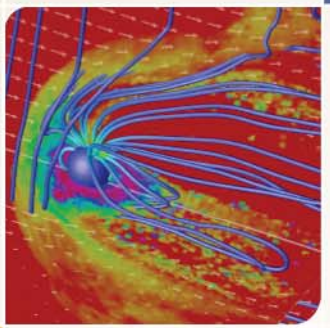
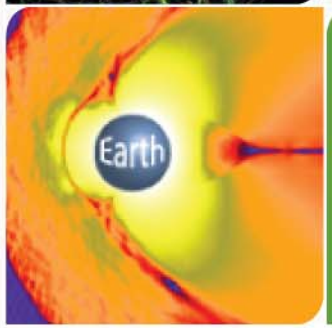
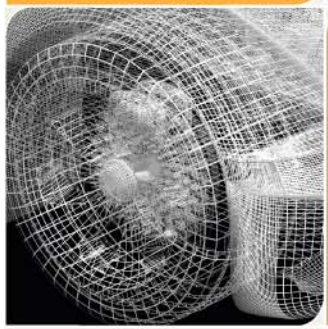





Exascale Analytics of Massive Social Networks

David A. Bader

Georgia Tech College of Computing
Computational Science and Engineering

SIAM AN09 Minisymposium on HPC on Massive Real-World Graphs



- Session 1: Wednesday, July 8, 10:30 AM - 12:30 PM
  - **Exascale Analytics of Massive Social Networks** *David A. Bader*, Georgia Institute of Technology
  - **Graph Detection Theory for Power Law Graphs** *Jeremy Kepner*, Massachusetts Institute of Technology
  - **Parallel Combinatorial BLAS and Applications in Graph Computation** *Aydin Buluc* and John R. Gilbert, University of California, Santa Barbara
  - **Scaling up Graph Algorithms on Emerging Multicore Systems** *Kamesh Madduri*, Lawrence Berkeley National Laboratory
- Session 2: Wednesday, July 8, 4:00 PM - 6:00 PM
  - **Parallel Implementation of Tensor Decompositions for Large Data Analysis** *Mark P. Sears*, Brett W. Bader, and Tammy Kolda, Sandia National Laboratories
  - **Novel Graph Algorithms for Structure Extraction from Informatics Networks** *Michael Mahoney*, Stanford University
  - **Scalable & Efficient Parallelization of Graph Methods for Boolean Satisfiability and Power Grid Analysis on the Cray XMT** *Daniel Chavarria*, Pacific Northwest National Laboratory
  - **Open Discussion / Challenge Problems**
- Session 3: Thursday, July 9 ,10:30 AM - 12:30 PM
  - **Detecting Community Structure in Dynamic Networks** *Sanjukta Bhowmick*, Shweta Bansal, Kelly Fermoyle, and Padma Raghavan, Pennsylvania State University
  - **Structure of Large Scale Social Contact Graphs and its Effect on Epidemics** *Anil Vullikanti*, Virginia Polytechnic Institute & State University
  - **High Performance Computing for Large Graph Problems** *Bruce Hendrickson* and Jonathan Berry, Sandia National Laboratories
  - **Anatomy of a Distributed Graph** *Andrew Lumsdaine*, Douglas Gregor, and Nick Edmonds, Indiana University

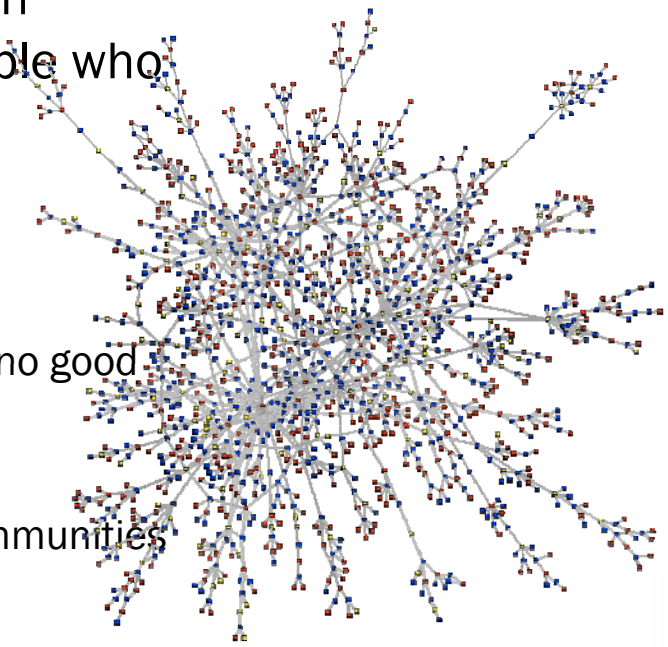
Workshop Co-Chairs: **David A. Bader**, *Georgia Institute of Technology;* **Jeremy Kepner**, *Massachusetts Institute of Technology Lincoln Laboratory;* **John R. Gilbert**, *University of California, Santa Barbara;* **Sanjukta Bhowmick**, *Pennsylvania State University;* **Padma Raghavan**, *Pennsylvania State University*

www.graphanalysis.org ←- We will post talks here

Social Networks: Spatio-Temporal Internation Networks and Graphs (STING)

- Facebook has more than 200 million active users

- Example application: Malcolm Gladwell, in *The Tipping Point*, identifies three personality types that play central roles in epidemic/viral spread: Connectors, Mavens, and Salespeople. We can identify, for example, Connectors who are people who bridge between social communities.



- Traditional graph partitioning often fails:
  - **Topology**: Interaction graph is low-diameter, and has no good separators
  - **Irregularity**: Communities are not uniform in size
  - **Overlap**: individuals are members of one or more communities



Open Questions: Algorithmic Kernels for Spatio-Temporal Interaction Graphs and Networks (STING)

- Traditional graph theory:
  - Graph traversal (e.g. breadth-first search)
  - S-T connectivity
  - Single-source shortest paths
  - All-pairs shortest paths
  - Spanning Tree
  - Connected Components
  - Biconnected Components
  - Subgraph isomorphism (pattern matching)
  - ....
  - Others?

Graph Analytics for Social Networks

- Are there new graph techniques? Do they parallelize? Can the computational systems (algorithms, machines) handle massive networks with millions to billions of individuals? Can the techniques tolerate noisy data, massive data, streaming data, etc. …
- Communities may overlap, exhibit different properties and sizes, and be driven by different models
  - Detect communities (static or emerging)
  - Identify important individuals
  - Detect anomalous behavior
  - Given a community, find a representative member of the community
  - Given a set of individuals, find the best community that includes them

Suddenly, the flock became suspicious:
How come the newcomer wasn't shorn?



Centrality in Massive Social Network Analysis

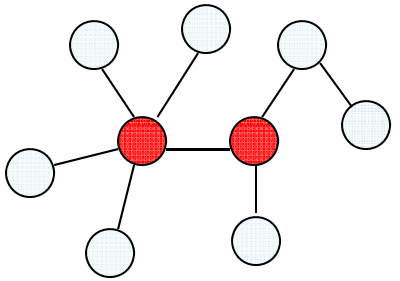
- Centrality metrics: Quantitative measures to capture the importance of person in a social network
  - Betweenness is a global index related to shortest paths that traverse through the person
  - Can be used for community detection as well

- Identifying *central* nodes in large complex networks is the key metric in a number of applications:
  - Biological networks, protein-protein interactions
  - Sexual networks and AIDS
  - Identifying key actors in terrorist networks
  - Organizational behavior
  - Supply chain management
  - Transportation networks
- Current Social Network Analysis (SNA) packages handle 1,000's of entities, our techniques handle BILLIONS (6+ orders of magnitude larger data sets)

Betweenness Centrality (BC)

- Key metric in social network analysis
  [Freeman '77, Goh '02, Newman '03, Brandes '03]

$$BC(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$



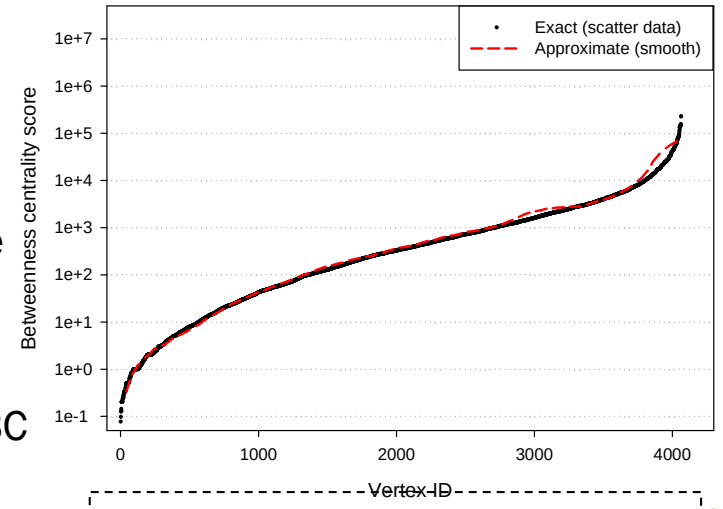
- σ_{st} : Number of shortest paths between vertices s and *t*
- $\sigma_{st}(v)$: Number of shortest paths between vertices s and *t* passing through *v*

- Exact BC is compute-intensive

BC Algorithms

- Brandes [2003] proposed a faster sequential algorithm for BC on sparse graphs
  - $O(mn + n^2 \log n)$ time and $O(n)$ space for weighted graphs
  - $O(mn)$ time for unweighted graphs
- We designed and implemented the first parallel algorithm:
  - [Bader, Madduri; ICPP 2006]

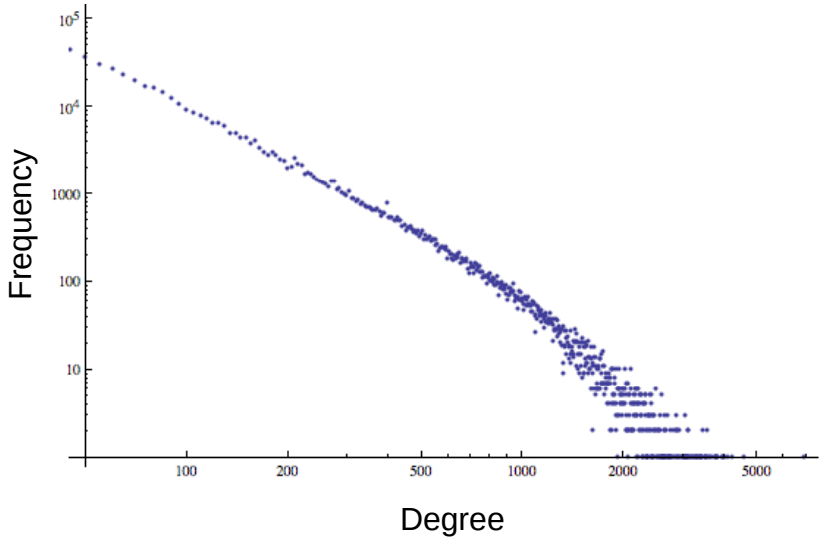
- Approximating Betweenness Centrality [Bader Kintali Madduri Mihail 2007]
  - Novel approximation algorithm for determining the betweenness of a *specific vertex or edge* in a graph
  - *Adaptive* in the number of samples
  - Empirical result: At least 20X speedup over exact BC

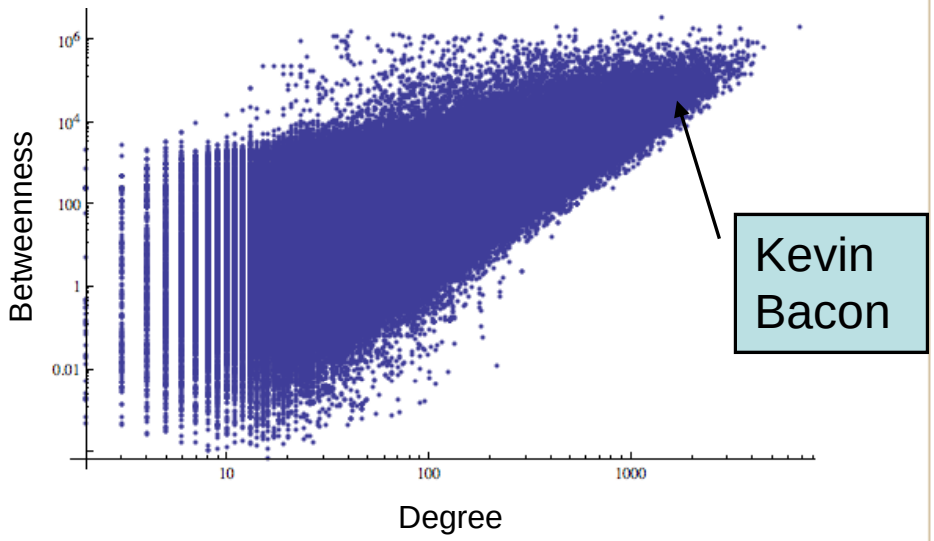


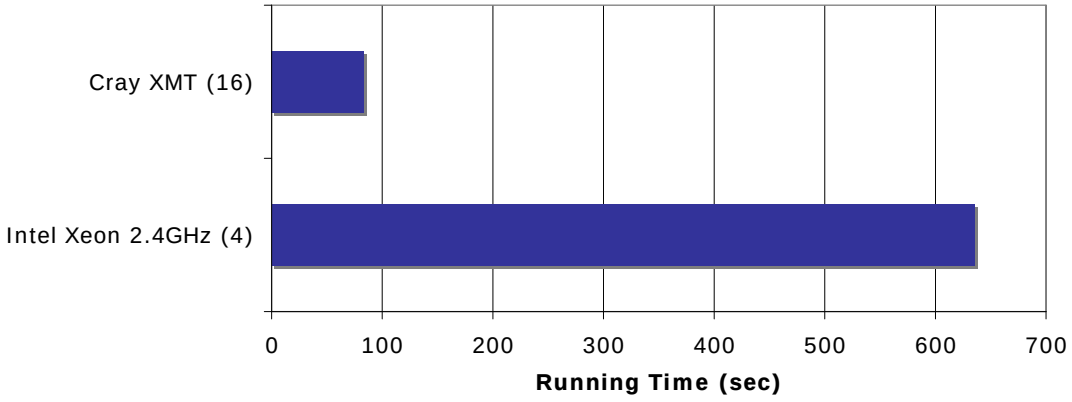
Graph: 4K vertices and 32K edges,
System: Sun Fire T2000 (Niagara 1)

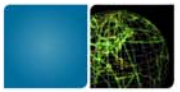
IMDB Movie Actor Network (Approx BC)

An undirected graph of 1.54 million vertices (movie actors) and 78 million edges. An edge corresponds to a link between two actors, if they have acted together in a movie.








HPC Challenges for Massive SNA

- Algorithms that work on complex networks with hundreds to thousands of vertices often disintegrate on real networks with millions (or more) of vertices
  - For example, betweenness centrality is not robust to noisy data (biased sampling of the actual network, missing friendship edges, etc.)
  - Requires niche computing systems that can offer irregular and random access to large global address spaces.

k-Betweenness Centrality, BC_k



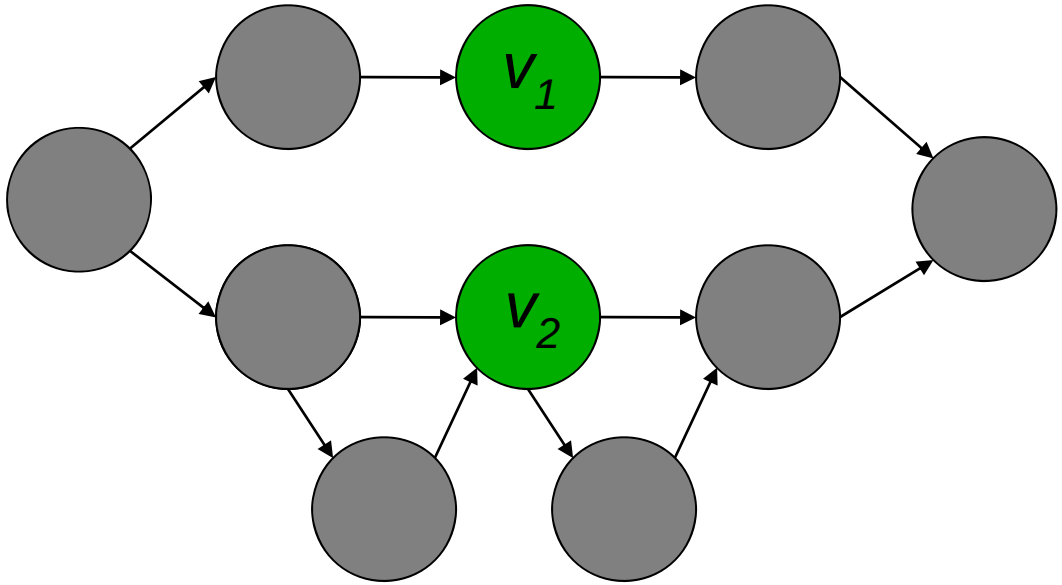
- A new twist on betweenness centrality:
  - Count **short** paths in addition to **shortest** paths
  - Captures wider connectivity information
- Applying BC_k to a real data set:
  - How do the BC indices behave with increasing *k*?
  - Which vertices have zero scores?
    - (Directed and undirected graphs are different.)
  - Can we approximating by BC_k random sampling?
- Scalability on the Cray XMT with RMAT graphs (generated by sampling from a Kronecker product).



k-Betweenness Centrality

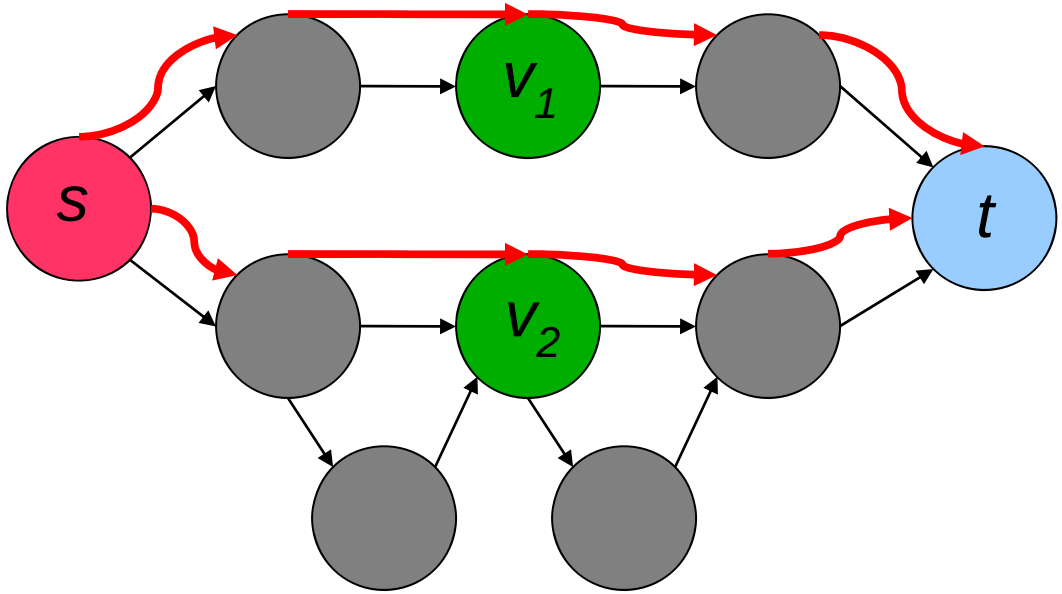
- Measure *centrality* of a vertex ***v*** by the number of paths passing through ***v*** between ***s*** and ***t*** relative to the number of paths connecting ***s*** and ***t***.
- High *betweenness centrality (BC):* many **shortest** paths
- High *k-betweenness centrality* (BC_k)*:* many **short** paths
  - All paths no longer than the shortest + parameter *k* counted.
  - 0-Betweenness centrality is simply betweenness centrality.
  - 1-BC also counts paths one step longer than the shortest.
- BC_k captures more connectivity information with *k*.
- Expensive to compute as *k* grows, but approximated…

Betweenness Centrality



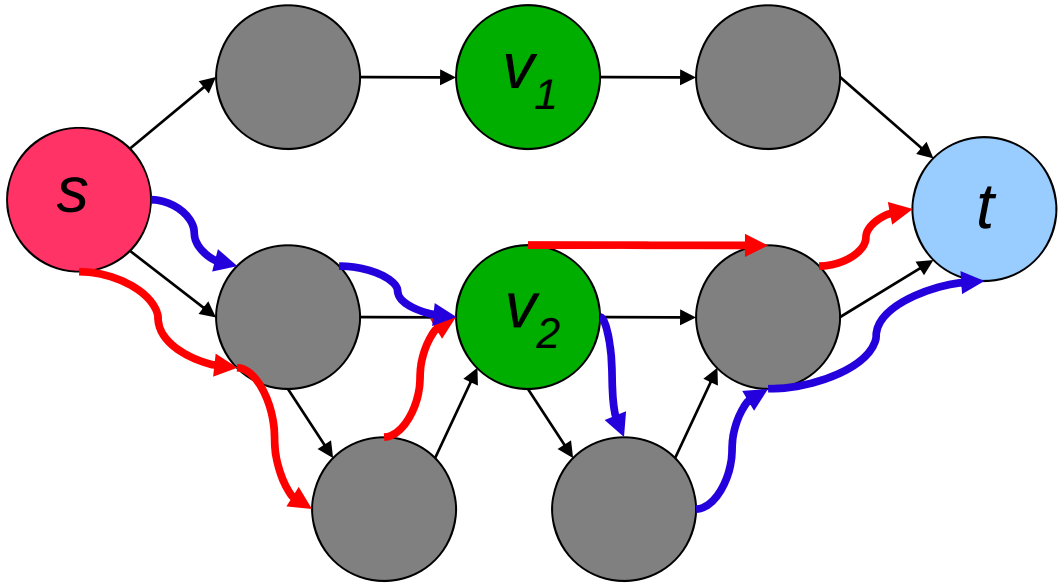
- How important are v_1 and v_2? Use betweenness centrality.
- The betweenness centrality of v_1, $BC(v_1)$:
  - Consider **shortest** paths between any two vertices s, $t \neq v_1$.
  - Sum over all such s, t: fraction of paths passing through v_1

BC: Need More Than the Shortest Path?



- Consider the view from a particular vertex pair s, *t*.
- Total of five paths, so the *st* contributions to v_1, v_2 = 1/5.
- But there is more redundancy through v_2, more nodes influence / are influenced by v_2…

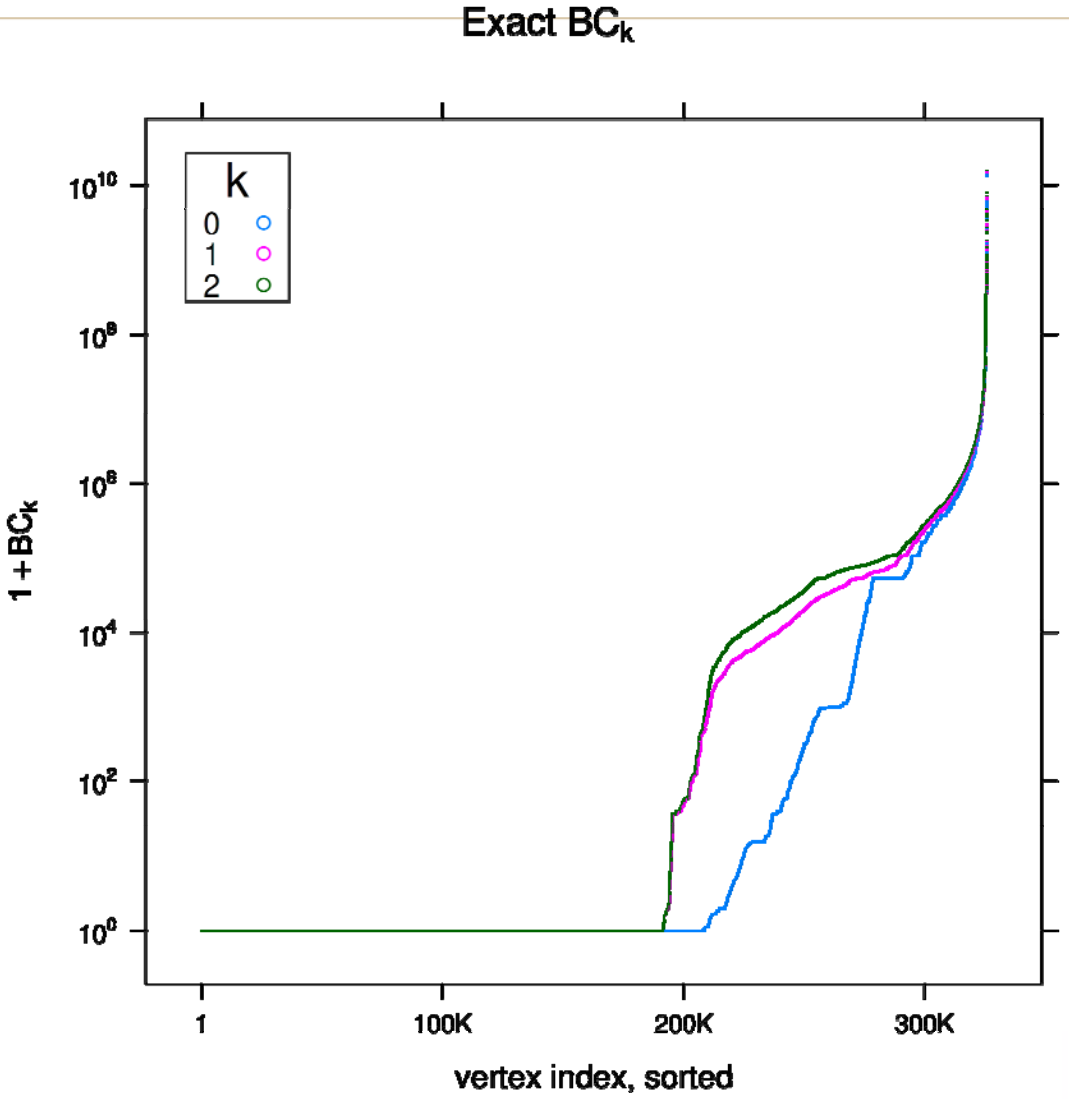
k-Betweenness Centrality: Shortest + *k*



- Consider counting paths **one longer** than the shortest.
- Nothing new through v_1. Two new paths cross through v_2!
- *k*-Betweenness Centrality (BC_k):
  - Consider paths within *k* of the shortest path. Above is BC_1.
  - 0-Betweenneess centrality is regular BC, $BC_0(v) = BC(v)$.

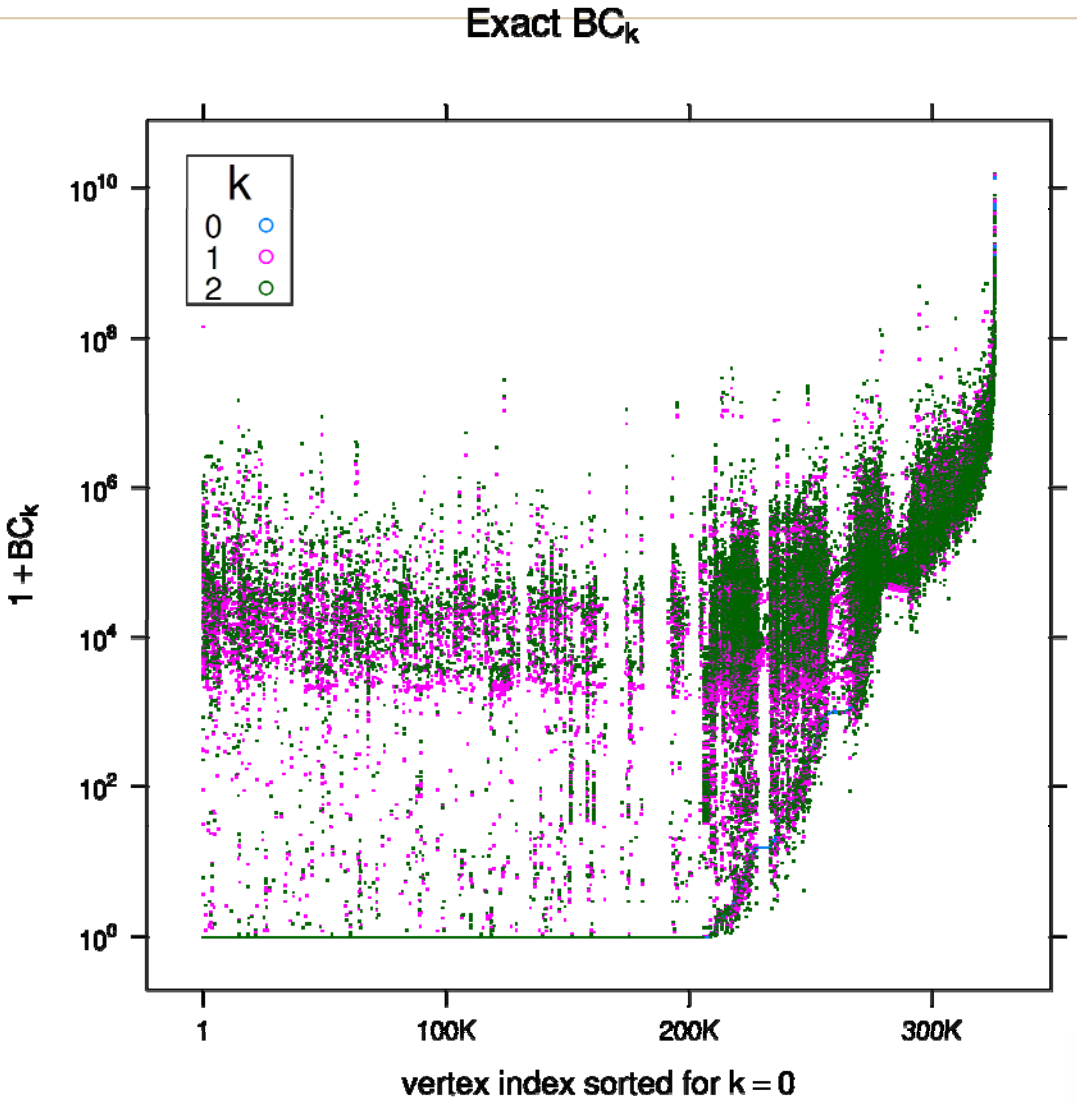
BC_k for $k > 0$: More Path Information

- Exact BC_k for $k = 0, 1, 2$
- On directed web graph
- Vertices in increasing BC_k order (independently)
- Large difference going from $k = 0$ to $k > 0$
- Few additional paths found in $k = 2$
- **$k > 0$ captures more path information, somewhat converges**



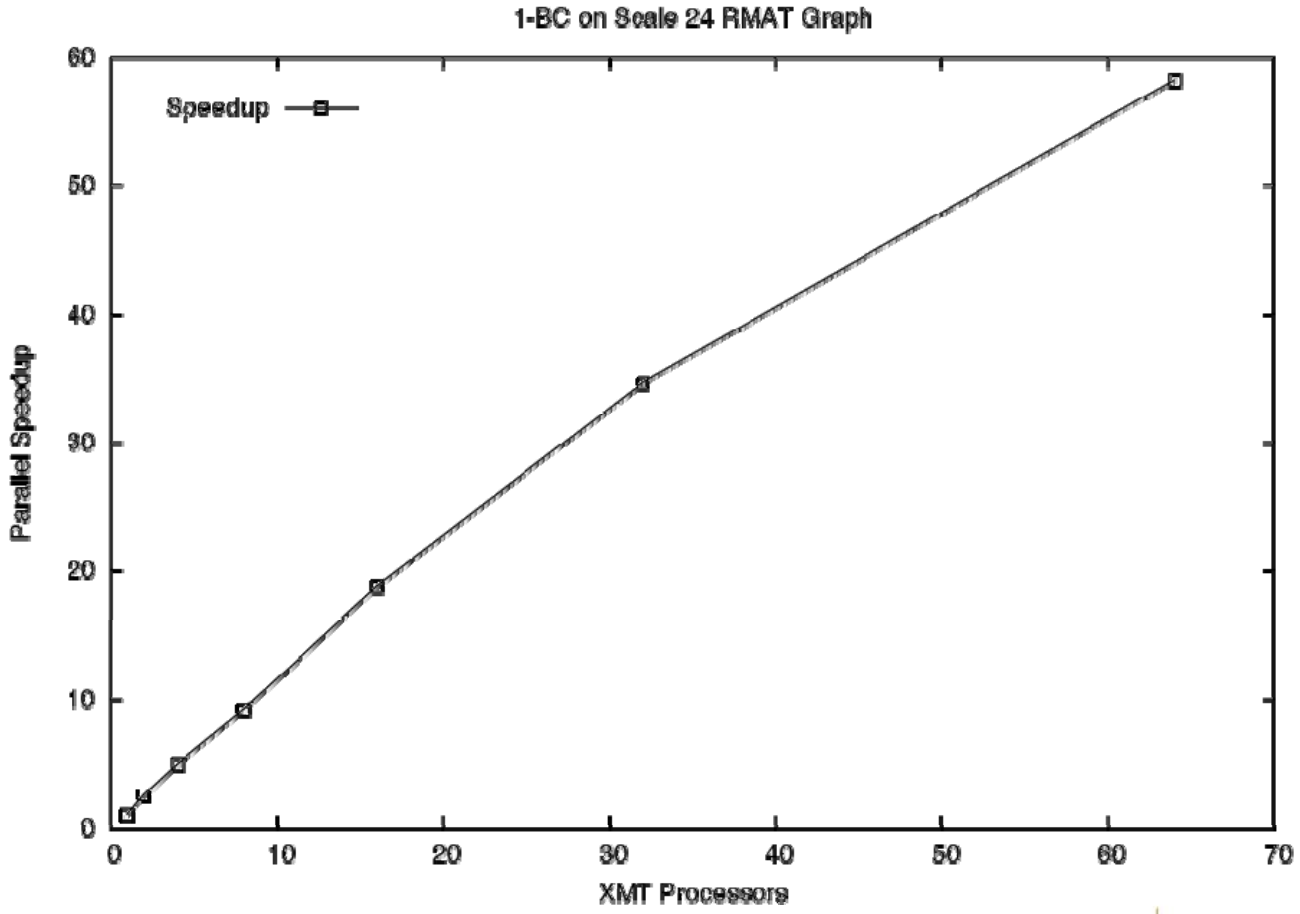
BC_k for $k > 0$: More Path Information

- Exact BC_k for $k = 0, 1, 2$
- On directed web graph
- Vertices in increasing BC_k order (by $k = 0$)
- Large difference going from $k = 0$ to $k > 0$
- Few additional paths found in $k = 2$
- Note how many vertices jump from $BC_0 = 0$ to $BC_k > 0$!



Scalability of k-Betweenness Centrality

- 52x speedup for k=1 on a 64p Cray XMT





Open Questions

- Improve models for representing massive, dynamic social networks
- Can we design community structure detection algorithms that overcome the serious limitations of the current literature (performance, scale, accuracy, resolution limits, etc.)
- Explore community detection on real data, rather than toy networks. Real data is often noisy, massive, streaming, biased, …



Collaborators and Acknowledgments

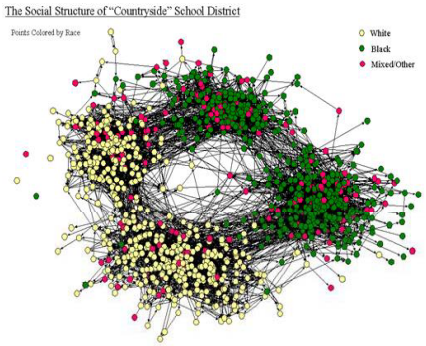
- David Ediger (Georgia Tech)
- Karl Jiang (Georgia Tech)
- Jason Riedy (UC Berkeley & Georgia Tech)
- Kamesh Madduri (Lawrence Berkeley National Lab)
- John Feo and Daniel G. Chavarría-Miranda (Pacific Northwest Lab)
- Jon Berry and Bruce Hendrickson (Sandia National Laboratories)
- Guojing Cong (IBM TJ Watson Research Center)
- Jeremy Kepner (MIT Lincoln Laboratory)

Center for Adaptive Supercomputing Software (CASS-MT)

- CASS-MT, launched July 2008
- Pacific-Northwest Lab
  - Georgia Tech, Sandia, WA State, Delaware

Pacific Northwest NATIONAL LABORATORY

- The newest breed of supercomputers have hardware set up not just for speed, but also to better tackle large networks of seemingly random data. And now, a multi-institutional group of researchers has been awarded $4.0 million to develop software for these supercomputers. Applications include anywhere complex webs of information can be found: from internet security and power grid stability to complex biological networks.



CRAY

Georgia Tech College of Computing

Acknowledgment of Support