SAND2012-0068C

# Scalable Methods for Characterizing and Generating Large Graphs

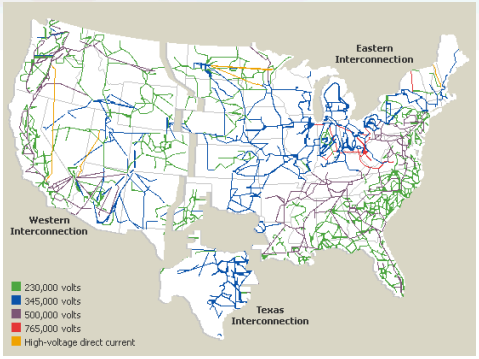
Ali Pinar, C. Seshadri and Tamara G. Kolda

Sandia National Labs

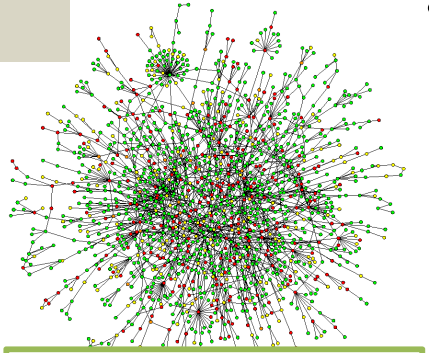

U.S. Department of Energy
Office of Advanced Scientific Computing Research

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.

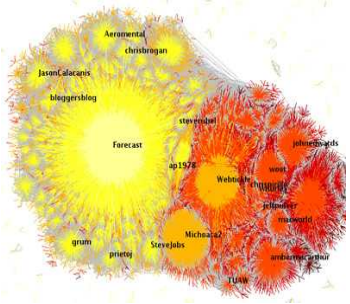
# Networks are everywhere




U.S. Power Grid [GENI]



Yeast protein interactions [Bordalier institute]



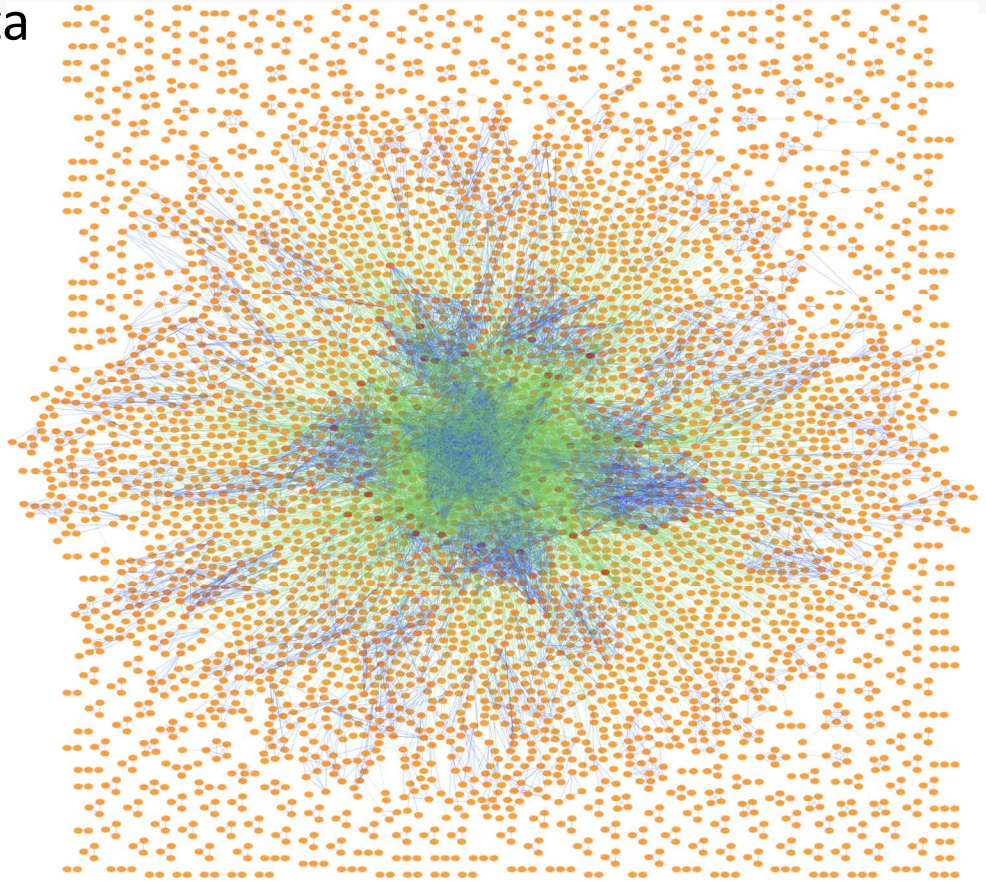

Twitter social network[Akshay Java, 2007]

- Physical networks
  - Defined by physical connections
  - Clearly defined for each system
  - Applications: Power, water, communication networks
- Functional networks
  - Defined by well-defined functional dependencies between entities
  - Complete information is available; Needs abstraction
  - Applications: Supply chains, chemical reaction networks
- Interaction networks
  - Defined by interactions between entities
  - Information is incomplete and noisy; Needs abstraction
  - Applications: Cybersecurity, intelligence, epidemics

*Network science is built on the common ground among these wide variety of applications.*

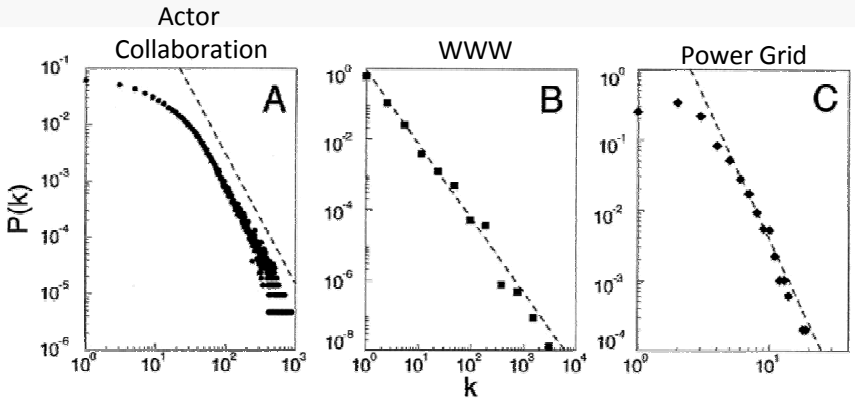
# Why Model Massive Graphs?

- Enable sharing of surrogate data
  - Computer network traffic
  - Social networks
  - Financial transactions
- Insight into…
  - Generative process
  - Community structure
  - Evolution
  - Uncertainty
- Testing graph algorithms
  - Scalability
  - Versatility (e.g., vary degree distributions)
  - Verification & validation

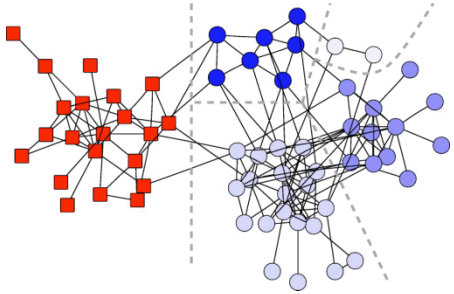


Block Two-Level Erd}os-Renyi (BTER) graph; image courtesy of Nurcan Durak.

# Model Desiderata




A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. Science, 286(5349):509-512, 1999.



M.E.J. Newman and M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69, 026113, 2004.


http://www.graph500.org/

- Capture heavy-tailed degree distribution
  - Not necessarily exactly power law
- Capture community structure
  - Measured indirectly through clustering coefficient, *k*-cores, and other measures
- Able to "fit" real-world data
  - Reproduce degree distribution
  - Reproduce community structure

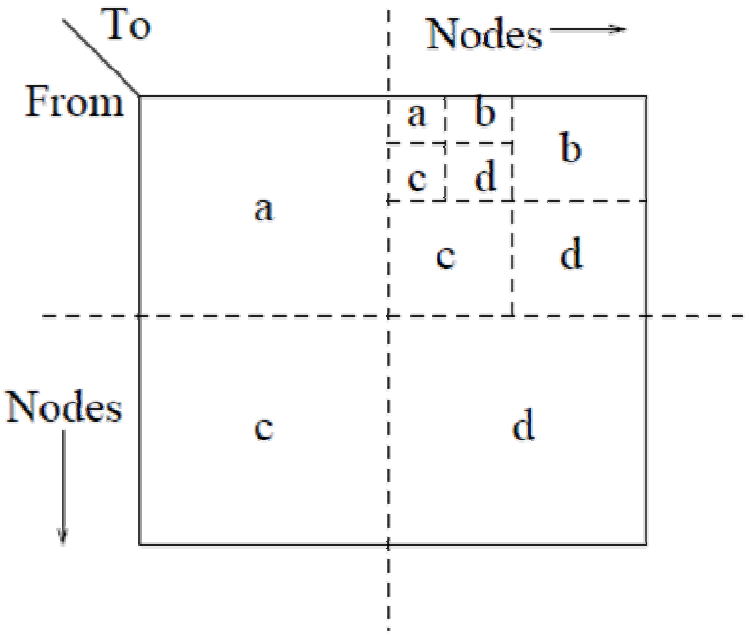
Scales to 1T nodes

- Motivated by GRAPH500 benchmark
- Typically also need for randomized fitting procedures

# Graph 500 Model: Stochastic Kronecker Graphs (SKG)

Chakrabarti , Zhan, & Faloutsos, SDM04; Leskovec et al., JMLR, 2010

- SKG Inputs
  - L = # of levels
  - T = 2 x 2 generator matrix (entries sum to 1)
  - M = # edges
- SKG Edge Insert Procedure
  - Choose a quadrant of the adjacency matrix proportional to entries of T
  - Repeat for a total of L times to land at a single entry of the matrix
- Notes
  - Size of adjacency matrix is $2^L$ x $2^L$
  - Some edges may be duplicates or self-links and are ignored
  - Edge generation is fully parallelizable
  - We make the graph undirected in our studies
  - Fitting to real data using "KronFit" takes between 7 mins for 20K nodes to 4 hrs for 500K nodes




Graph 500 Parameters

$T = [0.57, 0.19; 0.19, 0.05]$

$L \in \{26, 29, 32, 26, 39, 42\}$

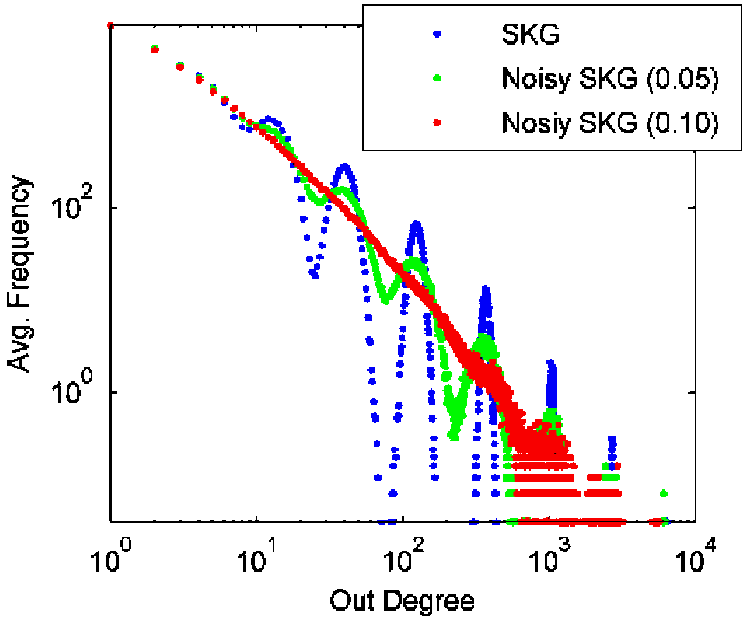
$M = 16 \times 2^L$

# Degree Distribution of SKG

Seshadhri, Pinar & Kolda, arXiv:1102.5046, Sept 2011; short version in ICDM11

- Standard degree distribution has large oscillations
  - Theorem: Between lognormal and exponential tail
- Choose fixed random value $^{1}_{i}$ for each of the L levels
  - Formula provided in paper
  - Noise goes down as size grows
- Level-specific generator matrix is given by

$$T_i = \begin{bmatrix} a - \frac{2\mu_i a}{a+d} & b + \mu_i \\ b + \mu_i & d - \frac{2\mu_i d}{a+d} \end{bmatrix}$$




SKG for Graph 500 for L=16

# Isolates in SKG for Graph 500

Seshadhri, Pinar & Kolda, arXiv:1102.5046, Sept 2011; short version to appear in ICDM11

- Assume symmetric generator, i.e., b=c, and L even
- Number of isolates is

$$I = \sum_{r=-L/2}^{L/2} \binom{L}{L/2+r} \exp(-2\lambda\tau^r),$$

$$\tau = (a+b)/(1-(a+b))$$

$$\lambda = \frac{M}{N}[4(a+b)(1-(a+b))]^{L/2}$$

- Impacts benchmark because number of nodes is less than anticipated and average degree is much higher!

| L | Isolated Nodes | Avg. Degree |
|---|---|---|
| 26 | 51% | 32 |
| 29 | 57% | 37 |
| 32 | 62% | 41 |
| 36 | 71% | 55 |
| 39 | 71% | 55 |
| 42 | 74% | 62 |

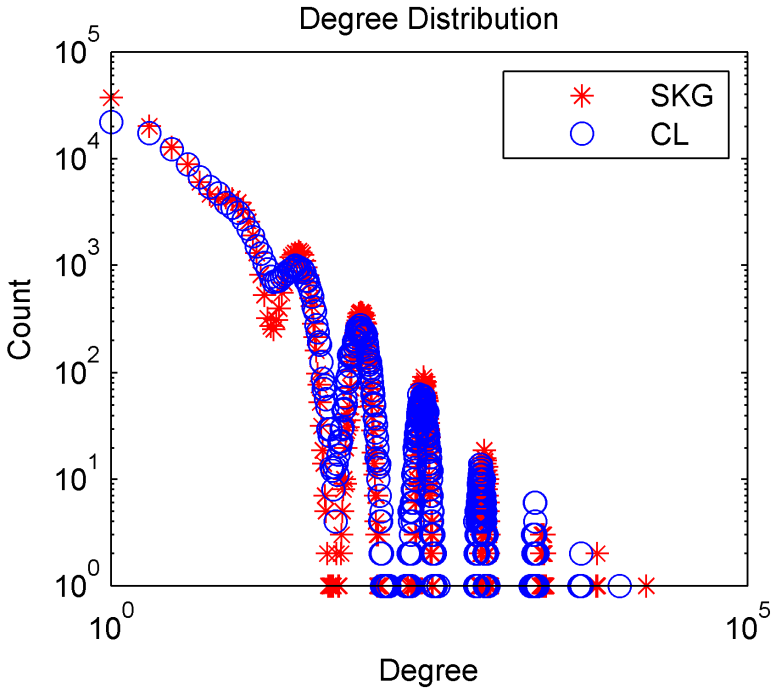
# The Chung-Lu (CL) Model: An Alternative to SKG

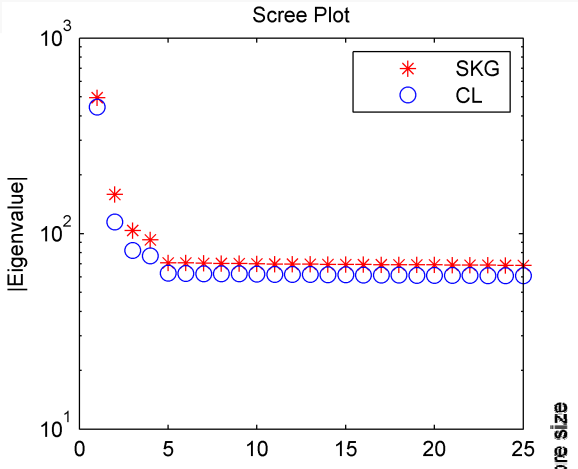
Chung & Lu, PNAS, 2002; Chung & Lu, Annals of Combinatorics, 2002

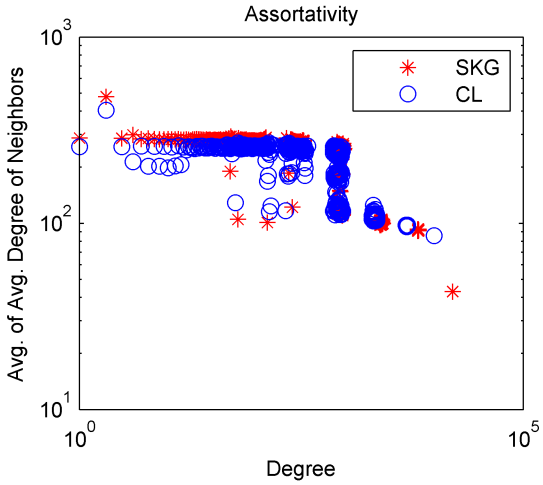
- Is there a model that can give better fits to degree distribution than SKG and is easier to fit?
- Chung-Lu Model
  - $d_i$ = (desired) degree of node i ($\sum d_i$ = M = # edges)
  - Probability of single edge insertion at (i,j) = $d_i d_j / M^2$
- Scalable implementation chooses source and sink for each edge independently
  - Probability of choosing node i is $d_i / M$
- Other names:
  - Configuration Model (Newman, SIREV, 2003)
  - Weighted Erdös-Rényi Model

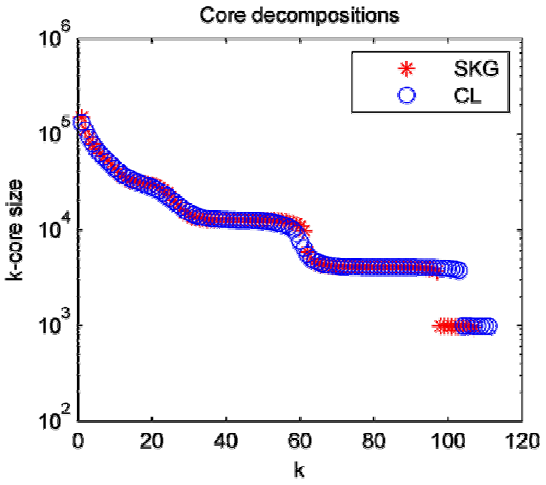
# Similarity of CL to SKG for Graph 500

Fit CL to the degree distribution produced by SKG for Graph 500 with L = 18

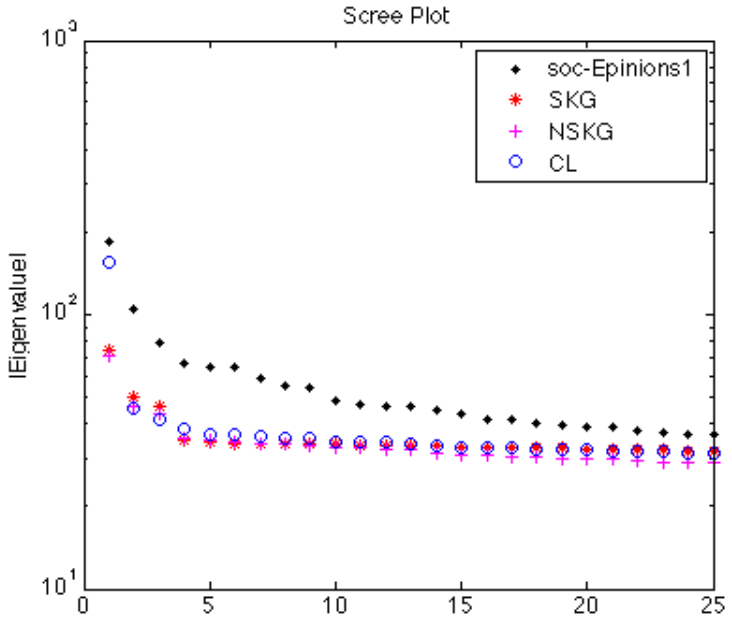


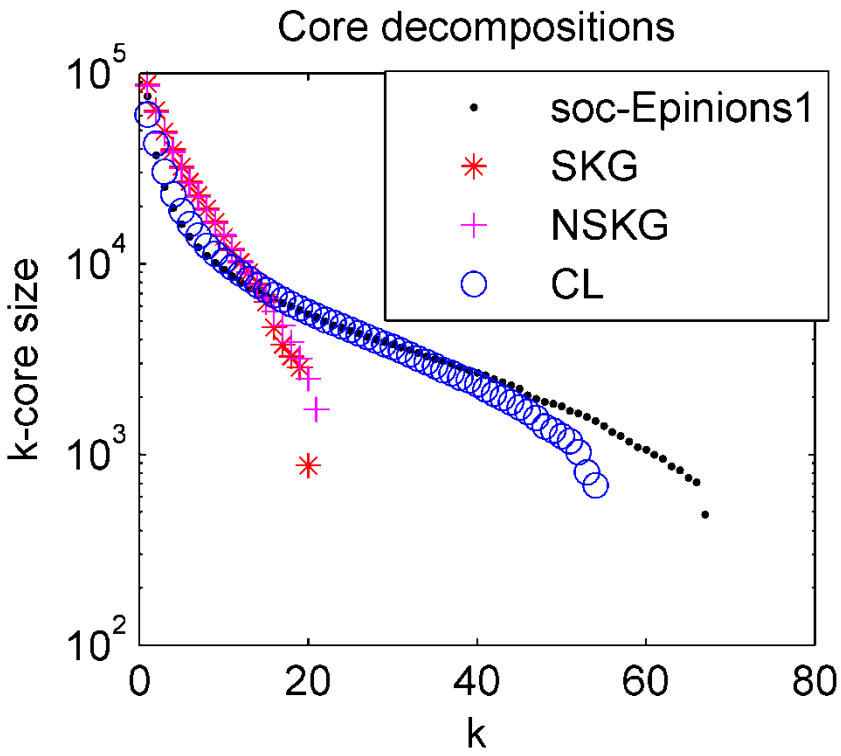



# CL Better Fit to E-values, Core, Etc.



# Both CL and SKG Produce Poor Clustering Coefficients

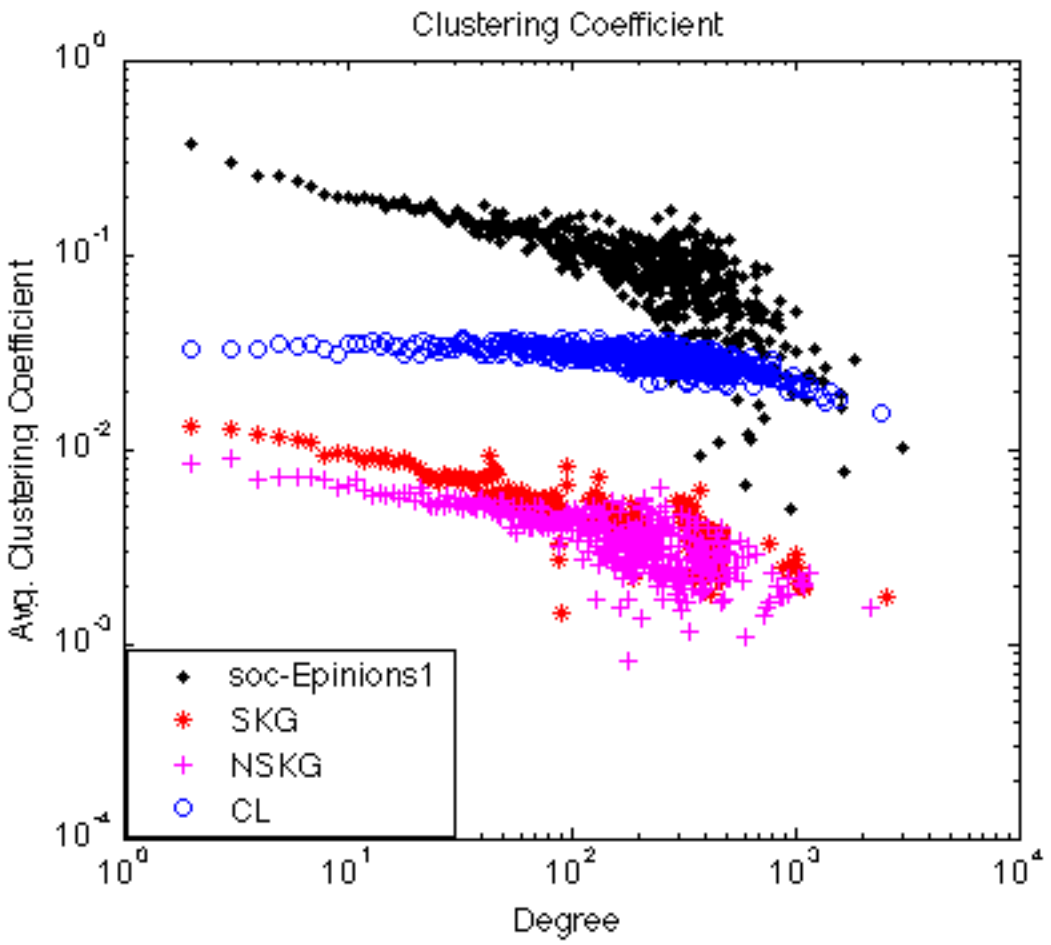
**Clustering Coefficient**

$$C_i = \frac{t_i}{\binom{d_i}{2}}$$

$t_i$ = # triangles at vertex i
$d_i$ = degree of vertex i

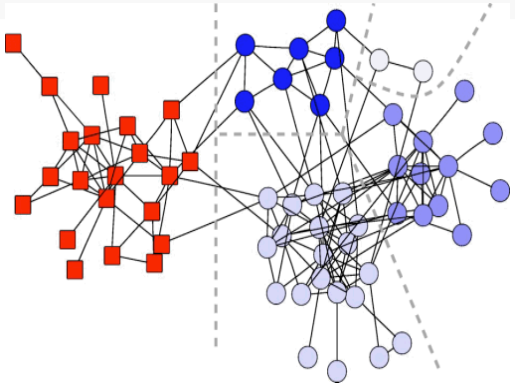
**Global Clustering Coeff.**

$$C = \frac{\sum_i t_i}{\sum_i \binom{d_i}{2}}$$

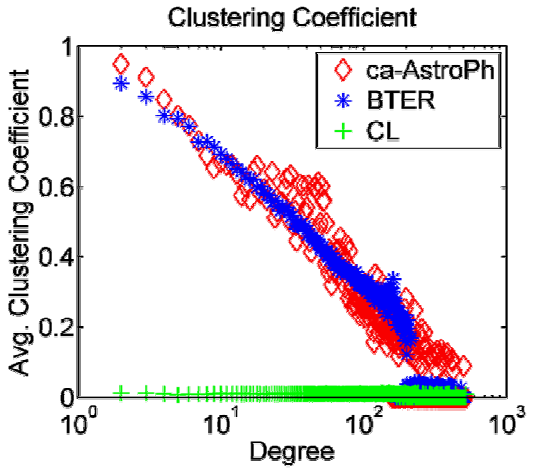


# Community Structure in Graphs

- Numerous community finding algorithms exist
  - Difficult to validate
  - Trouble in finding full range of sizes
- Instead, use related measures like clustering coefficient
  - Triangles arise because of community structure
- What "community" structure must be present to ensure a high clustering coefficient, especially for low-degree nodes?
  - How many communities?
  - What do they look like?



M.E.J. Newman and M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69, 026113, 2004.



# Building the basis for a model

## Empirical Observations

- Networks contain communities.
- The sizes of the communities are small, and do not grow (or grow very slowly) for larger graphs. (e.g., Dunbar number).
- Clustering coefficients are highest for small degree vertices.
- Vertex degree is correlated to the average degree of neighbors.

## Theoretical Analysis

- Theorem: *If a community has s edges then there must be Ω(√s) vertices with degree Ω(√s).*
- Corollary: For graphs with skewed degree distribution, the number of communities grows with the number of nodes.
- Corollary: Within a community the degree should be small.

**Hypothesis: Real-world interaction networks consist of a scale –free collection of dense Erdős-Rényi graphs.**

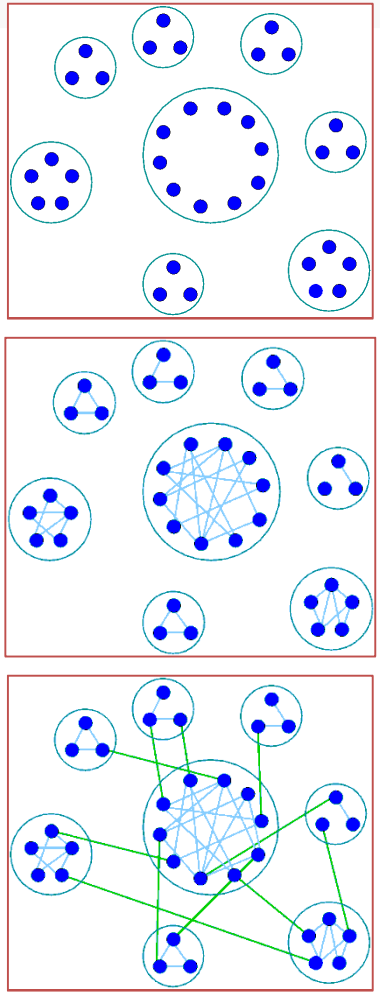
We are not only trying to build a formal model, we are trying to formalize the model building process itself.

# BTER: A New Model with Explicit Community Structure

**Preprocessing:** Create explicit communities

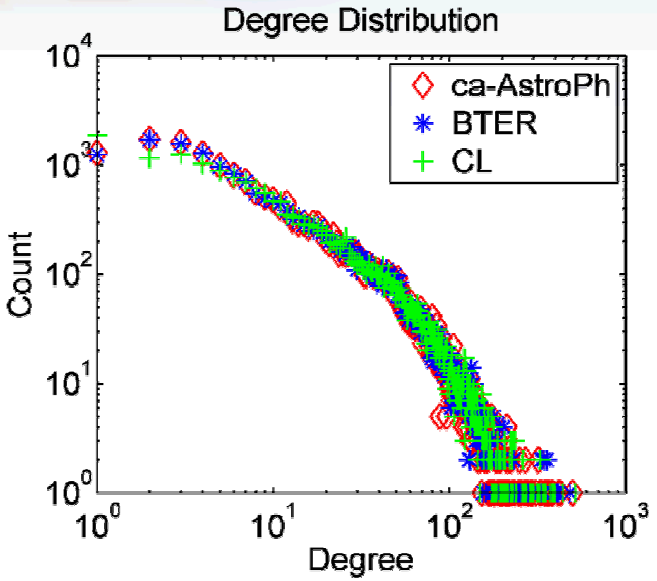
**Phase 1:** Erdös-Rényi graphs in each community

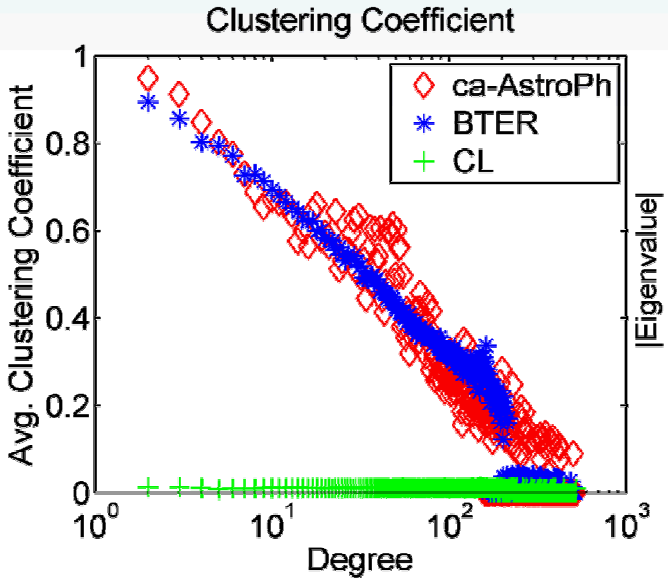
**Phase 2:** CL model on "excess" degree

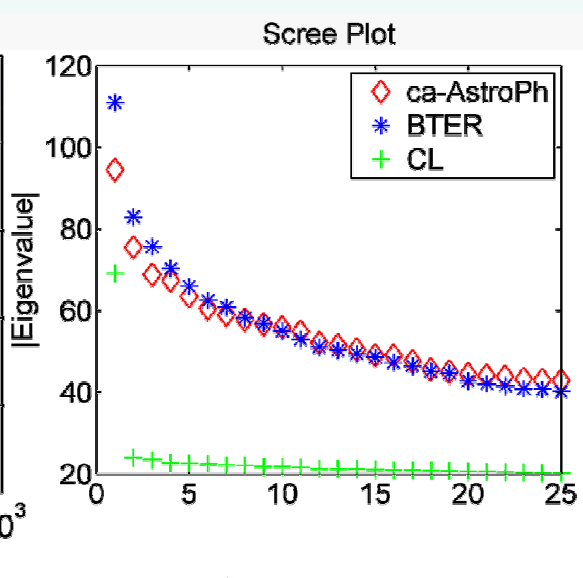


- **Idea:** Model should have O(n) communities in order to get high clustering coefficients
- **Preprocessing:** Generate communities
  - Determined by **desired degree distribution**
  - All nodes have (close to) the same degree
  - Size of cluster = min degree + 1
- **Phase 1:** Generate ER graph on each community
  - User must **specify connectivity coefficient** for each community, $\frac{1}{2}_k$
  - We use a function of the min degree in the community, $d_k$
- **Phase 2:** Generate CL graph on "excess" degree
  - $e(i) = d(i) - \frac{1}{2}_k d_k$ where vertex i is in community k

# Co-authorship (ca-AstroPh)



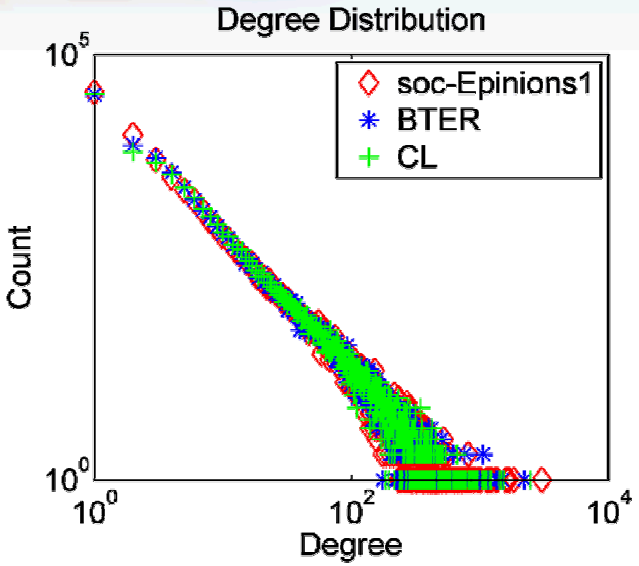



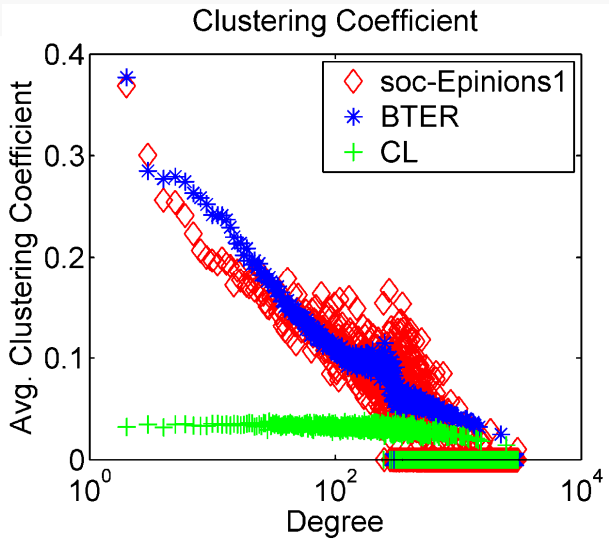



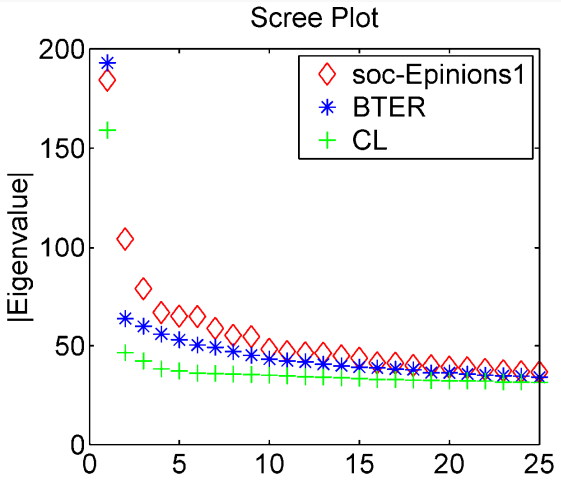

- 18,771 nodes 396,100 edges; based on arxiv repository
- Global clustering coefficients:
  - Original: 0.32, BTER: 0.31, CL: 0.01
- Normalized size of the connected components
  - Original: 0.95, BTER: 0.86, CL: 1.0

Eigenvalues are not determined by degree distribution

# Trust Network

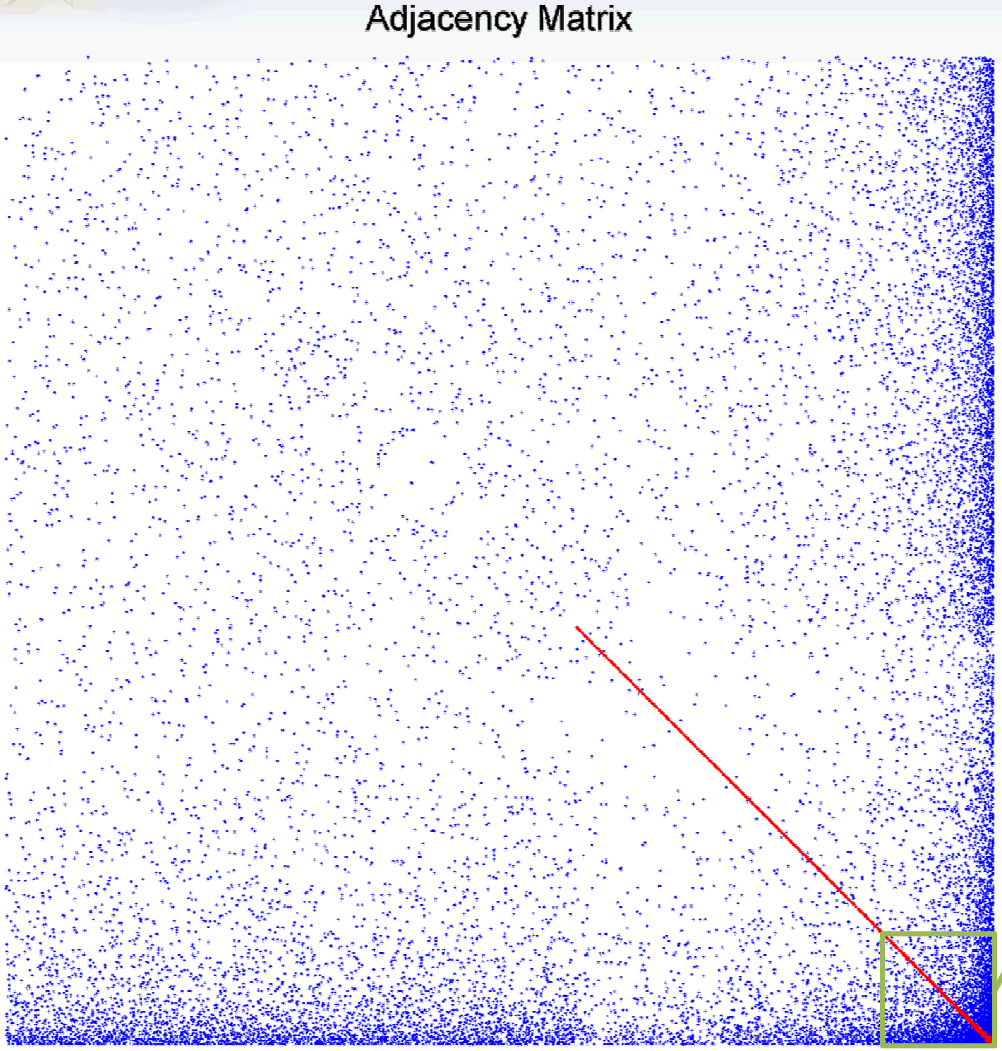


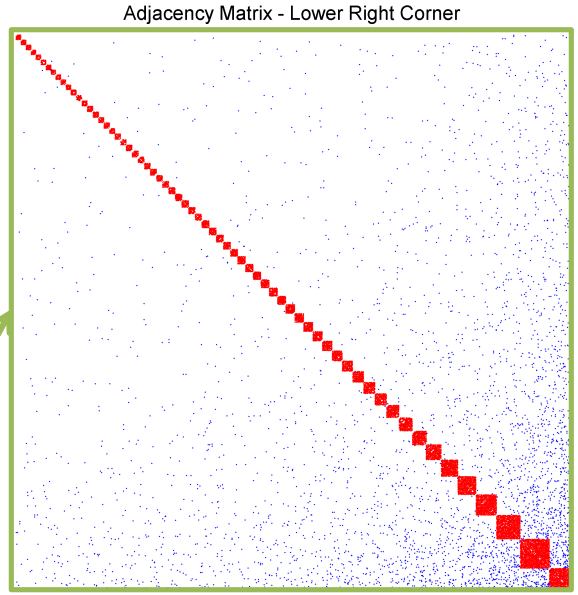




- 75,879 vertices, 811,480 edges; based on Epinion web site; edges represent rust between two users
- Global clustering coefficients:
  - Original: 0.07, BTER: 0.07, CL: 0.03
- Normalized size of the connected components
  - Original: 1.0, BTER: 0.96, CL: 0.98

# Visualization of BTER Adjacency Matrix

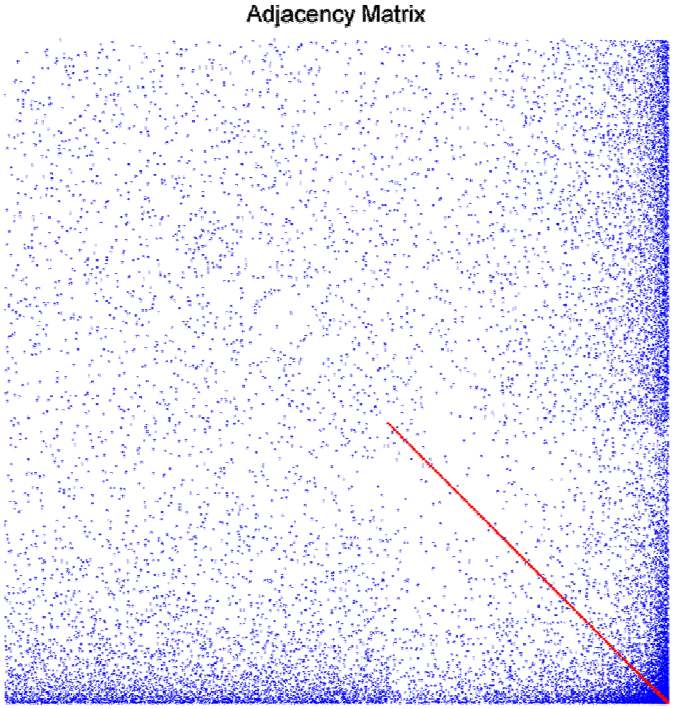



Red = Phase 1
Blue = Phase 2



# Observations on BTER

- Requires desired degree distribution
  - Approximation can be used to save space
- Phase 1: Communities
  - All nodes have the same (expected) degree; easy generation of dense subgraphs
  - But there are ways we could allow the communities to be heterogeneous
  - Community edge density is a parameter which may be tuned to fit real data
- Phase 2: Uses expected excess degree
  - Enables “streaming edge” generation
- BTER edge generation is fully parallelizable
  - community membership for each node
  - edge density for each community
  - excess degree (in expectation) for each node

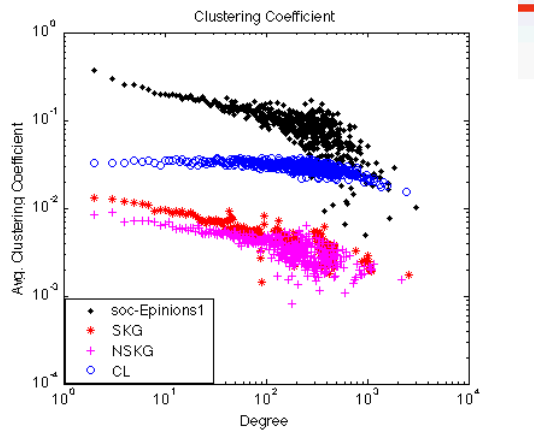


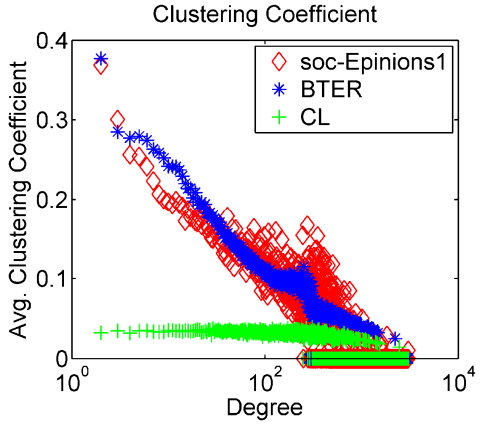
# Concluding Remarks

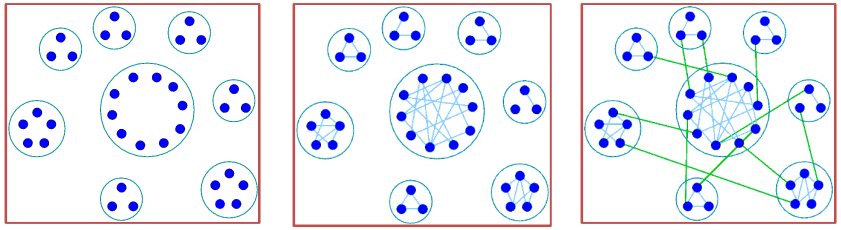
- Modeling of graphs underlie many challenges for principled graph analysis
- The challenge is not in building a formal model, but formalizing the modeling process itself.
- We proposed the Block Two-Level Erdös-Rényi (BTER)
  - New theory says there must be many dense subgraphs for high clustering coefficient
  - New BTER model explicitly creates dense communities using ER
  - Exceptional similarities to real data in terms of clustering coefficients and eigenvalues
- The code is available at

http://www.sandia.gov/~tgkolda/bter_supplemet/.

- For more information,
  - Ali Pinar apinar@sandia.gov



# Relevant Publications

- Modeling of graphs
  - C. Seshadhri, T. Kolda, and A. Pinar, “The Blocked Two-Level Erdos Renyi Graph Model,” submitted for journal publication
  - C. Seshadhri, A. Pinar, and T. Kolda, “An In Depth study of Stochastic Kronecker Graphs," submitted for journal publication.
  - A. Pinar, C. Seshadhri, and T. Kolda, “Comparison of Scalable Graph Generation Models,” submitted for conference publication.
  - C. Seshadhri, A. Pinar, and T. Kolda, “An In Depth study of Stochastic Kronecker Graphs,” to appear in Proc. Int. Conf. on Data Mining (ICDM).
- Sampling Graphs
  - I. Stanton and A. Pinar, “Constructing and uniform sampling graphs with prescribed joint degree distribution using Markov Chains,” submitted for journal publication.
  - I. Stanton and A. Pinar, “Sampling graphs with prescribed joint degree distribution using Markov Chains,” Proc. ALENEX 11.
- Community structure
  - M. Rocklin, and A. Pinar, “On Clustering on Graphs with Multiple Edge Types,” submitted for journal publication.
  - M. Rocklin and A. Pinar, “Latent Clustering on Graphs with Multiple Edge Types,” Proc. 8th Workshop on Algorithms and Models for the Web Graph (WAW11).
  - M. Rocklin and A. Pinar, “Computing an Aggregate Edge-weight function for Clustering Graphs with Multiple Edge Types,” in Proc. 7th Workshop on Algorithms and Models for the Web Graph (WAW10).

# Supplementary Material

# Naïve Addition of Noise

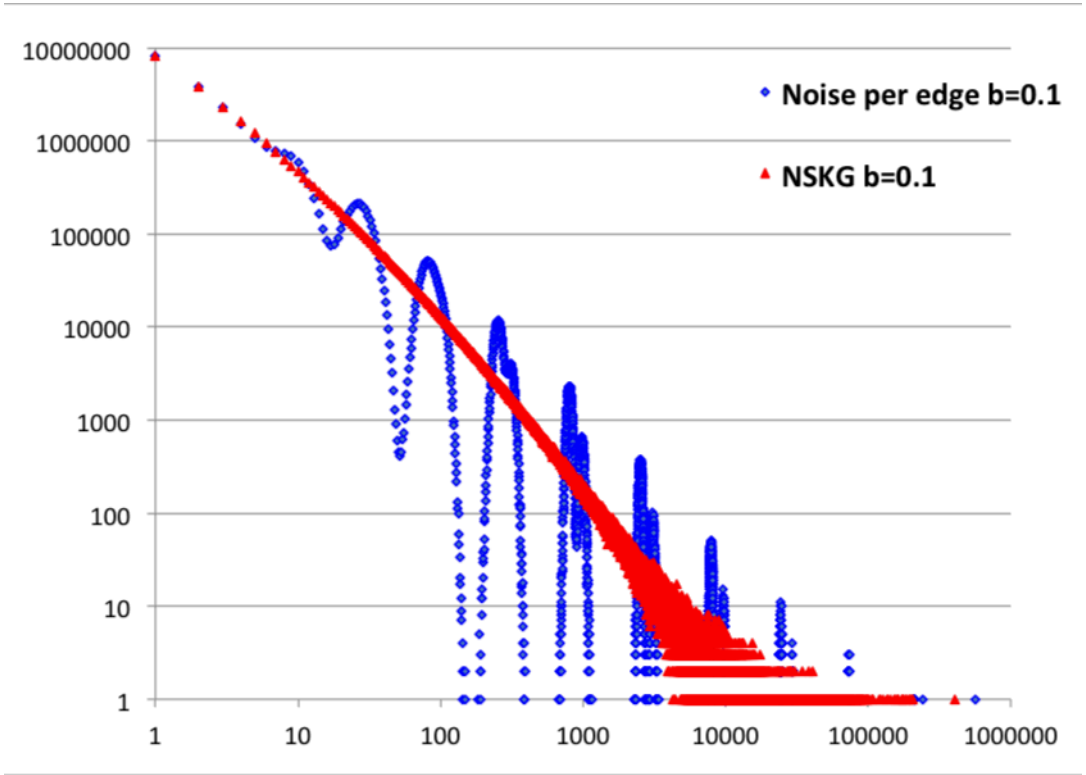
Seshadhri, Pinar & Kolda, arXiv:1102.5046, Sept 2011; short version in ICDM11

"To smooth out fluctuations in the degree distributions, we add some noise to the (a, b, c, d) values at each stage of the recursion and then renormalize (so that a+b+c+d = 1)." – CZF04

Adding noise at every edge insertion *does not work*!

Example at left in Graph500 with L=26.

Figure by Todd Plantenga, using his Hadoop MapReduce SKG implementation.



# Fitting CL to the SKG deg. dist. yields exceptionally similar graph

Seshadhri, Pinar & Kolda, preprint, Oct 2011

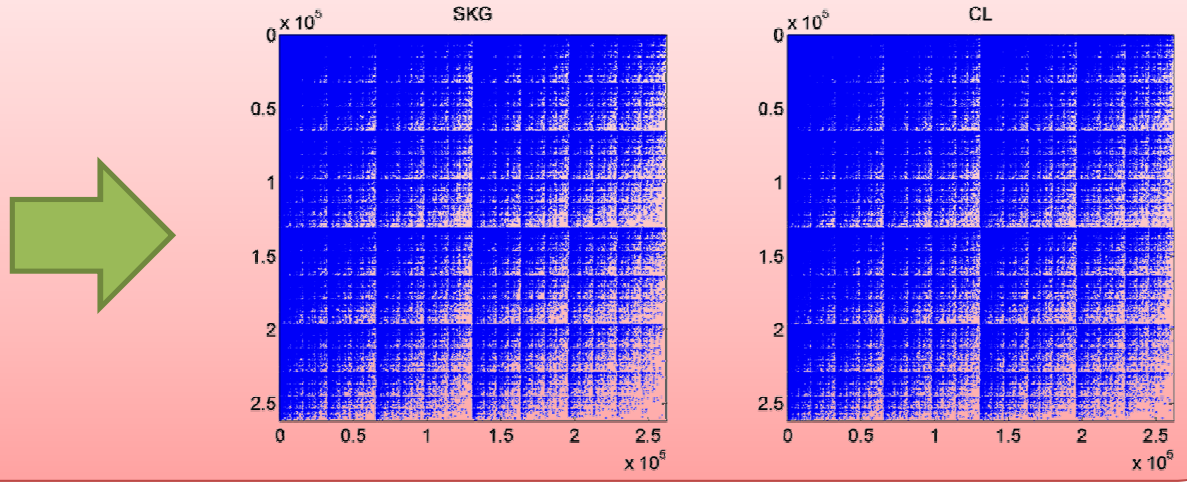
Let z(i) = # zeros in binary representation of i (likewise for j).
Let c = # zeros common to binary representations of i and j.

$$P_{\mathrm{SKG}}(i,j) = a^c b^{z(i)-c} c^{z(j)-c} d^{L-z(i)-z(j)-c}$$

$$P_{\mathrm{CL}}(i,j) = (a+b)^{z(i)}(c+d)^{L-z(i)}(a+c)^{z(j)}(b+d)^{L-z(j)}$$

If a/b = c/d, then $P_{SKG} = P_{CL}$

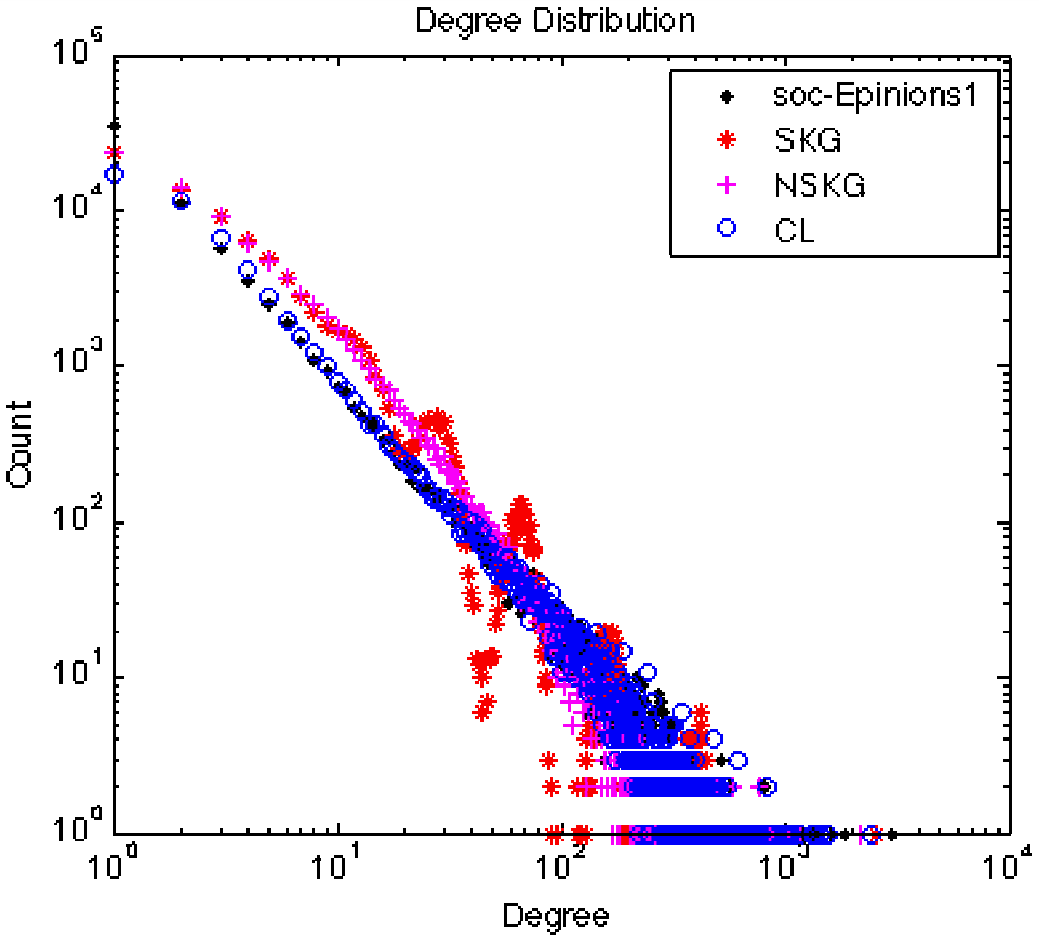
*Note similarity of adjacency matrices generated by SKG and CL for Graph 500*



# CL Trivial to Fit to Real Data and More Accurate than SKG or NSKG

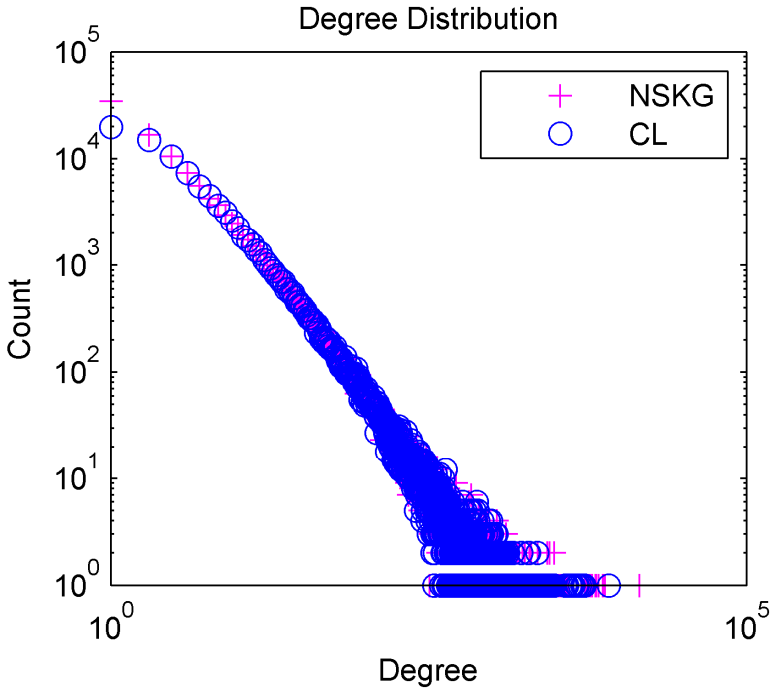
CL has a closer fit to the degree distribution than SKG or NSKG.

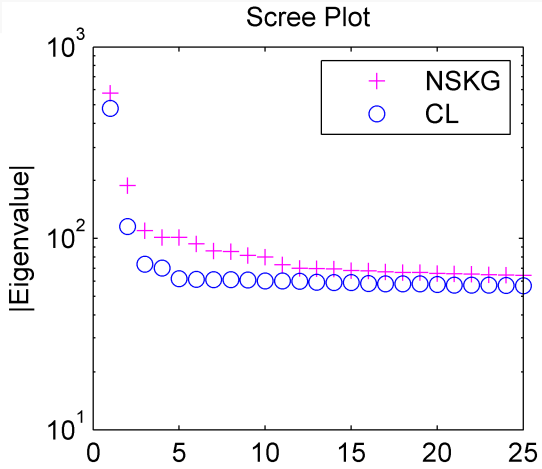
Using SKG parameters from Leskovec et al., JMLR, 2010.

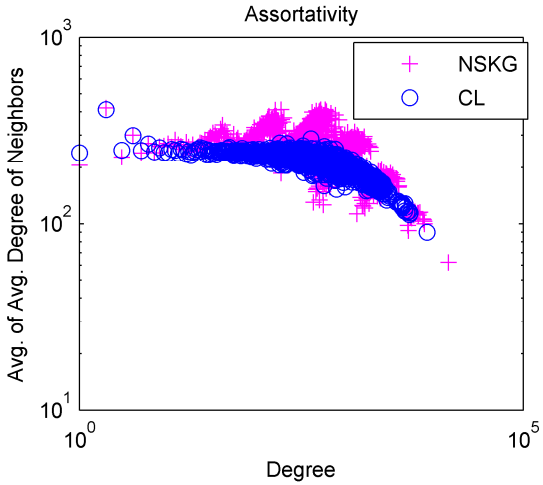


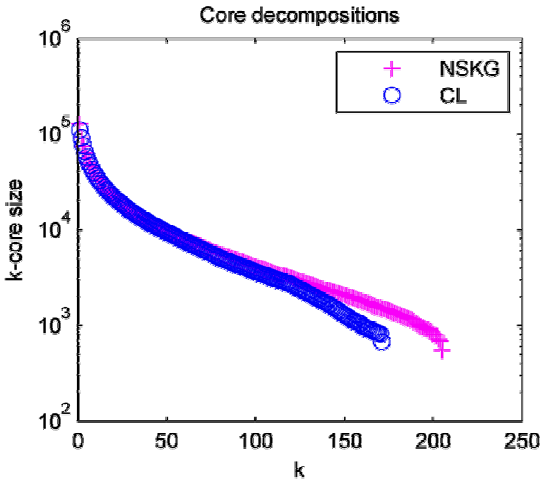
# Similarity of CL to NSKG for Graph 500

Fit CL to the degree distribution produced by NSKG for Graph 500 with L = 18



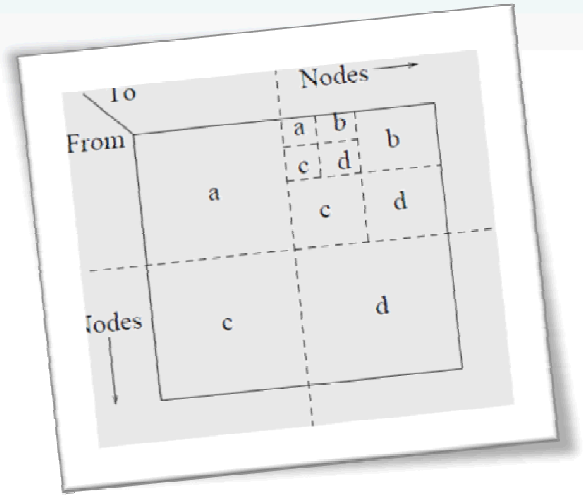



# CL vs. SKG

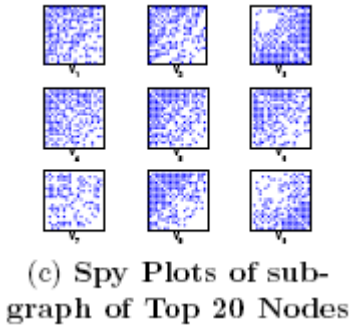
- SKG Model (generator for Graph 500)
  - Only 6 parameters!
  - Embarrassingly parallel edge generation
    - Some work must be done to remove duplicate links, if desired
  - Constrained to lognormal degree distribution (assuming noise)
  - Difficult to fit to real data
    - Minutes to hours and cannot reproduce degree distribution
- CL Model (should be used at least as a control)
  - Requires full degree distribution
    - May be possible to approximate with a few parameters
  - Embarrassingly parallel edge generation
    - Still need work to remove duplicates
  - Trivial to fit to real data by using degree distribution
    - Note that the fit of CL w.r.t. SKG likelihood function is actually quite good
- Neither model yields good high clustering coefficients
  - May not be appropriate for capturing community structure



# COMMUNITY STRUCTURE IN GRAPHS

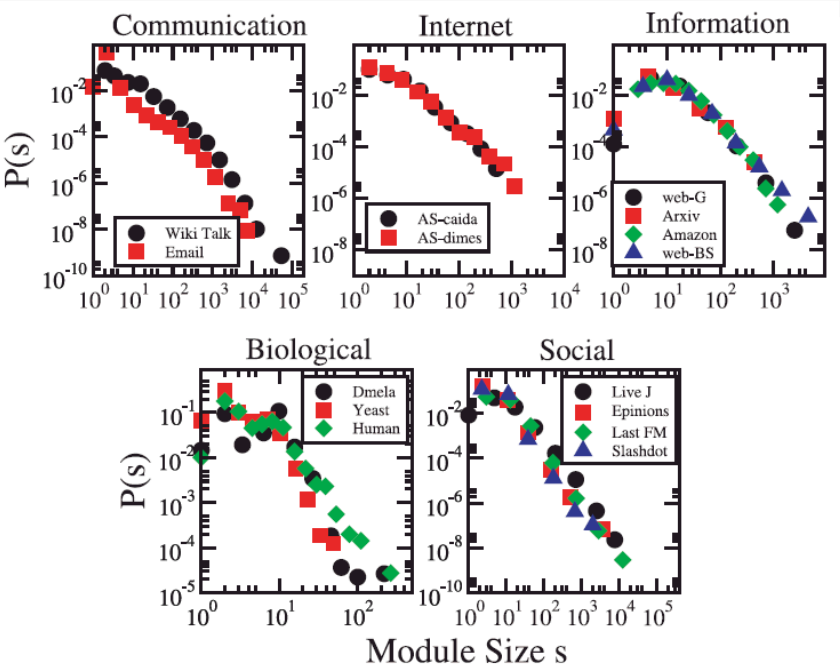
# Related Work on Community Structure

- Lancichinetti et al. (PLoS 2010) shows that there are many communities and a variety of sizes
  - Use two different methods for detecting communities
  - Communities differ, but trends in sizes are the same
- Eigenspoke analysis of Prakash et al. (2010) reveals dense subgraphs



(c) Spy Plots of subgraph of Top 20 Nodes

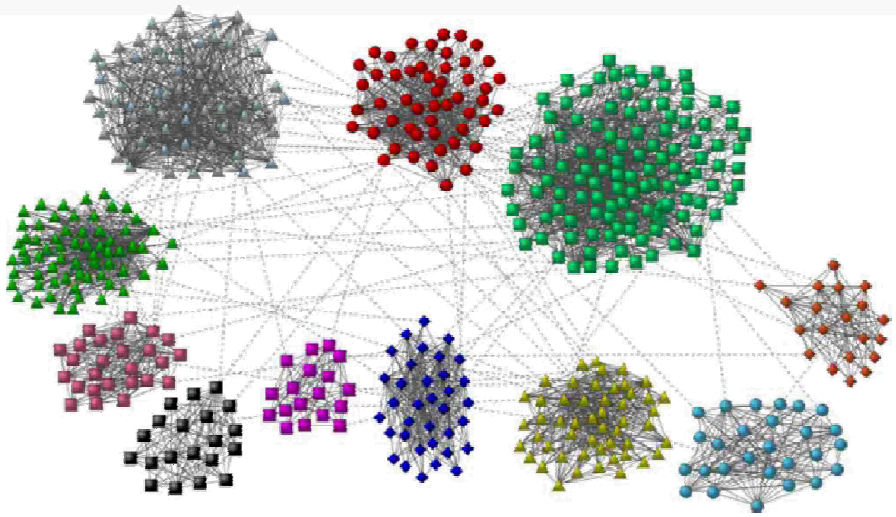
Prakash et al., EigenSpokes: Surprising Patterns and Scalable Community Chipping in Large Graphs, *Advances in Knowledge Discovery and Data Mining,* **2010**




Lancichinetti, Kivelä, Saramäki, & Fortunato, Characterizing the Community Structure of Complex Networks, *PLoS ONE,* **2010**

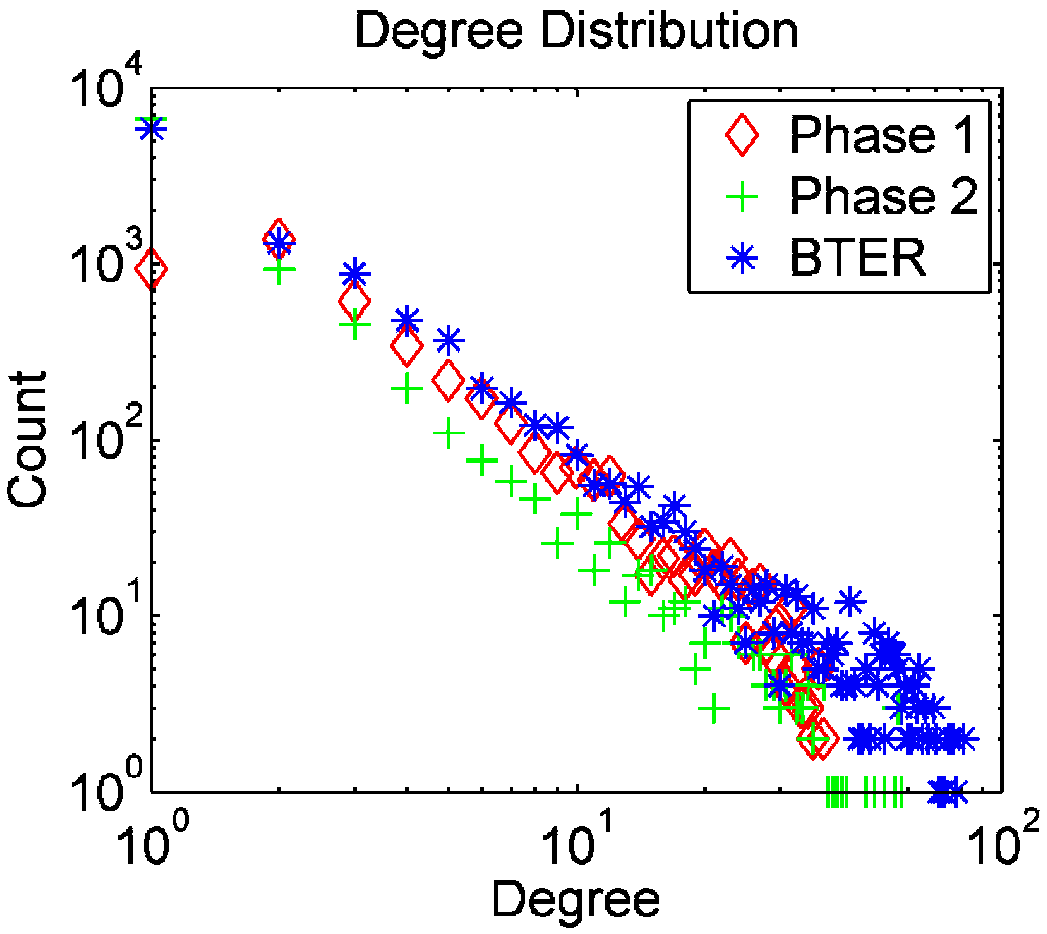
# Relationship of BTER and LFR Models

- Both explicitly insert communities
- Community structure
  - BTER: generated automatically according to degree distribution
  - LFR: power law distributed
- Assignment of nodes to communities
  - BTER: Determined during community building phase
  - LFR: Random assignments; any node can go into a community where the size is higher than its "internal degree"
- Internal vs. external links
  - BTER: Varies by node
  - LFR: constant proportions for all nodes
- Community sizes
  - BTER: All sizes down to 3 nodes
  - LFR: Minimum community size specified by user
- Scalabilty
  - FLR community assignment procedure is not obviously parallelizable



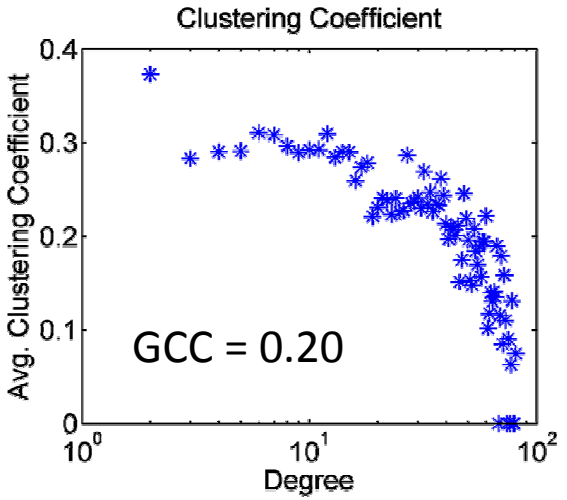
Lancichinetti, Fortunato, & Radicchi, Benchmark graphs for testing community detection algorithms, *Phys. Rev. E,* **2008**

# BTER Illustration for Power Law Distribution

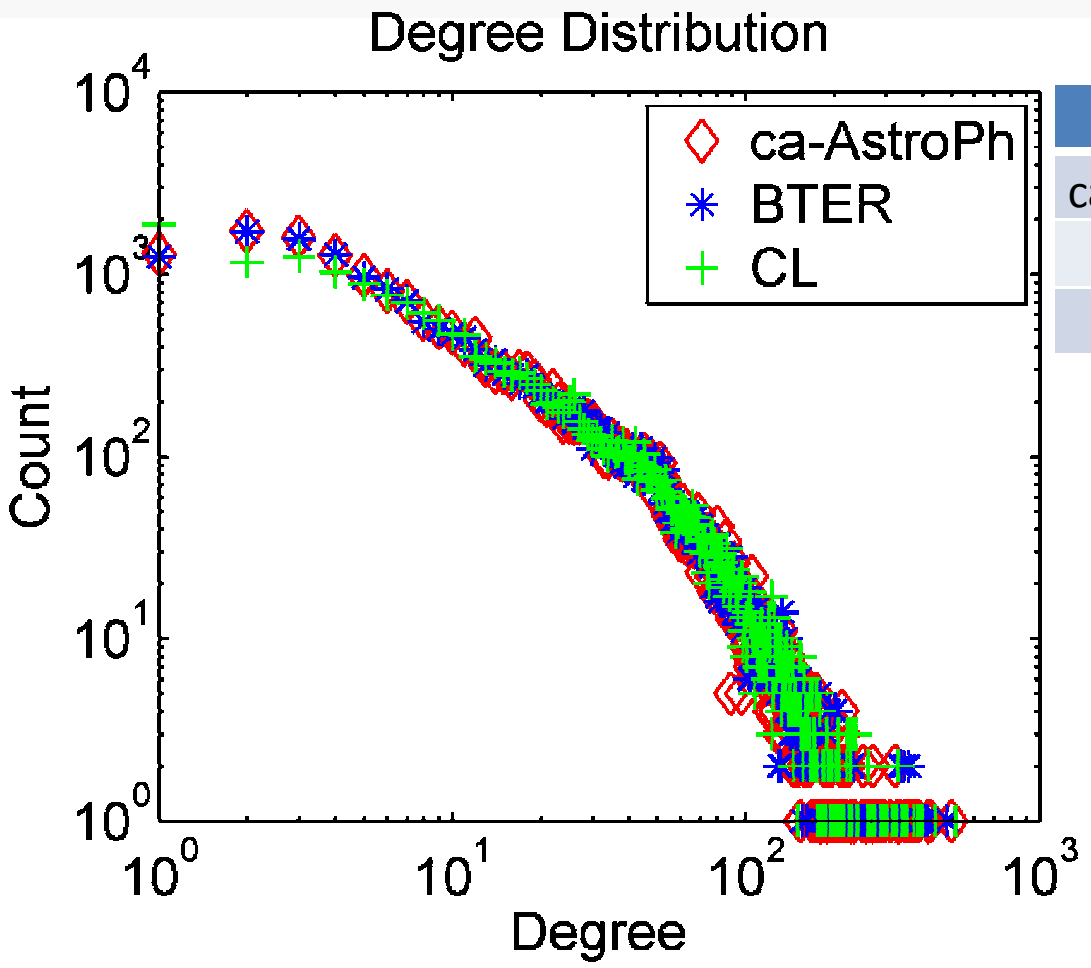



Power Law Gamma = 1.9
Nodes: 10, 269
Total Edges: 38, 628
Phase 1 Edges: 20,426
Phase 2 Edges: 18,344

$$\rho = 0.7\left(1 - 0.2\,\frac{\log(d-1)}{\log(d_{\max}-1)}\right)^2$$



# Co-authorship Network (ca-AstroPh)



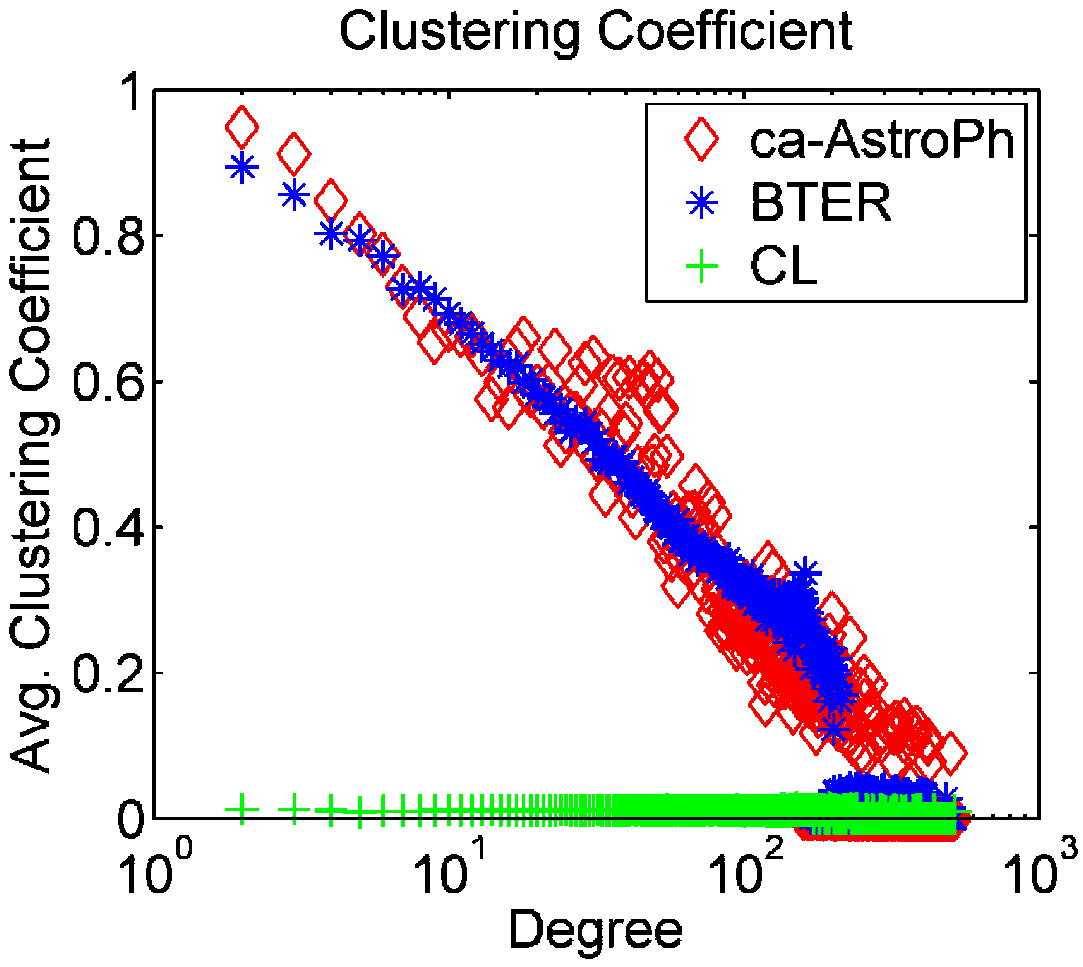

| Graph | Nodes | Edges | LCC % |
|---|---|---|---|
| ca-AstroPh | 18,771 | 396,100 | 95 |
| BTER | 18,681 | 401,788 | 86 |
| CL | 18,352 | 412,384 | 100 |

# Co-authorship Network (ca-AstroPh)

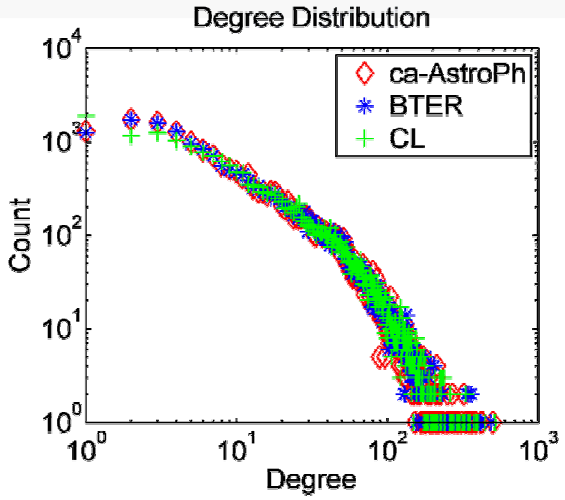
BTER
Phase 1 Edges: 290,268
Phase 2 Edges: 112, 808

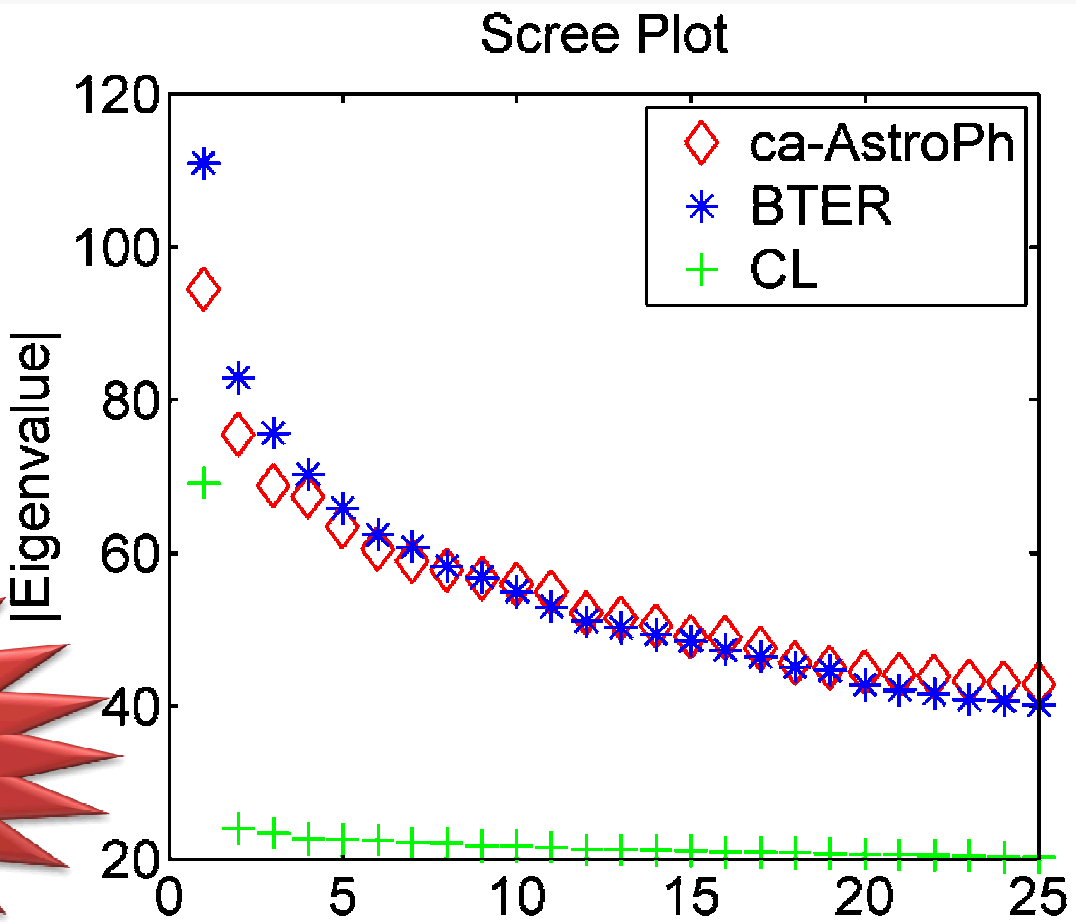
$$\rho = 0.95 \left(1 - 0.5 \frac{\log(d-1)}{\log(d_{\max}-1)}\right)^2$$

| Graph | GCC |
|---|---|
| ca-AstroPh | 0.32 |
| BTER | 0.31 |
| CL | 0.01 |



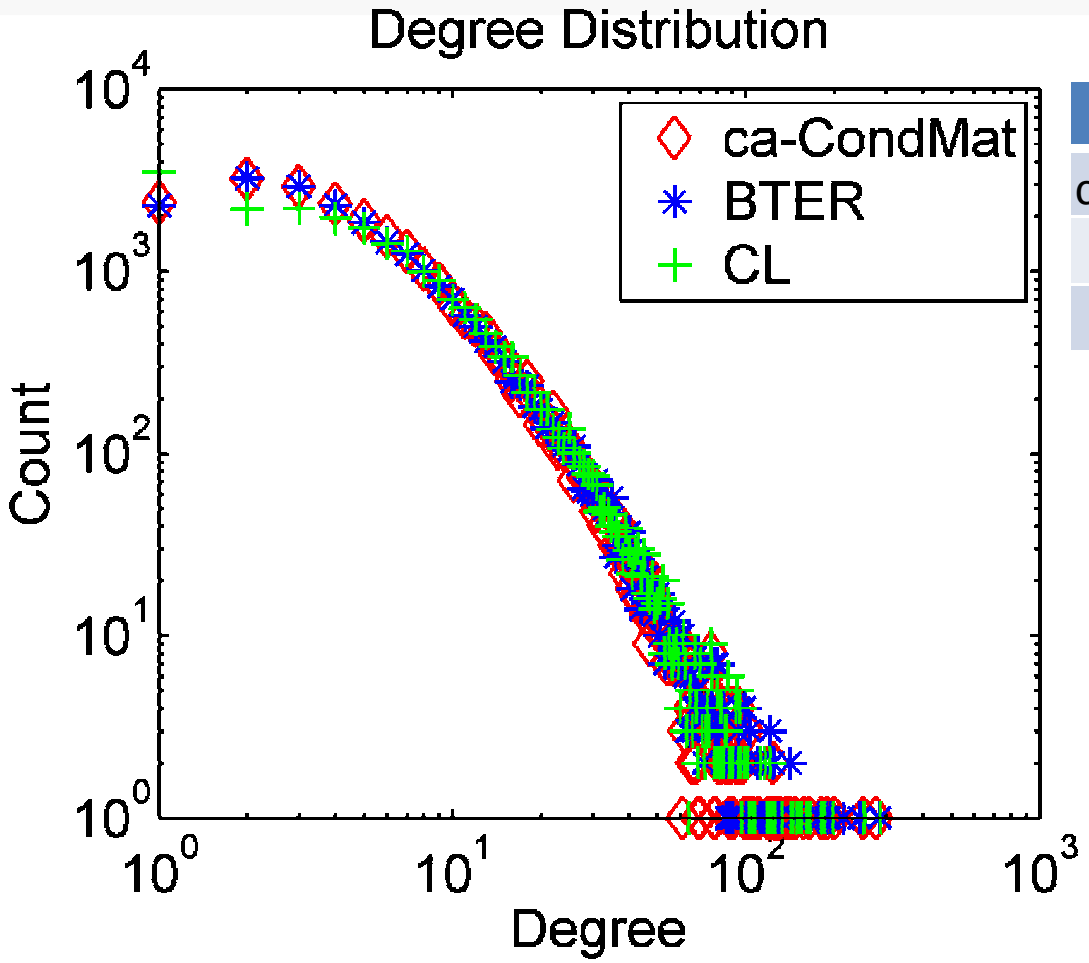
# Co-authorship Network (ca-AstroPh)




Eigenvalues are not determined by degree distribution

# Co-authorship Network (ca-ContMat)



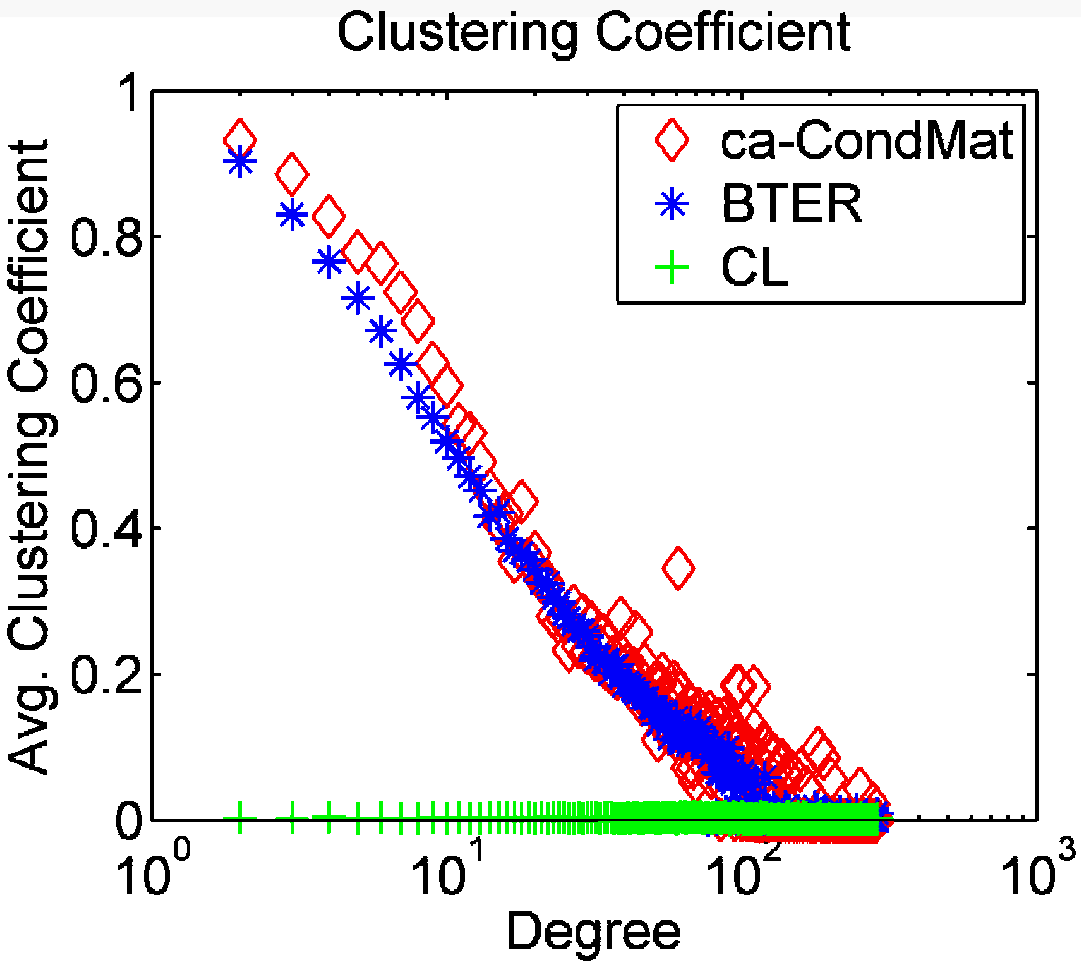

| Graph | Nodes | Edges | LCC % |
|---|---|---|---|
| ca-CondMat | 23,133 | 186,878 | 92 |
| BTER | 22,938 | 190,144 | 82 |
| CL | 22,390 | 194,190 | 100 |

# Co-authorship Network (ca-ContMat)

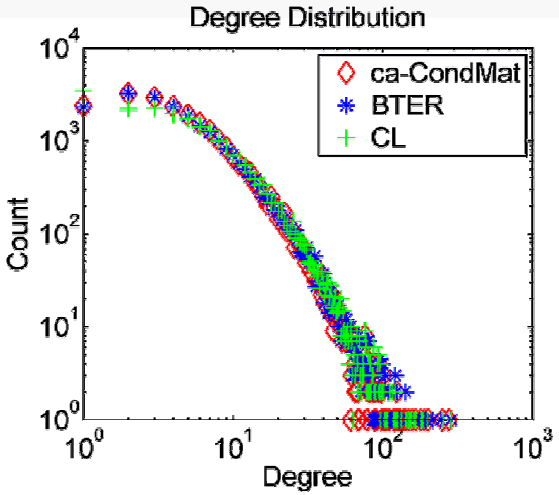
BTER
Phase 1 Edges: 135,246
Phase 2 Edges: 55,124

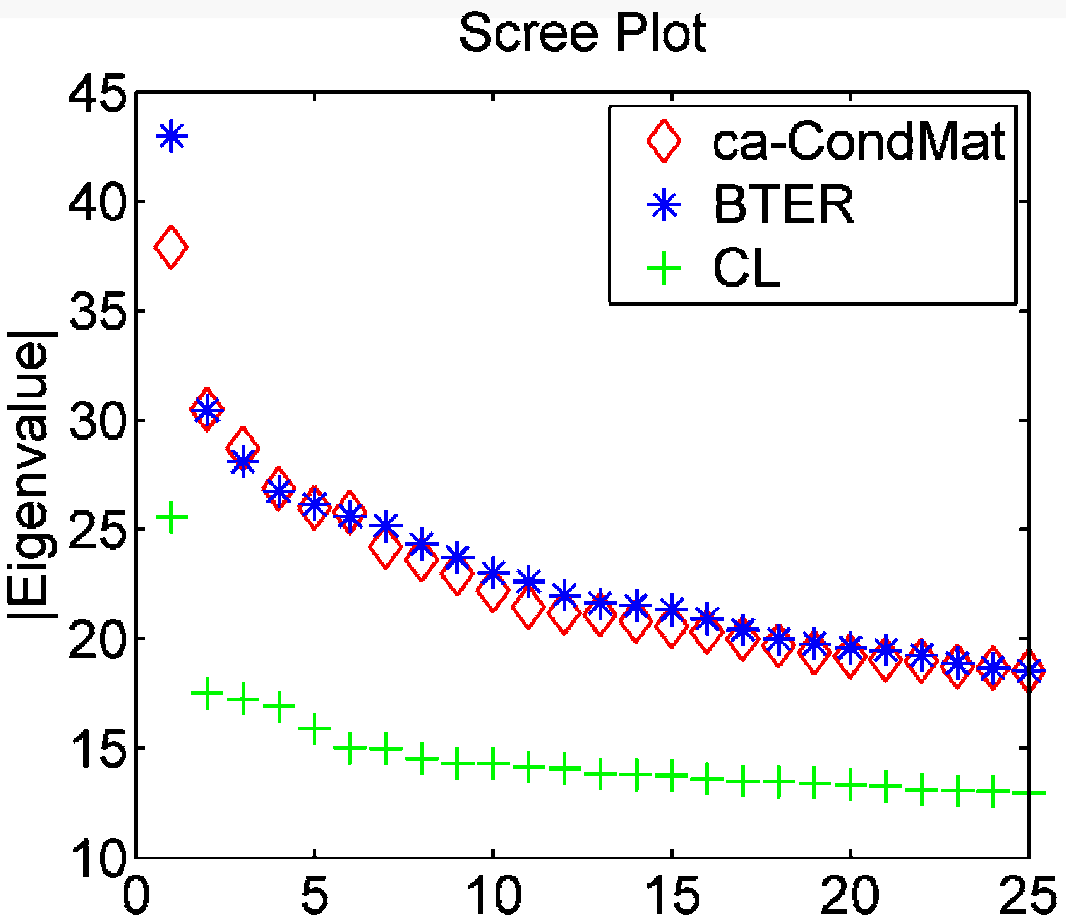
$$\rho = 0.95 \left(1 - 0.95 \frac{\log(d-1)}{\log(d_{\max}-1)}\right)^2$$

| Graph | GCC |
|---|---|
| Ca-CondMat | 0.26 |
| BTER | 0.23 |
| CL | 0.00 |

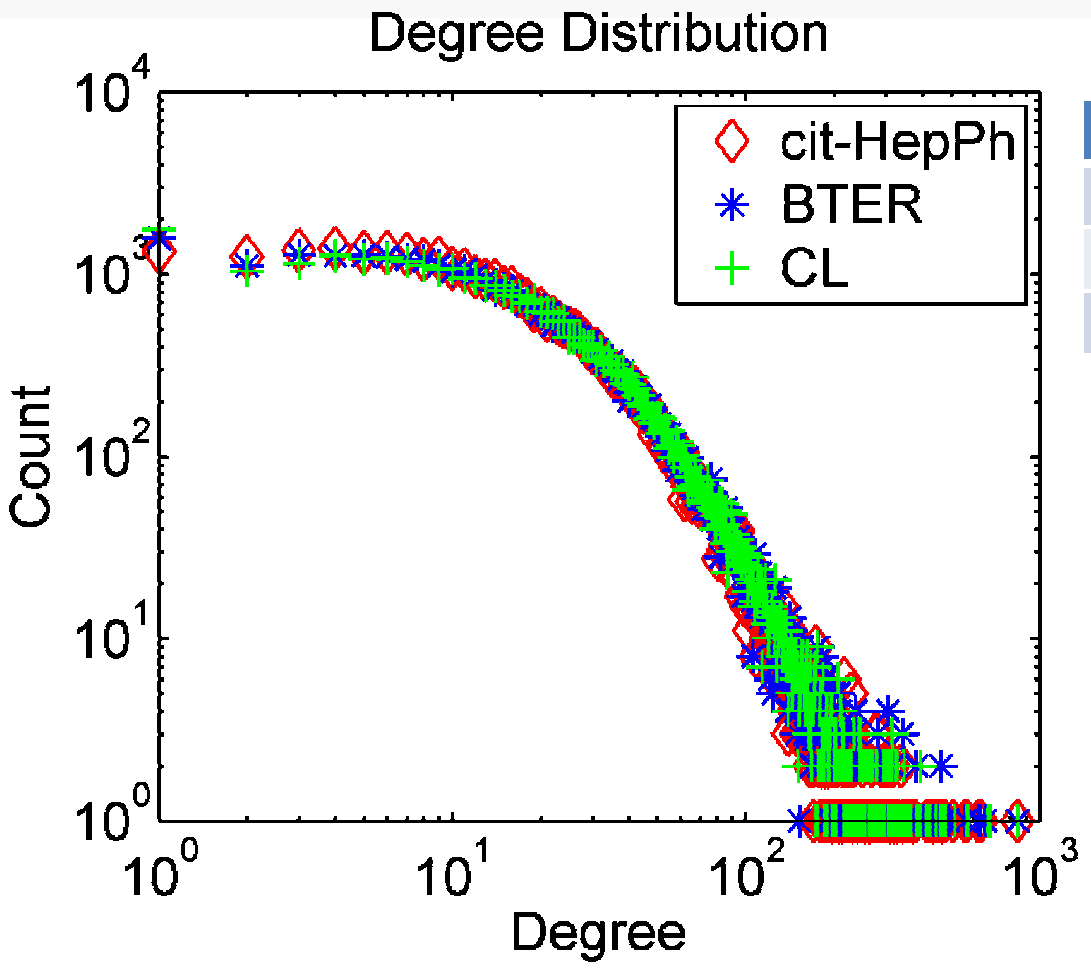


# Co-authorship Network (ca-ContMat)



# Citation Network (cit-HepPh)



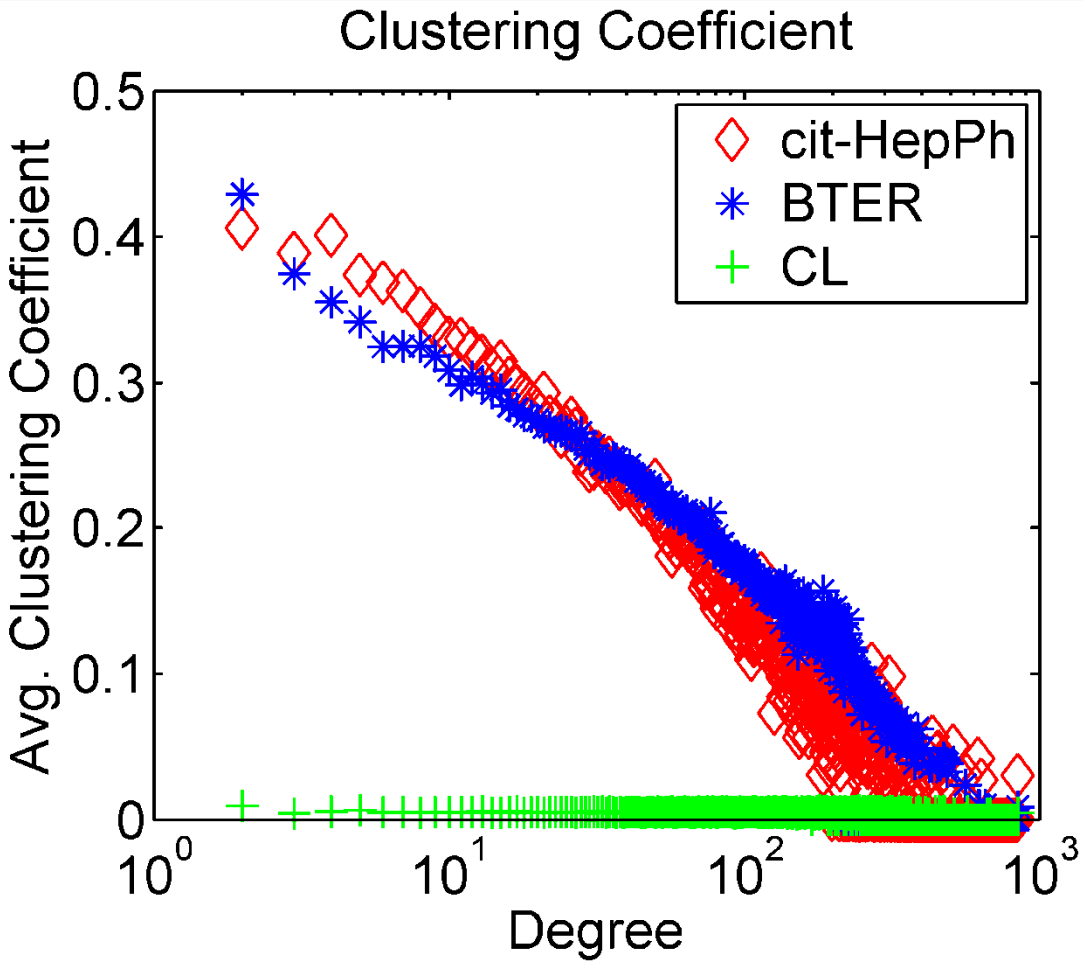

| Graph | Nodes | Edges | LCC % |
|---|---|---|---|
| cit-HepPh | 34,546 | 841,754 | 100 |
| BTER | 34,351 | 870,750 | 99 |
| CL | 34,174 | 880,520 | 100 |

# Citation Network (cit-HepPh)

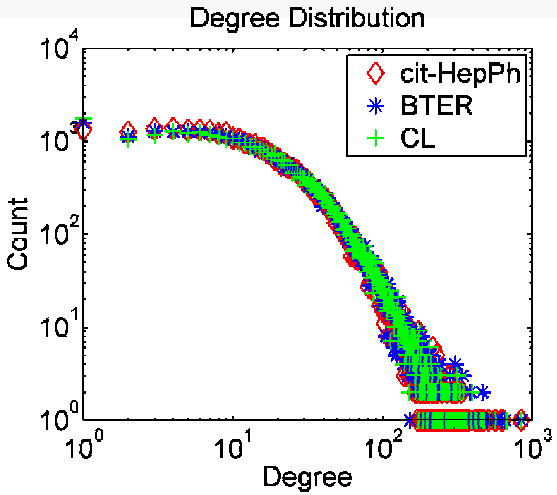
BTER
Phase 1 Edges: 510,864
Phase 2 Edges: 362,358

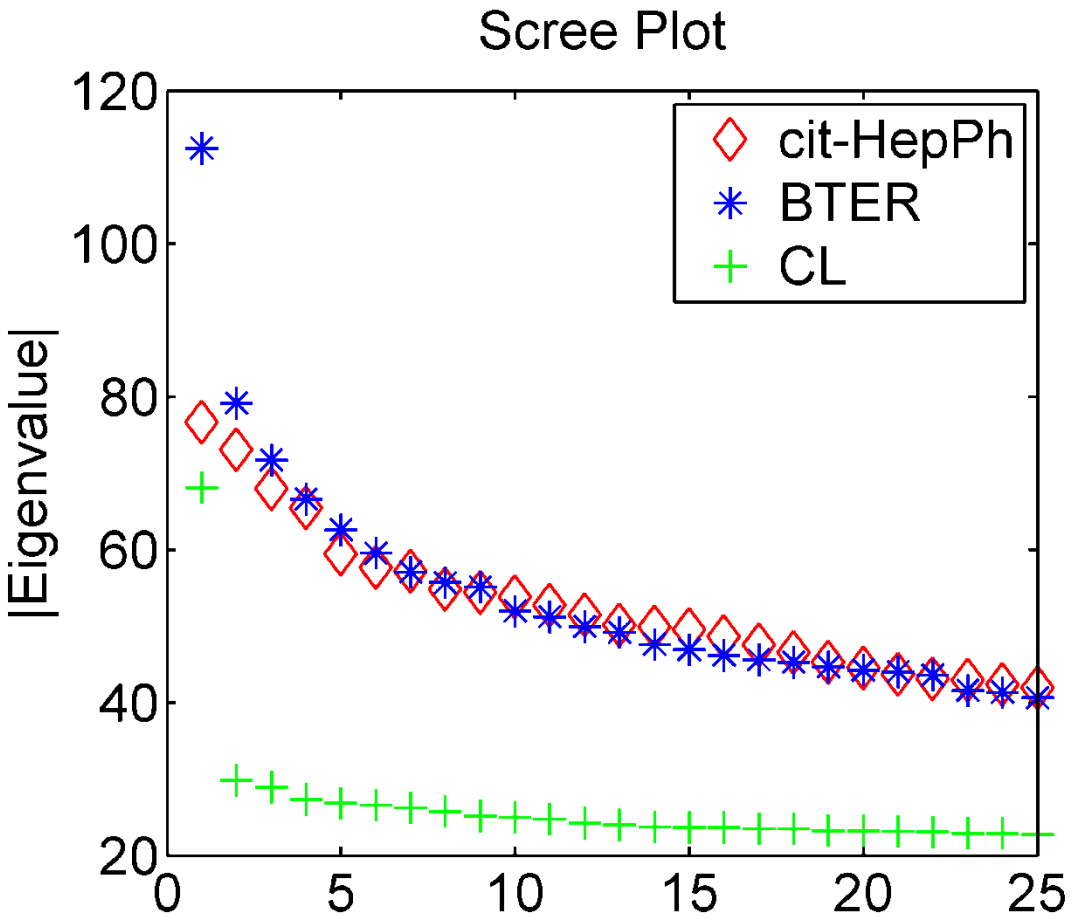
$$\rho = 0.7\left(1 - 0.6\,\frac{\log(d-1)}{\log(d_{\max}-1)}\right)^3$$

| Graph | GCC |
|---|---|
| cit-HepPh | 0.15 |
| BTER | 0.16 |
| CL | 0 |

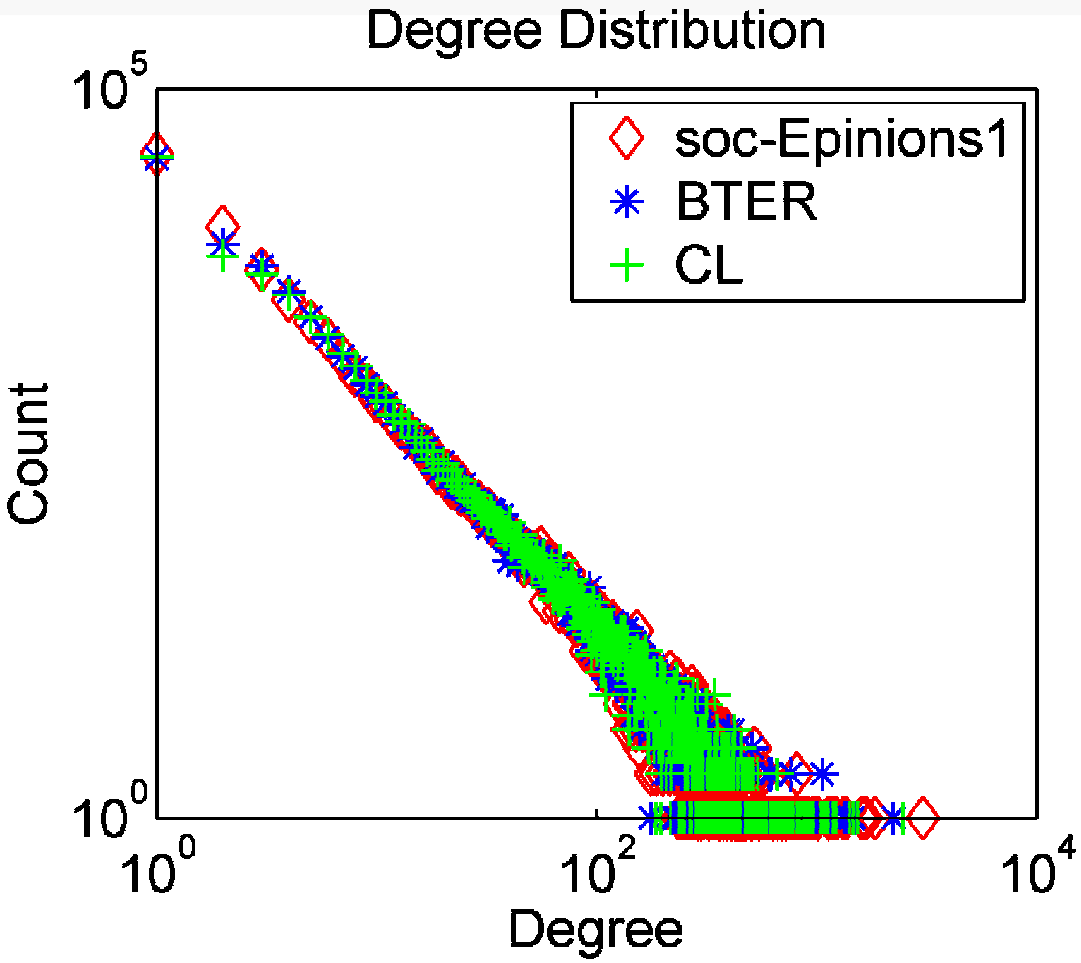


# Citation Network (cit-HepPh)



# Trust Network (soc-Epinions1)



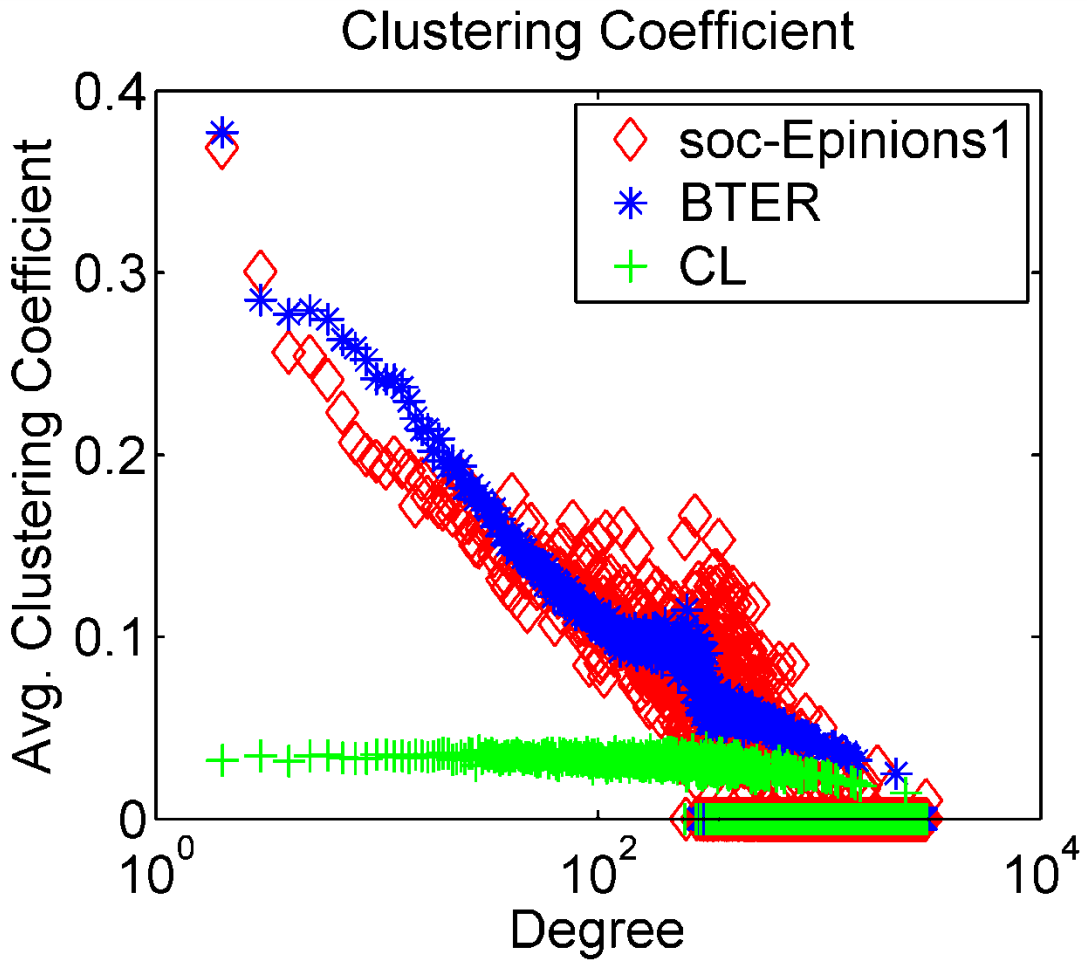

| Graph | Nodes | Edges | LCC % |
|---|---|---|---|
| Epinions1 | 75,879 | 811,480 | 100 |
| BTER | 72,425 | 812,724 | 96 |
| CL | 71,223 | 812,190 | 98 |

# Trust Network (soc-Epinions1)

BTER
Phase 1 Edges: 300,162
Phase 2 Edges: 515,192

$$\rho = 0.7\left(1 - 1.25\,\frac{\log(d-1)}{\log(d_{\max}-1)}\right)^2$$

| Graph | GCC |
|---|---|
| soc-Epinions1 | 0.07 |
| BTER | 0.07 |
| CL | 0.03 |



# Trust Network (soc-Epinions1)