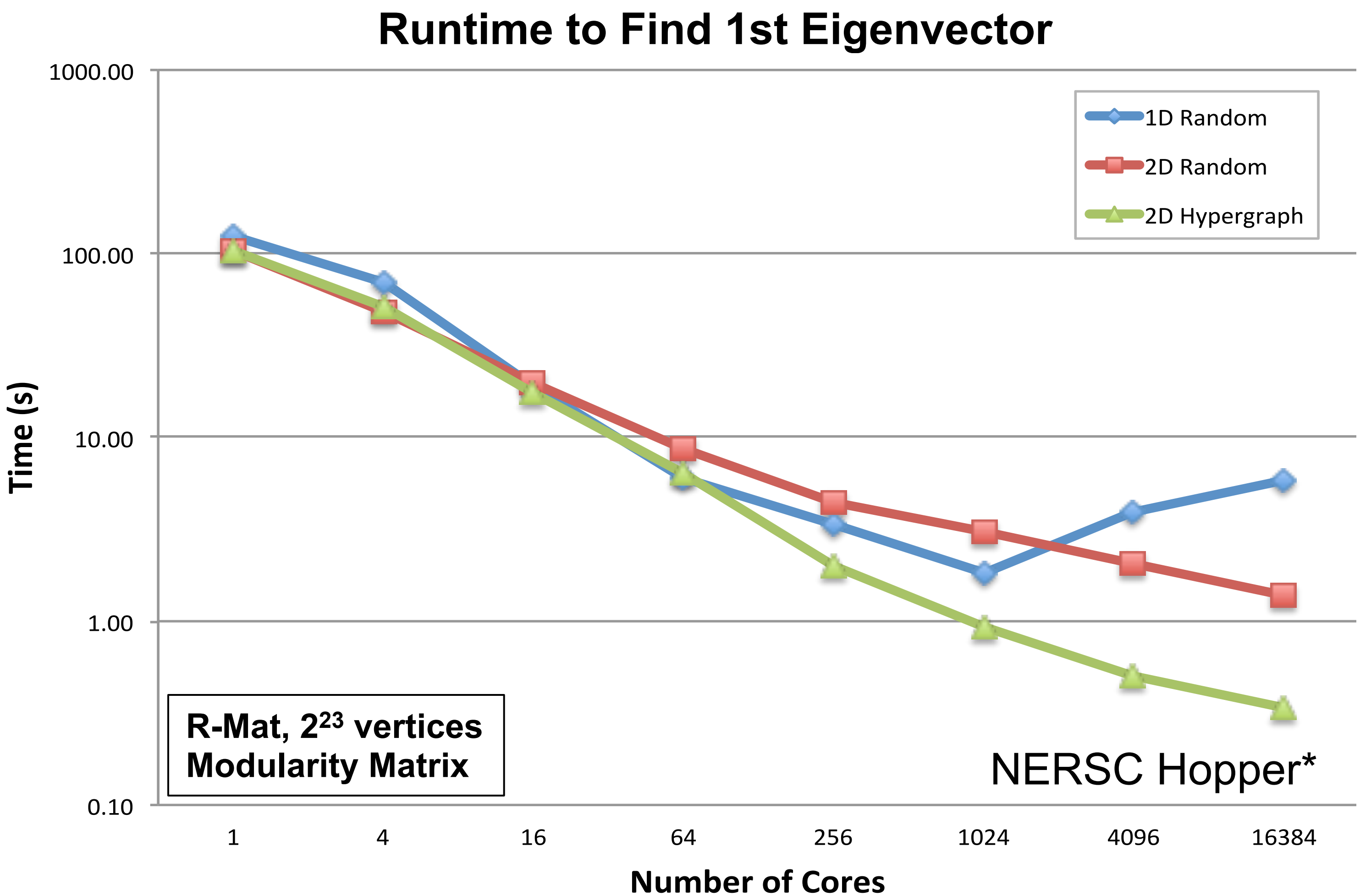
Improved Strong Scaling: SpMV



2D methods show improved scalability

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Improved Strong Scaling: Eigensolver



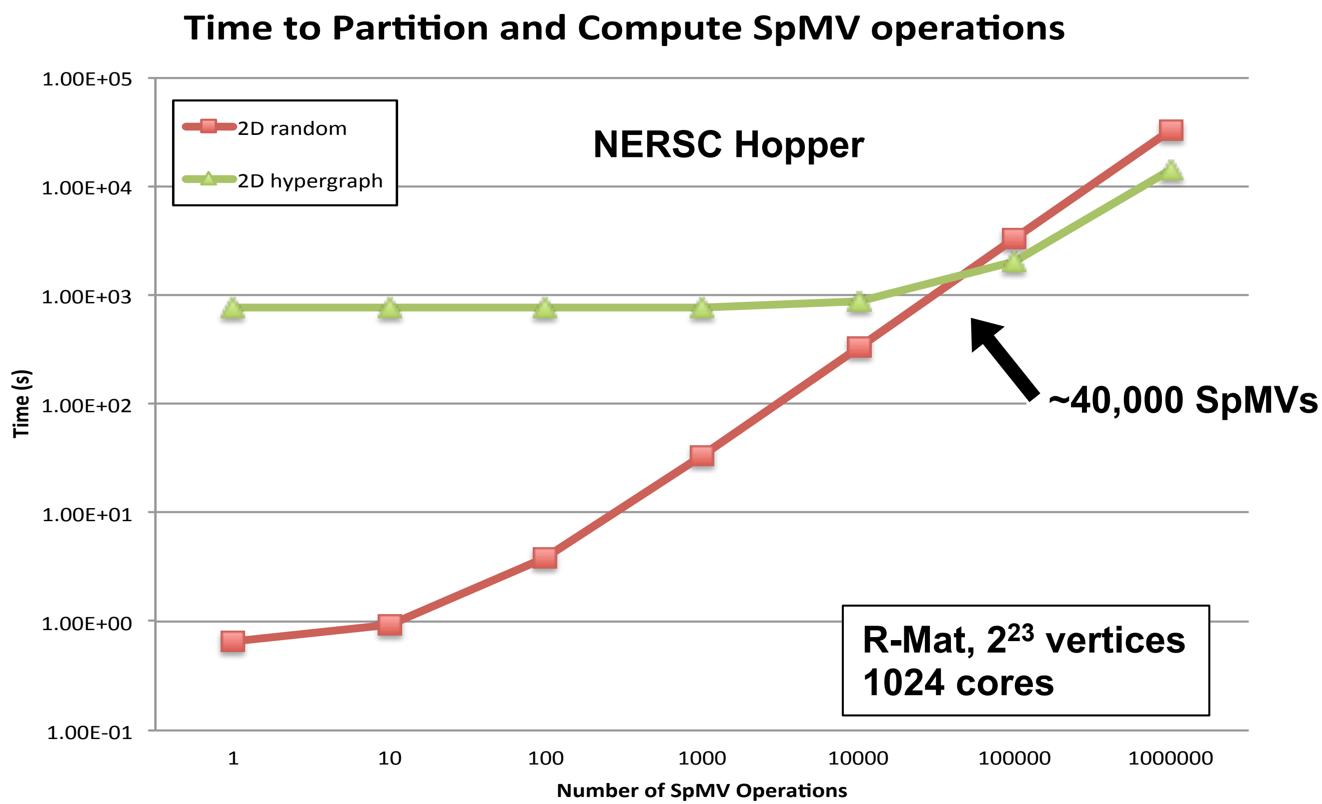
2D methods show improved scalability

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Outline

- Anomaly Detection in Very Large Graphs
- Eigenanalysis and Performance Challenges
- Improving Sparse Matrix-Vector Multiplication (SpMV) Performance through Data Partitioning
- Partitioning: Dynamic Graphs and Sampling
- Summary

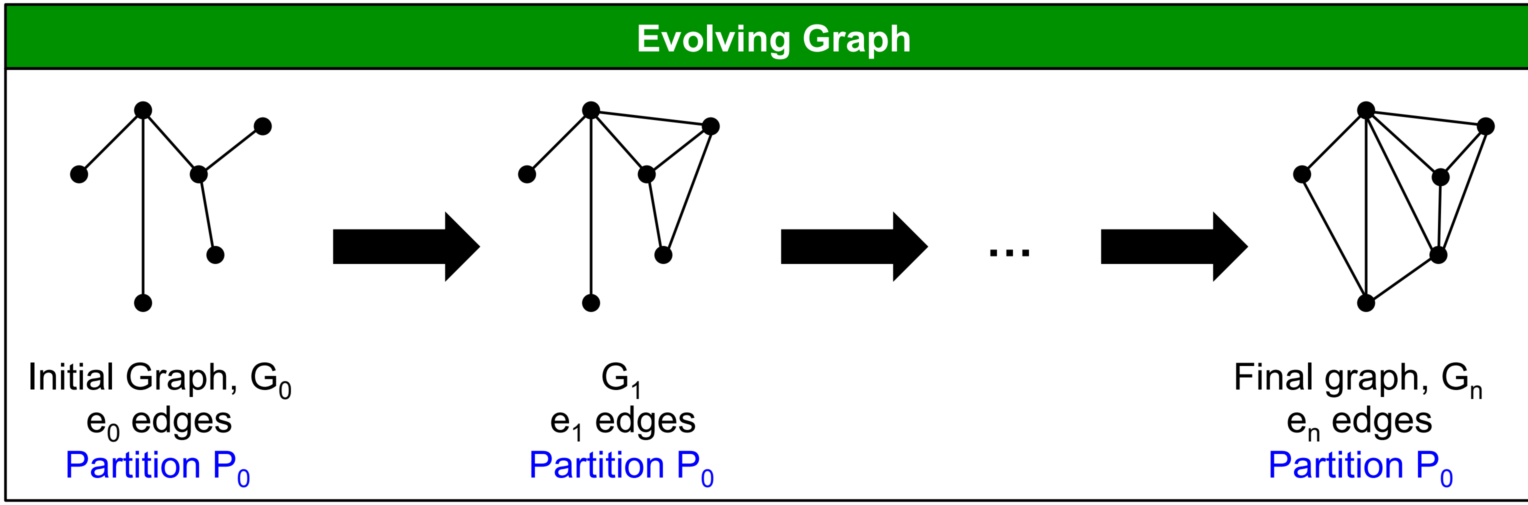
Challenge with Hypergraph Partitioning



- High partitioning cost of hypergraph methods must be amortized by computing many SpMV operations
- Detection* requires at most 1000s of SpMV operations
- Expensive partitions need to be effective for multiple graphs

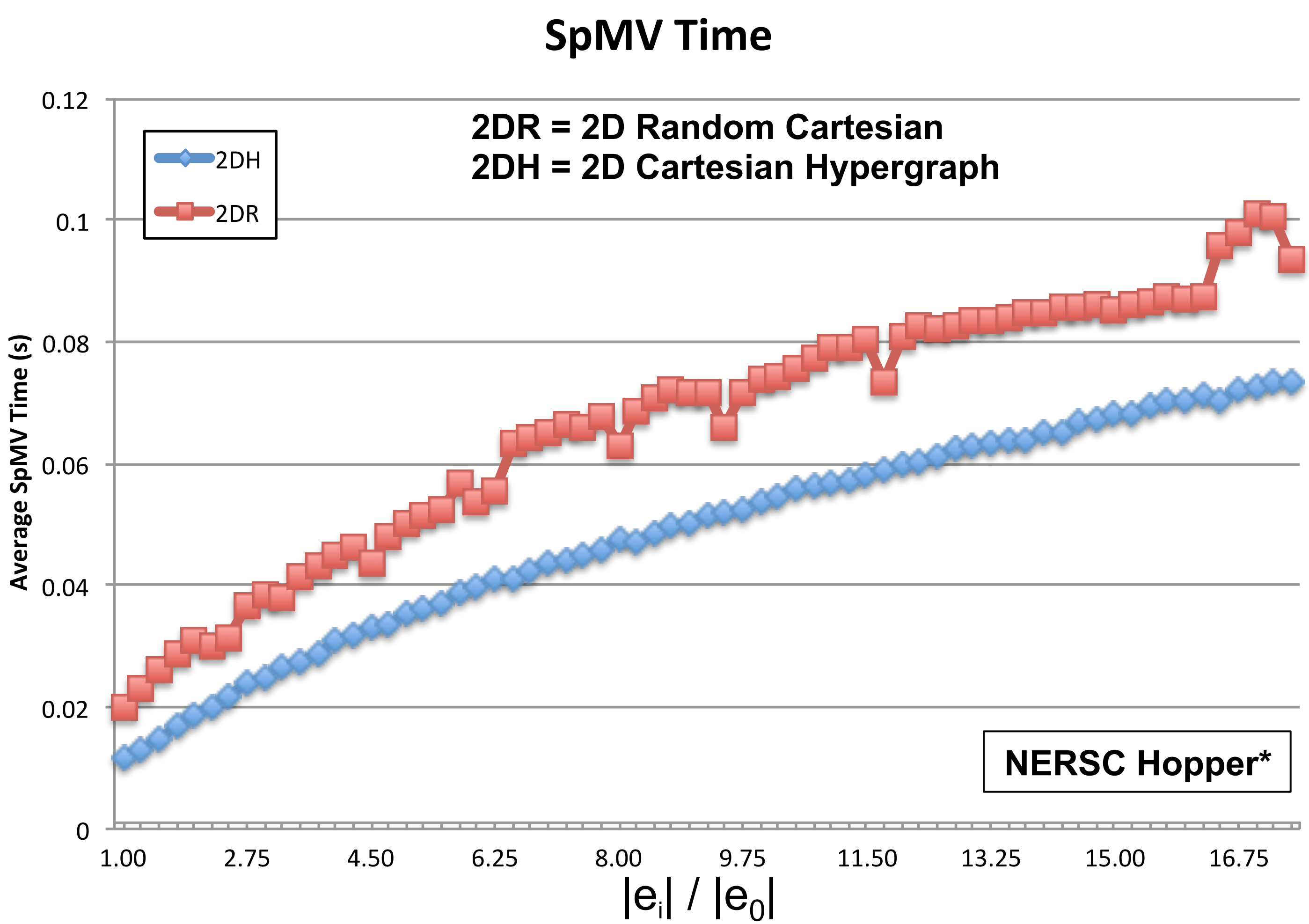
*L1 norm method: computing 100 eigenvectors

Experiment: Partitioning for Dynamic Graphs



- Key question: How long will a partition be effective?
- Initial experiment
  - Evolving R-Mat matrices: fixed number of rows, R-Mat parameters (a,b,c,d)
  - Start with a given number of nonzeros ($|e_0|$)
  - Iteratively add nonzeros until target number of nonzeros is reached ($|e_n|$)

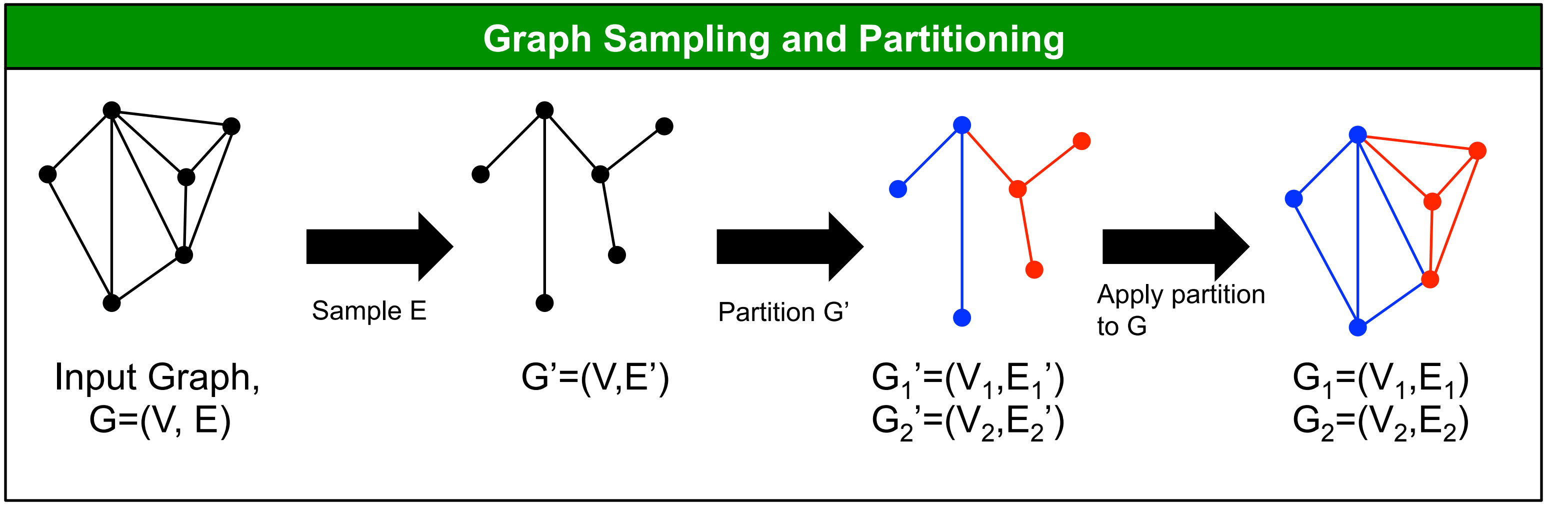
Results: Partitioning for Dynamic Graphs



Hypergraph partition surprising effective after more than 16x $|e_0|$ edges added

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

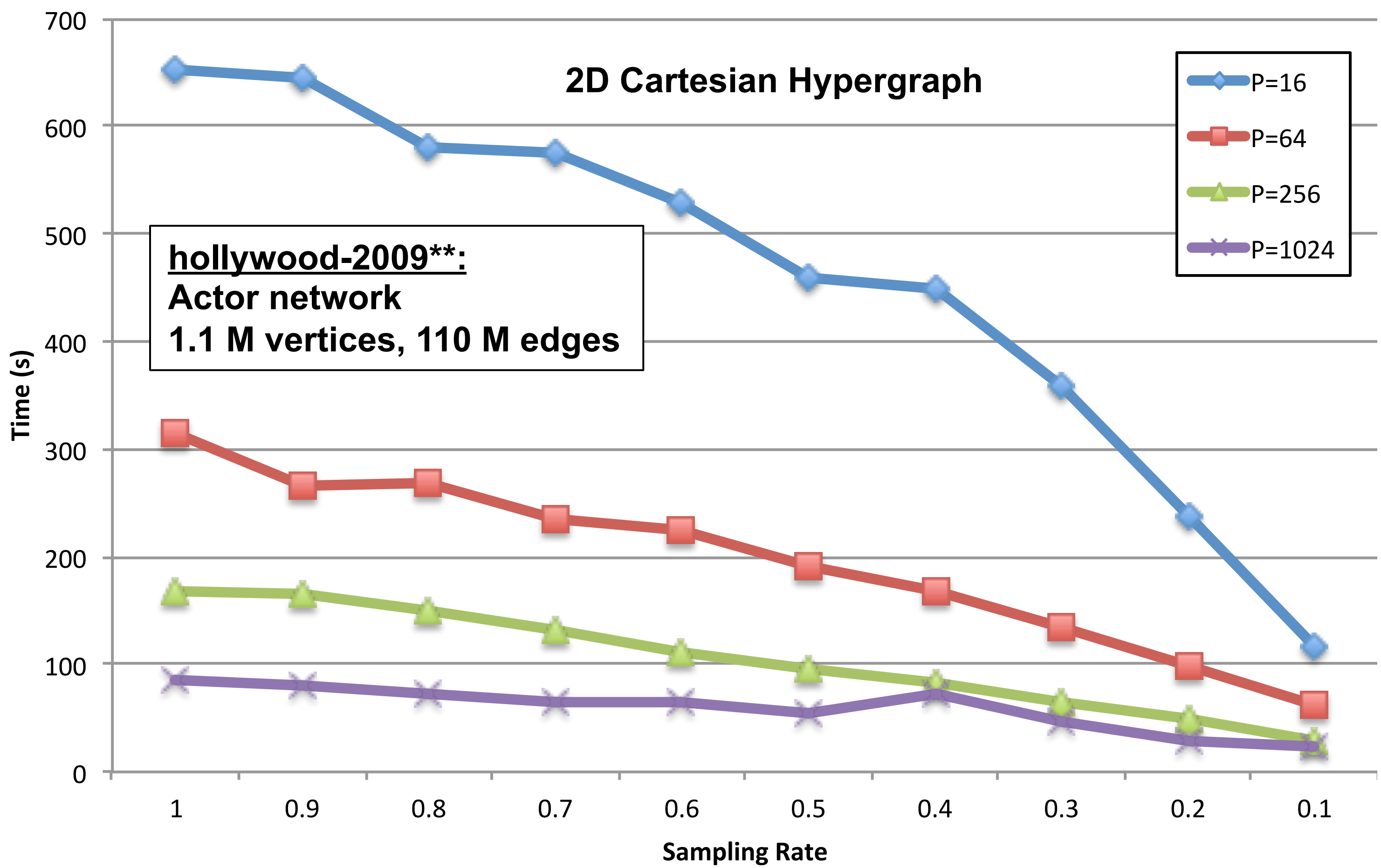
Sampling and Partitioning for Web/SN Graphs



- Sampling + Partitioning:
  1. Produce smaller graph G' by sampling edges in graph G (uniform random sampling), keep vertices same
  2. Partition G' (2D Cartesian Hypergraph)
  3. Apply partition to G

Idea: Partition sampled graph to reduce partitioning time

Partitioning + Sampling: Partitioning Time



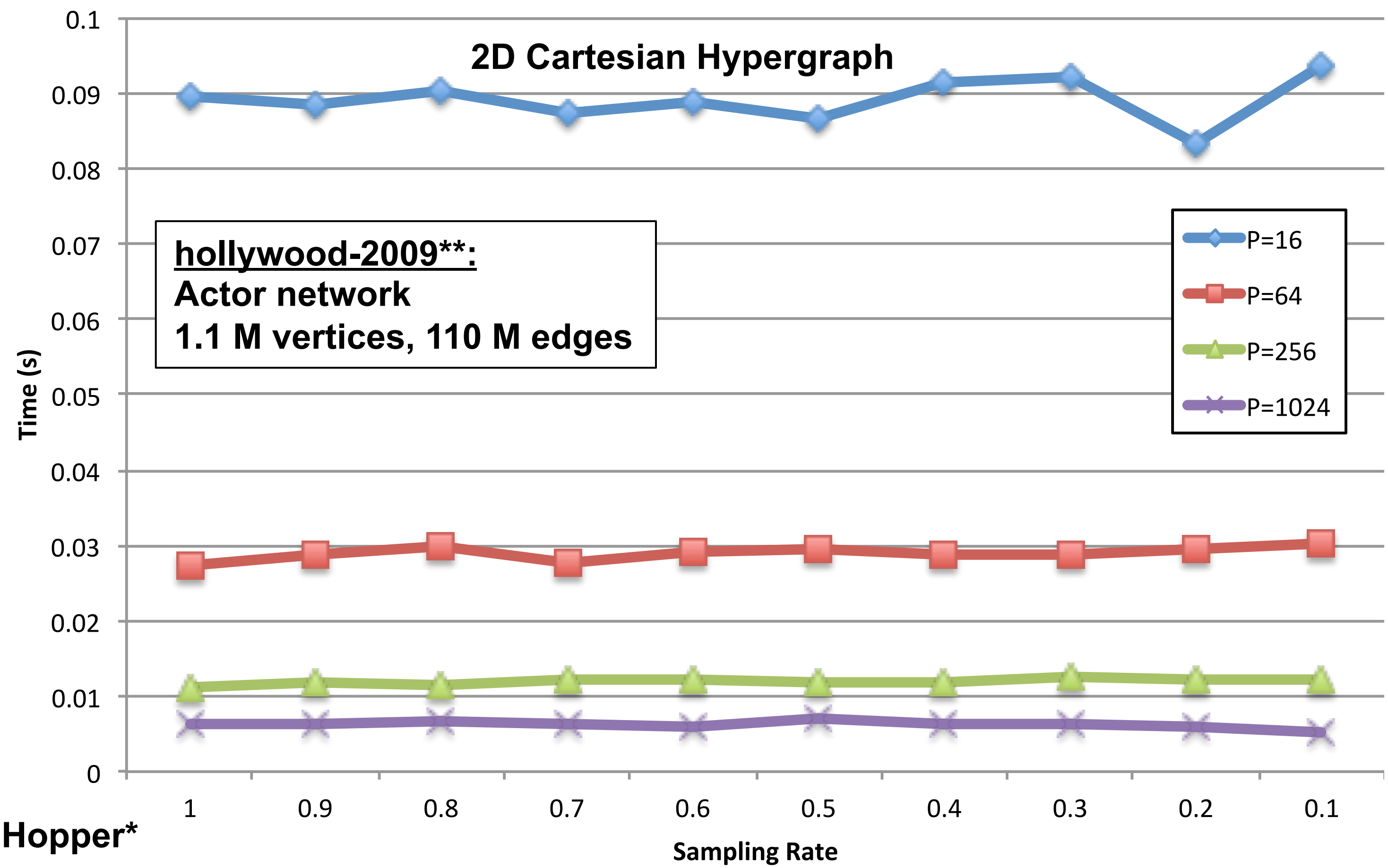
Edge sampling greatly reduces partitioning time (by up to 8x)

NERSC Hopper*

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

**The University of Florida Sparse Matrix Collection

Partitioning + Sampling: SpMV Time



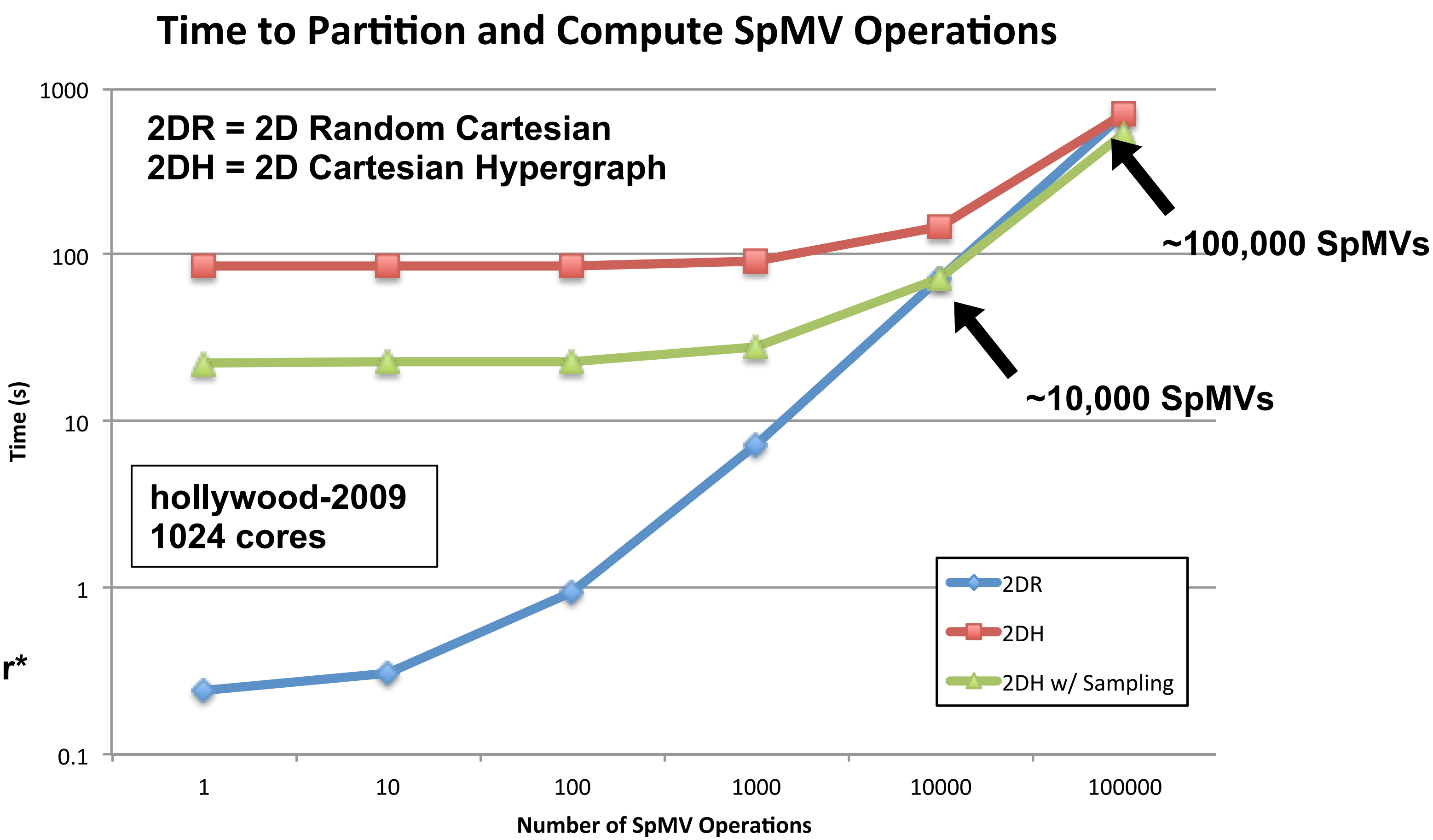
NERSC Hopper*

Resulting SpMV time does not increase for modest sampling

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

**The University of Florida Sparse Matrix Collection

Challenge with Hypergraph Partitioning Revisited



NERSC Hopper*

Sampling reduces overhead of hypergraph partitioning (fewer SpMVs needed to amortize partitioning cost)

* This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Summary

- Outlined HPC approach to detecting anomalies in big data
- Key component is eigensolver
- Solving resulting eigensystems challenging
  - Load imbalance
  - Poor data locality
- SpMV key computational kernel
  - 1D data partitioning limits performance due to all-to-all communication
  - 2D data partitioning can be used to improve scalability
- 2D hypergraph partitioning promising but expensive
- Sampling can improve 2D hypergraph partitioning performance for web/SN graphs