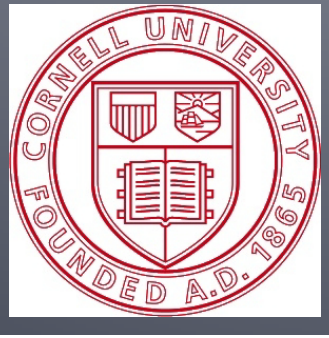
Modeling Large Social & Information Networks

Jure Leskovec (jure@cs.stanford.edu)
Computer Science Department
Cornell University / Stanford University

Tutorial at ICML 2009



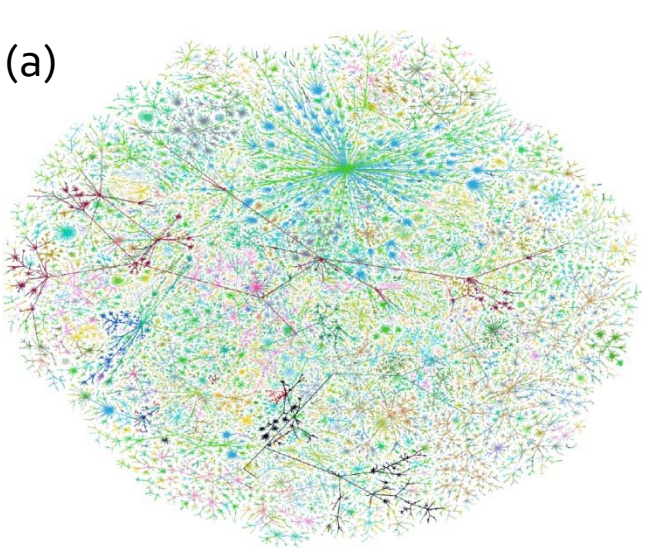
About the tutorial

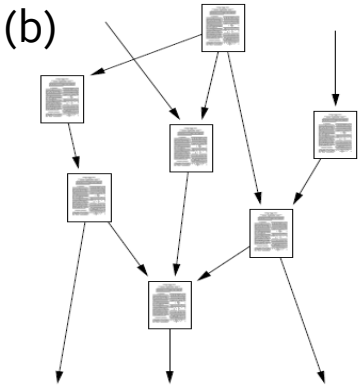
- Introduce properties, models and tools for modeling and analysis of large real-world networks

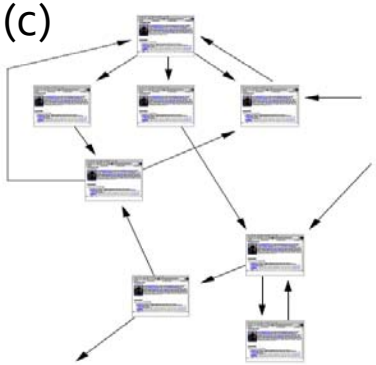
- **Goal:** find **patterns, rules, clusters, outliers, …**
  - in large static and evolving graphs

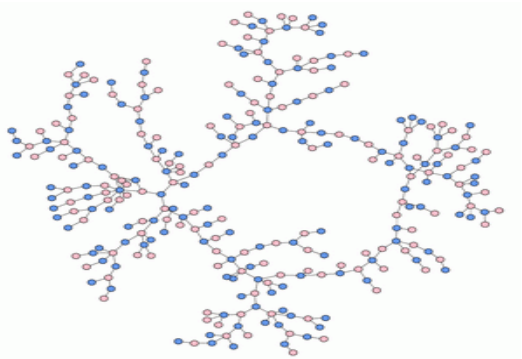
- Acknowledgements: Jon Kleinberg, Christos Faloutsos, Ravi Kumar, Andrew Tomkins, Lars Backstrom, Michael Mahoney, Anirban Dasgupta, Kevin Lang, Zoubin Ghahramani, Lise Getoor, Deepay Chakrabarti, Eric Horvitz

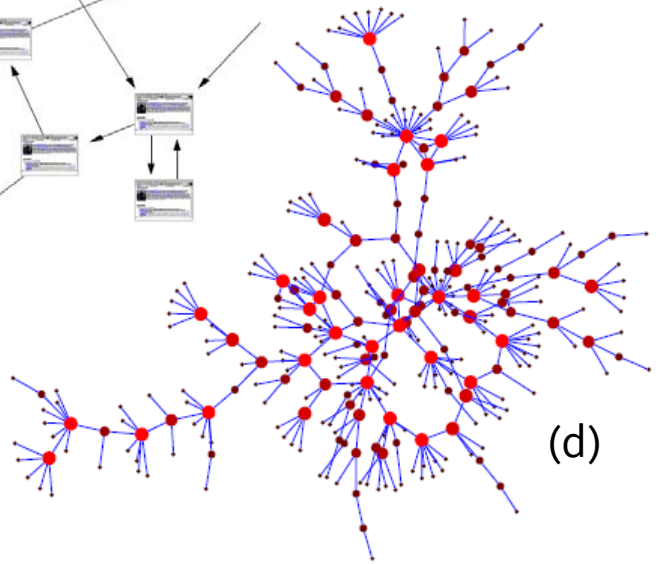
Networks: rich data











- Internet (a)
- Citation network (b)
- World Wide Web (c)

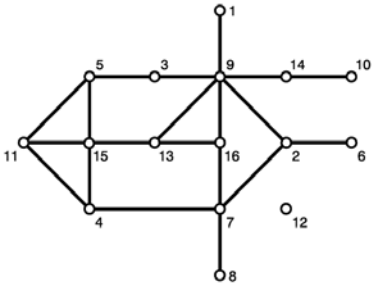
- Sexual network (d)
- Dating network (e)

Why networks?

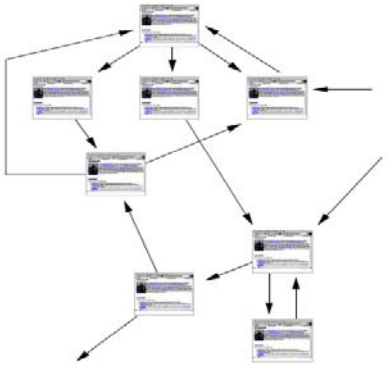
- Network data is increasingly available:
  - Large on-line **computing** applications where data can naturally be represented as a **network**:
    - On-line communities: Facebook (120 million users)
    - Communication: Instant Messenger (~1 billion users)
    - News and Social media: Blogging (250 million users)
  - Also in systems biology, health, medicine, …
- Network is a set of weakly interacting entities
- Links give added value:
  - Google realized web-pages are connected
  - Collective classification

Networks of the Real-world (1)

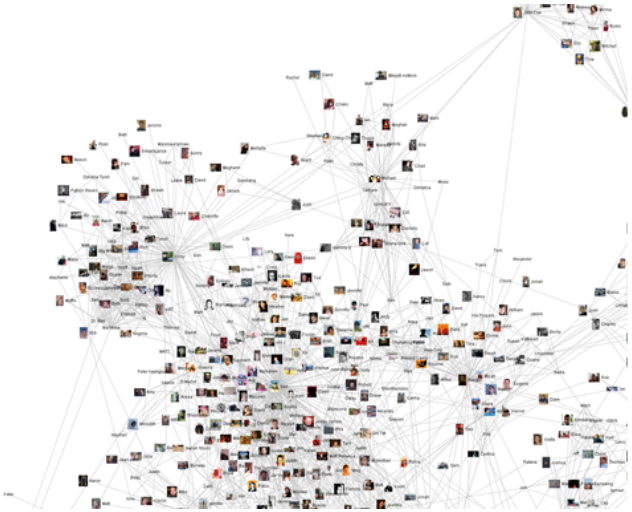
- Information networks:
  - World Wide Web: hyperlinks
  - Citation networks
  - Blog networks
- Social networks:
  - Organizational networks
  - Communication networks
  - Collaboration networks
  - Sexual networks
  - Collaboration networks
- Technological networks:
  - Power grid
  - Airline, road, river networks
  - Telephone networks
  - Internet
  - Autonomous systems



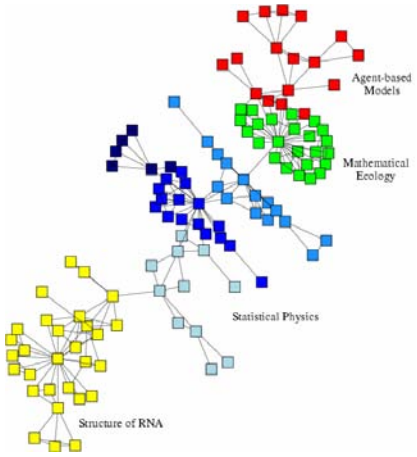
Florentine families



Web graph



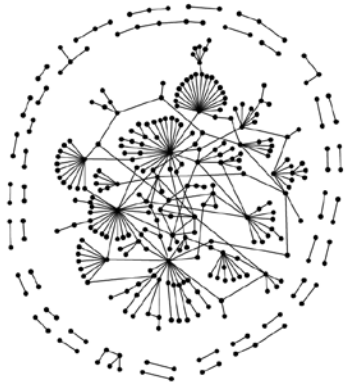
Friendship network



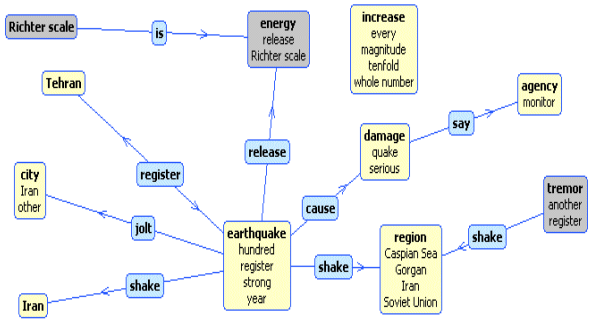
Collaboration network

Networks of the Real-world (2)

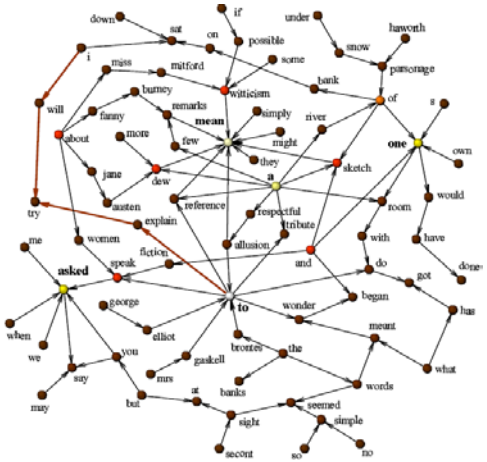
- Biological networks:
  - metabolic networks
  - food webs
  - neural networks
  - gene regulatory networks
- Language networks:
  - Semantic networks
- Software networks:
  - Call graphs
- …



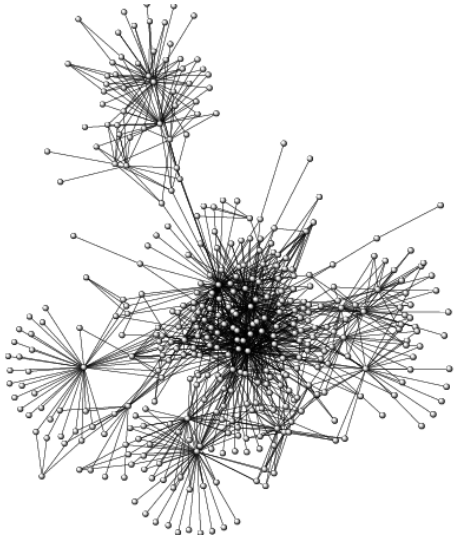
Yeast protein interactions



Semantic network



Language network



Software network

Networks as Phenomena

The emergence of 'cyberspace' and the World Wide Web is like the discovery of a new continent.

Jim Gray, 1998 Turing Award address

- Complex networks as phenomena, not just designed artifacts
- What are the common patterns that emerge?

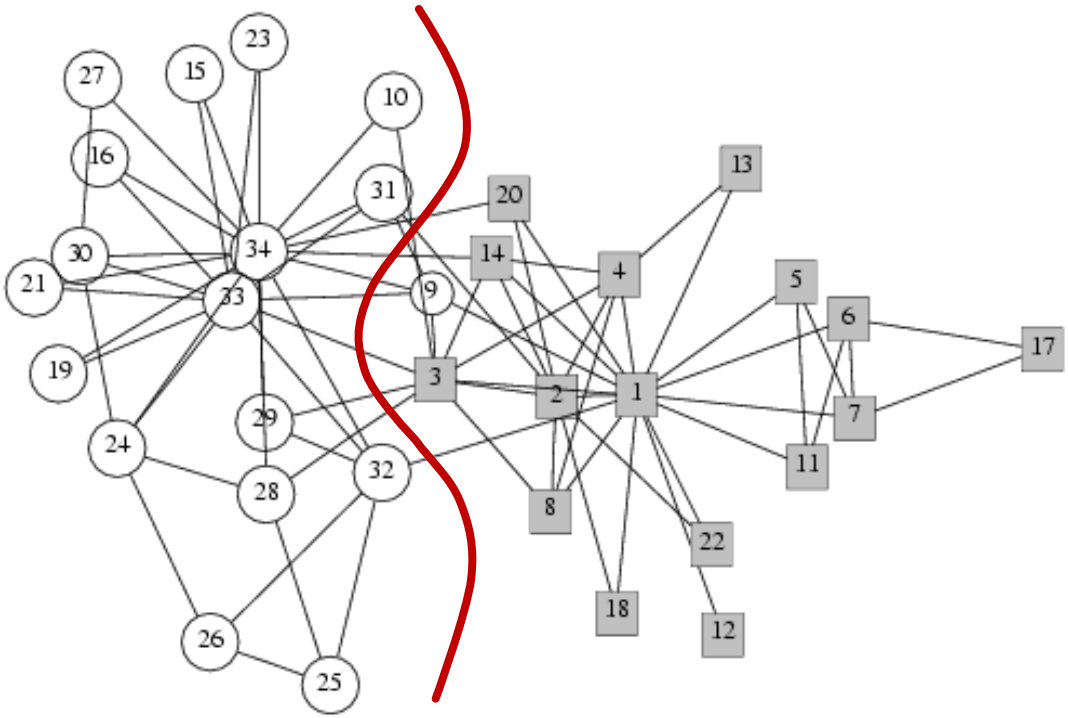
Models and Laws of Networks

We want Kepler's Laws of Motion for the Web.

Mike Steuerwalt, NSF KDI workshop, 1998

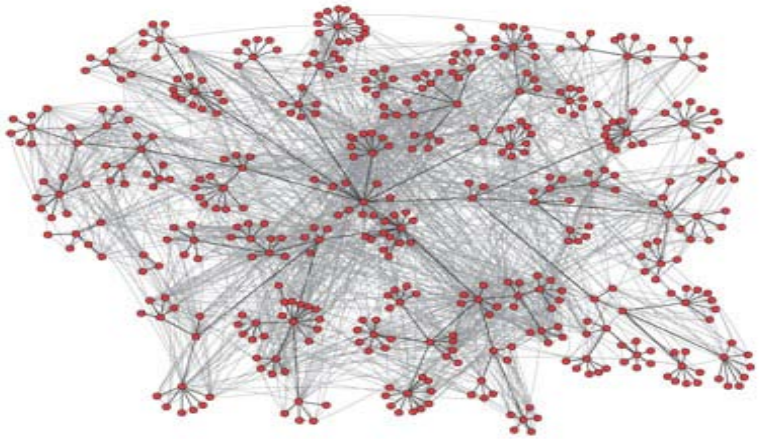
- Need statistical methods to quantify large networks
- What do we hope to achieve from models of networks?
  - Patterns and statistical properties of network data
  - Design principles and models
  - Understand why networks are organized the way they are (predict behavior of networked systems)

Mining Social Network Data



- Mining social networks has a long history in social sciences:
  - Wayne Zachary's PhD work (1970-72): observe social ties and rivalries in a university karate club
  - During his observation, conflicts led the group to split
  - Split could be explained by a minimum cut in the social network

Networks: Rich Social Data



- Traditional obstacle:
  Can only choose 2 of 3:
  - Large-scale
  - Realistic
  - Completely mapped
- **Now:** large on-line systems leave detailed records of social activity
  - On-line communities: MyScace, Facebook, LiveJournal
  - Email, blogging, electronic markets, instant messaging
  - On-line publications repositories, arXiv, MedLine

Networks: A Matter of Scale

- Network data spans many orders of magnitude:
  - 436-node network of email exchange over 3-months at corporate research lab [Adamic-Adar, SocNets '03]
  - 43,553-node network of email exchange over 2 years at a large university [Kossinets-Watts, Science '06]
  - 4.4-million-node network of declared friendships on a blogging community [Liben-Nowell et al., PNAS '05, Backstrom et at., KDD '06]
  - 240-million-node network of all IM communication over a month on Microsoft Instant Messenger [Leskovec-Horvitz, WWW '08]

Networks: Scale Matters

- How does massive network data compare to small-scale studies?
- Massive network datasets give us more and less:
  - More: can observe global phenomena that are genuine, but literally invisible at smaller scales
  - Less: don't really know what any node or link means. Easy to measure things, hard to pose right questions
  - Goal: Find the point where the lines of research converge

Networks: Structure & Process

- What have we learned about large networks?
- Structure: Many recurring patterns
  - Scale-free, small-world, locally clustered, bow-tie, hubs and authorities, communities, bipartite cores, network motifs, highly optimized tolerance

Processes and dynamics:

Information propagation, cascades, epidemic thresholds, viral marketing, virus propagation, diffusion of innovation

Not in today's tutorial

Tutorial outline

- **Structure and models for networks**
  - What are properties of real graphs?
- **How to model them?**

- Part 1: Modeling global network structure
  - "Mechanistic" approaches (math, physics)
- Part 2: Modeling local network structure (links):
  - Statistical/ML approaches
- Part 3: Modeling network structure at the level of groups of nodes
  - Graph partitioning/clustering

Modeling global network structure

- Erdos-Renyi random graphs
- Preferential attachment
- Small-world model
- Power-law degree distributions
- Local clustering
- Six degrees of separation
- Kronecker graphs

Motivation

- How do large network “look like”?
  - Empirical: statistical tools to quantify structure networks
  - Models: mechanisms that reproduce such properties (models also make “predictions” about other properties)

- 3 parts/goals:
  - Large scale statistical properties of large networks
  - Models that help understand these properties
  - Predict behavior of networked systems based on measured structural properties and local rules governing individual nodes

Random graph model

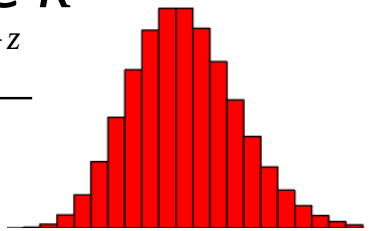
- What is the simplest way to generate a graph?
- Erdos-Renyi Random Graph model [Erdos-Renyi, '60]
  - aka.: Poisson/Bernoulli random graphs
- Two variants:
  - $G_{n,p}$: graph on n nodes and each edge (u,v) appears i.i.d. with prob. p. So a graph with m edges appears with prob. $p^m(1-p)^{M-m}$, where $M=n(n-1)/2$ is the max number of edges
  - $G_{n,m}$: graphs with n nodes, m uniformly at random picked edges

What kinds of networks does such process produce?

Properties of random graphs

- **Degree distribution** is Binomial (Poisson in the limit).
  Let p_k denote a fraction of nodes with degree k

$$p_k = \binom{n}{k} p^k (1-p)^{n-k} \approx \frac{z^k e^{-z}}{k!}$$

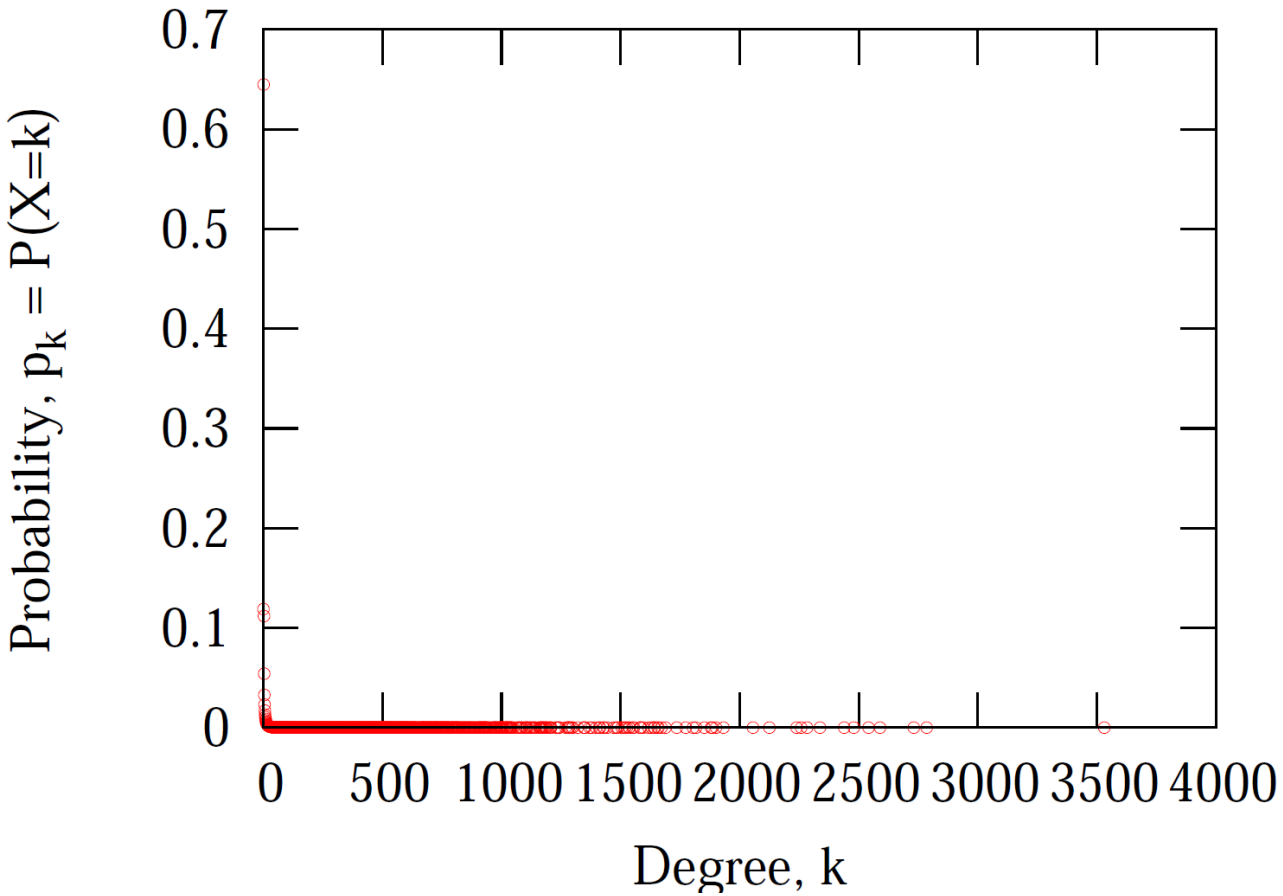


- **Diameter** is $O(\log n)$:
  - G has expansion α if $\forall S \subseteq V$: $\#edges\ leaving\ S \geq \alpha \cdot |S|$
  - Let S_j be a set of nodes within j steps of v. Then $|S_{j+1}| \geq \alpha |S_j|$. So in $O(\log n)$ steps $|S_j|$ grows to $\Theta(n)$.
- **Emergence of giant component:** avg. degree $k=2m/n$:
  - $k=1-\varepsilon$: all components are of size $\Omega(\log n)$
  - $k=1+\varepsilon$: 1 component of size $\Omega(n)$, others have size $\Omega(\log n)$

[Leskovec et al. KDD '08]

Degrees in real networks (1)

- Take real network plot a histogram of p_k vs. k

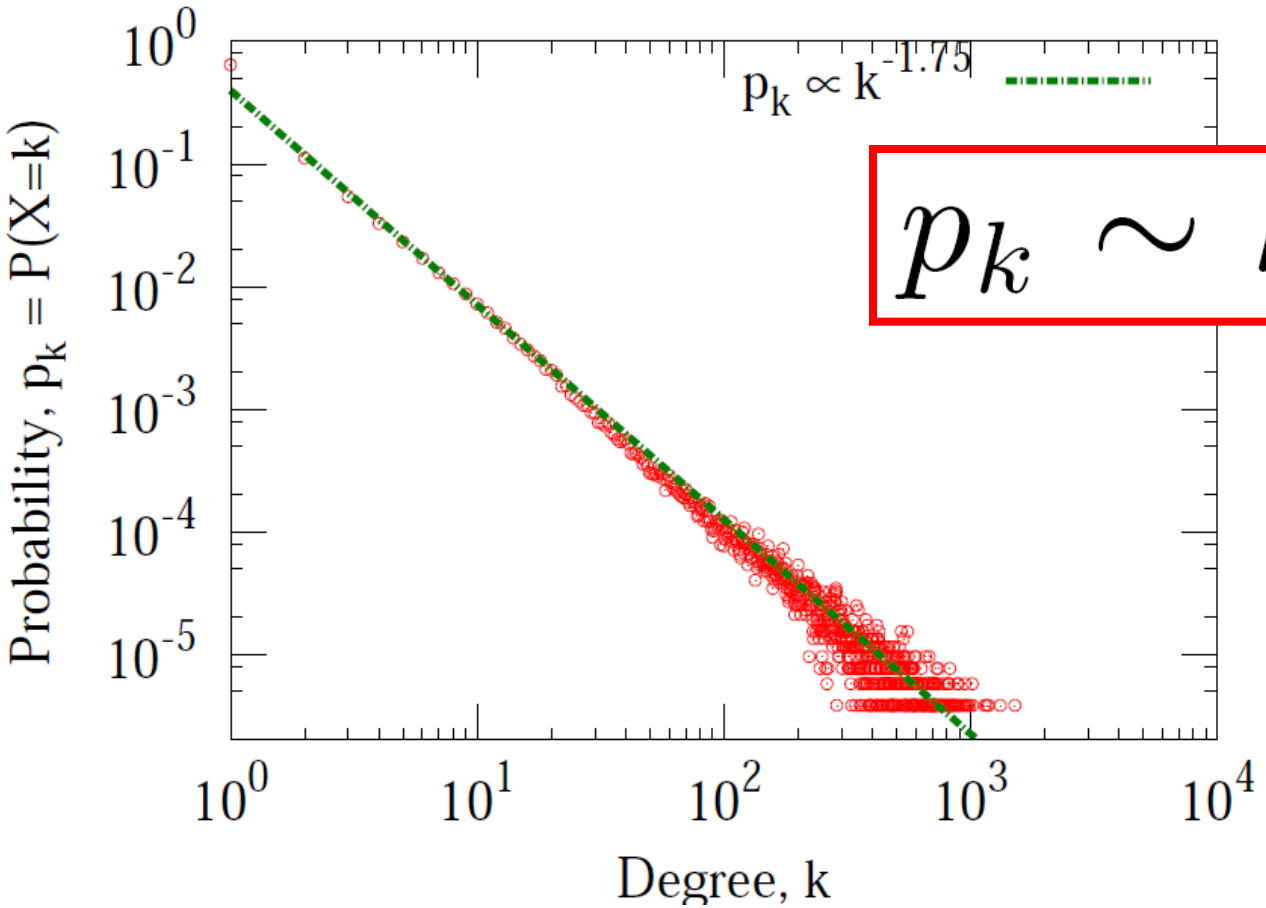


Flickr social network
n= 584,207,
m=3,555,115

[Leskovec et al. KDD '08]

Degrees in real networks (2)

- Plot the same data on *log-log* axis:



$$p_k \sim k^{-\alpha}$$

Flickr social network
n= 584,207,
m=3,555,115

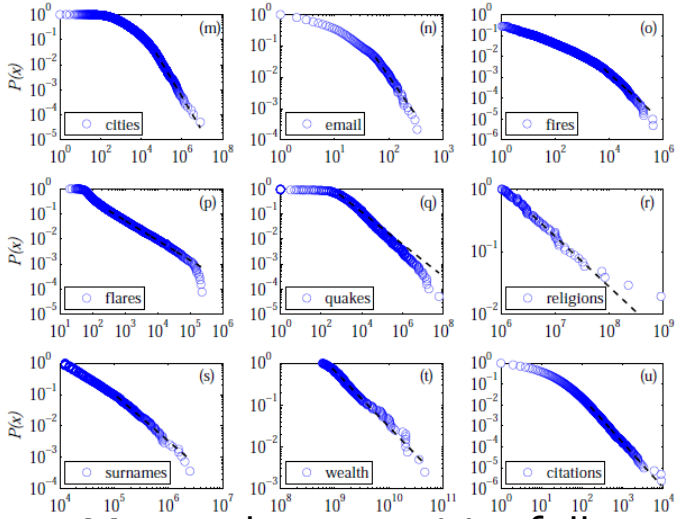
Degrees in real networks (2)

- Degrees are heavily skewed:

  Distribution is heavy tailed: $\lim_{x\to\infty} \frac{Pr(X > x)}{e^{-\epsilon x}} = \infty$

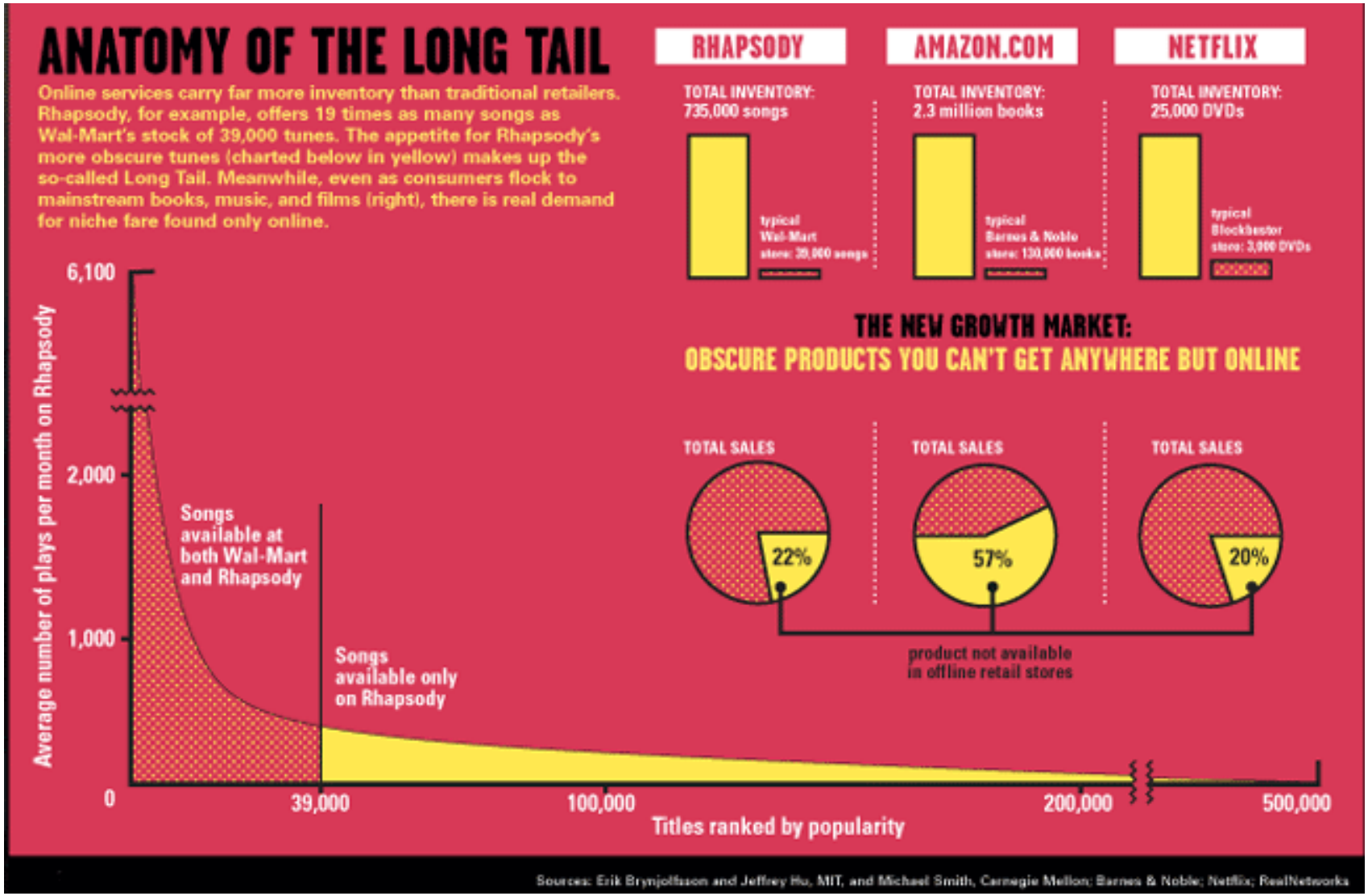
- Various names, kinds and forms:
  - Long tail, Heavy tail, Zipf's law, Pareto's law

power law	$x^{-\alpha}$
power law with cutoff	$x^{-\alpha}\mathrm{e}^{-\lambda x}$
stretched exponential	$x^{\beta-1}\mathrm{e}^{-\lambda x^{\beta}}$
log-normal	$\frac{1}{x}\exp\left[-\frac{(\ln x-\mu)^2}{2\sigma^2}\right]$



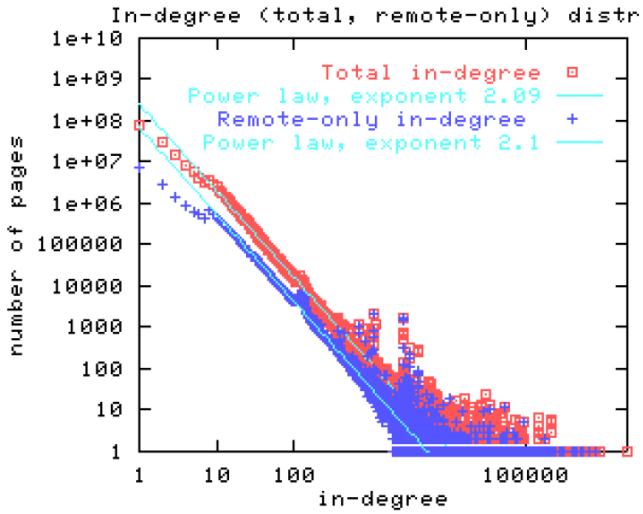
Many other quantities follow heavy-tailed distributions

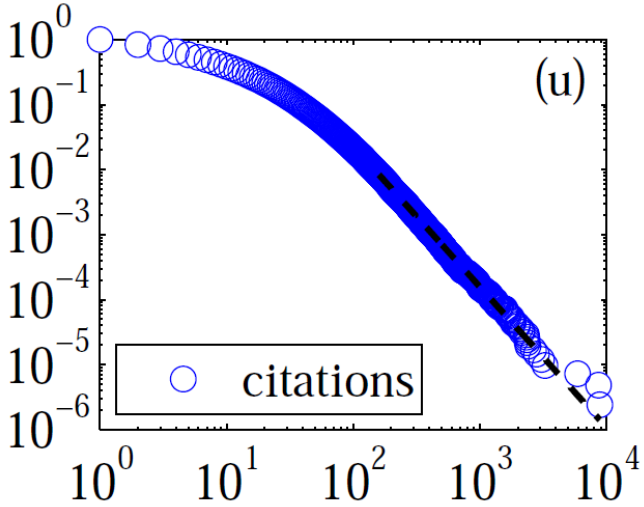
The long tail



Power law degree exponents

- Power law degree exponent is typically $2 < \alpha < 3$
  - Web graph [Broder et al. 00]:
    - $\alpha_{in} = 2.1$, $\alpha_{out} = 2.4$
  - Autonomous systems [Faloutsos et al. 99]:
    - $\alpha = 2.4$
  - Actor collaborations [Barabasi-Albert 00]:
    - $\alpha = 2.3$
  - Citations to papers [Redner 98]:
    - $\alpha \approx 3$
  - Online social networks [Leskovec et al. 07]:
    - $\alpha \approx 2$





Estimating α

$$p_k \sim k^{-\alpha}$$

- Tails are heavy:
  - If $\alpha \leq 2 : \mathrm{E}[x] = \infty$
  - If $\alpha \leq 3 : \mathrm{Var}[x] = \infty$

$$E[x] = \int_{x_{\min}}^{\infty} x p(x)\,\mathrm{d}x = C \int_{x_{\min}}^{\infty} x^{-\alpha+1}\,\mathrm{d}x$$
$$= \frac{C}{2-\alpha}\left[x^{-\alpha+2}\right]_{x_{\min}}^{\infty} = \frac{\alpha-1}{\alpha-2} x_{\min}$$

- Estimating power-law exponent α from data:

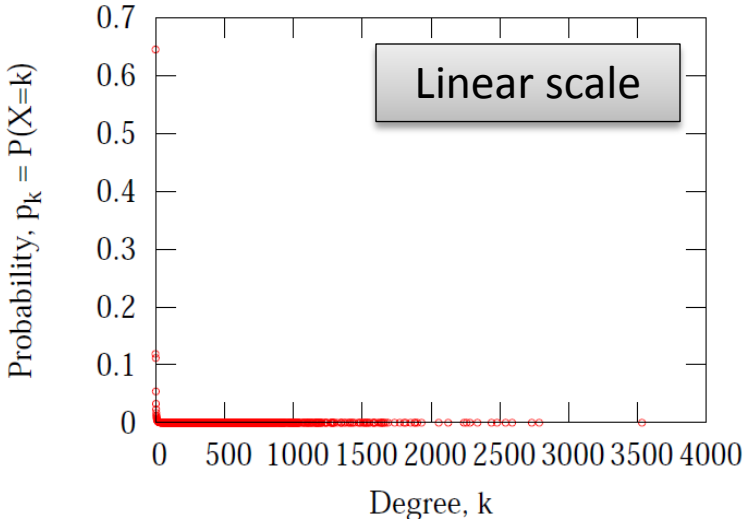
BAD! 1. Fit a line on log-log axis using least squares

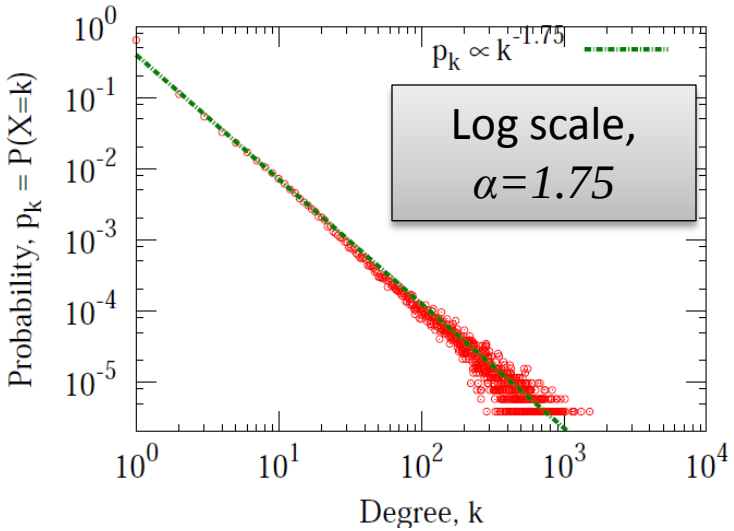
Ok 2. Plot Complementary CDF $P(X>x)$, then $\alpha = 1 + \alpha^*$ where α^* is the slope of $P(X>x)$. E.i., if $P(X=x)\ x^{-\alpha}$ then $P(X>x) = x^{-(\alpha-1)}$

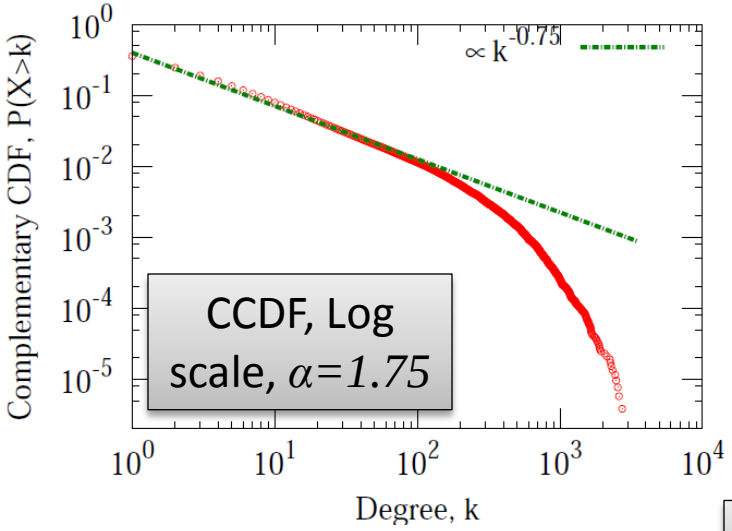
Ok 3. Use MLE: $1 + n\left[\sum_{i=1}^{n} \ln \frac{x_i}{x_{min}}\right]^{-1}$ x_i is degree of node i

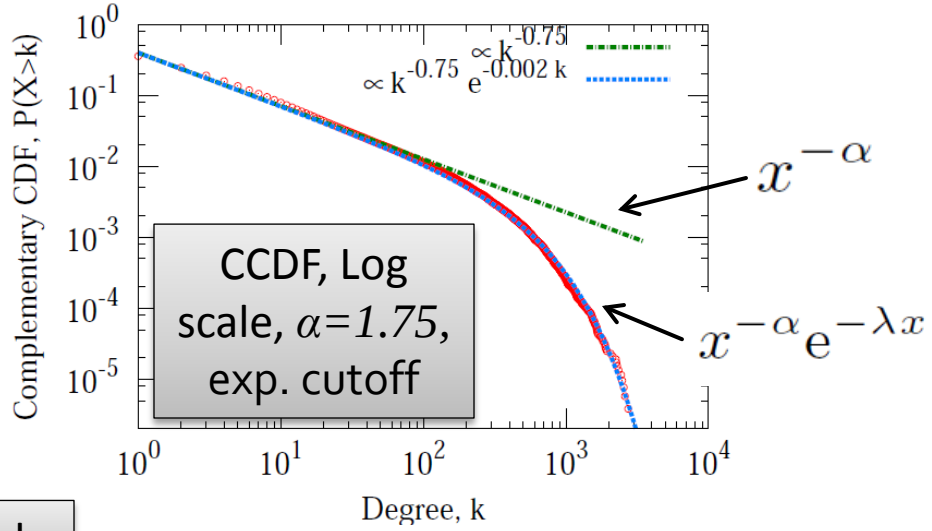
For further details see [Clauset-Shalizi-Newman 2007]

Flickr: Power-law degree exponent



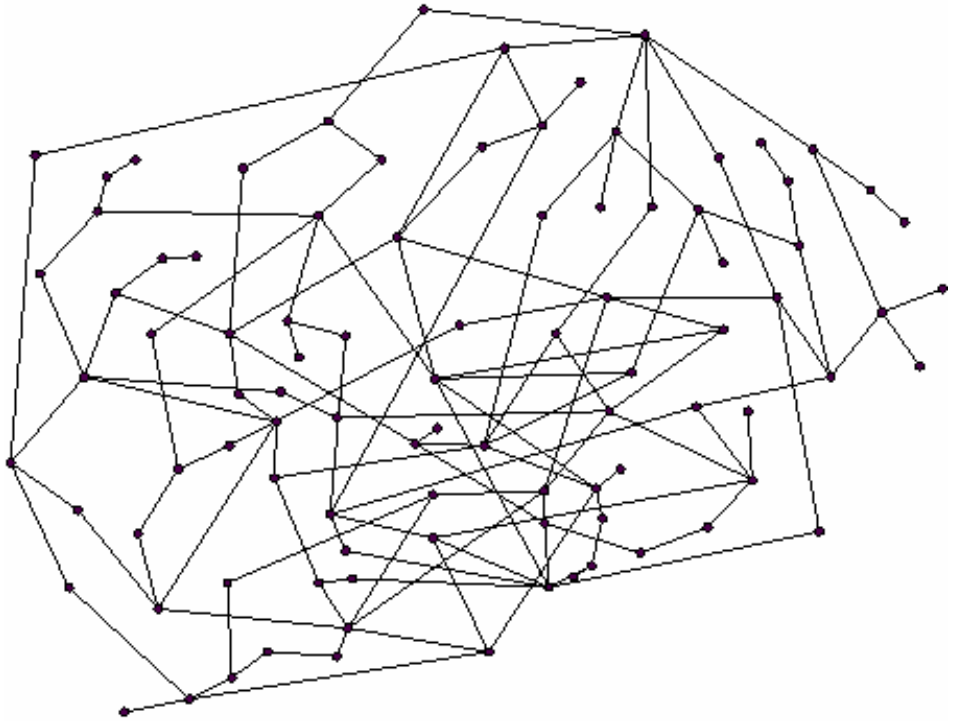






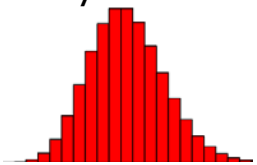
Flickr

Poisson vs. Scale-free network

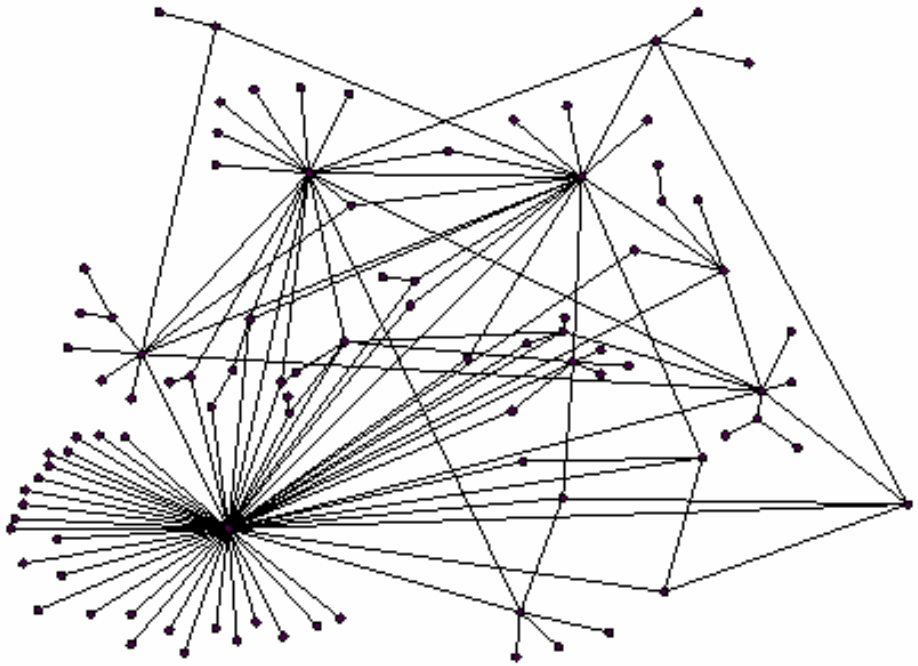


Random network

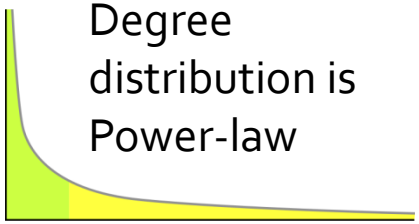
(Erdos-Renyi random graph)



Degree distribution is Binomial



Scale-free (power-law) network



Degree distribution is Power-law

Function is scale free if:

$f(ax) = c\, f(x)$

Model: Preferential attachment

- Preferential attachment [Price 1965, Albert-Barabasi 1999]:
  - A new node creates m out-links
  - Prob. of linking to node i is proportional to its degree k_i

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}$$

- Herbert Simon's result
  - Power-laws arise from "Rich get richer" (cumulative advantage)

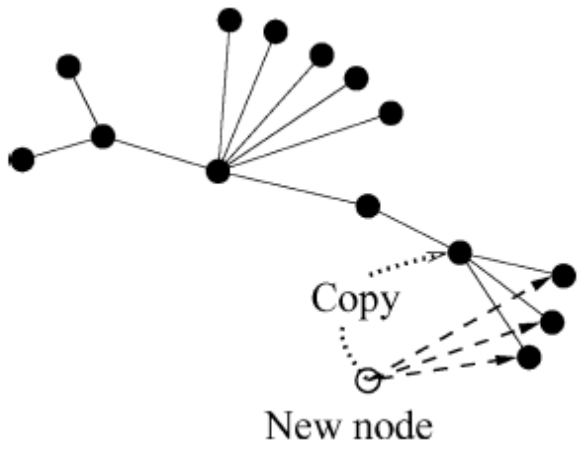
Gives graphs with power-law degree distribution: α=3

$$p_k \propto k^{-3}$$

- Examples [Price 65]:
  - Citations: new citations of a paper are proportional to the number it already has

Many extensions and variations

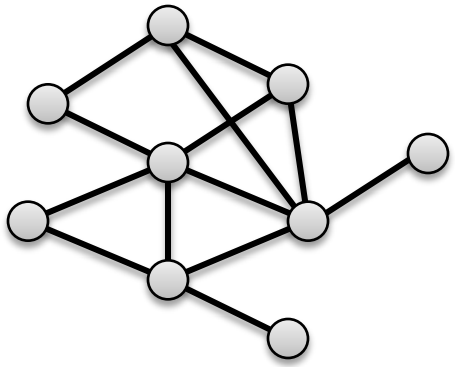
- Preferential attachment is a key ingredient
- Extensions:
  - Early nodes have advantage: node fitness
  - Geometric preferential attachment
- Copying model [Kleinberg et al.]:
  - Picking a node proportional to the degree is same as picking an edge at random (pick node and then it's neighbor)



[Leskovec et al. KDD 08]

Does Preferential attachment hold?

- 4 online social networks with exact edge arrival sequence
- Directly observe mechanisms leading to global network properties



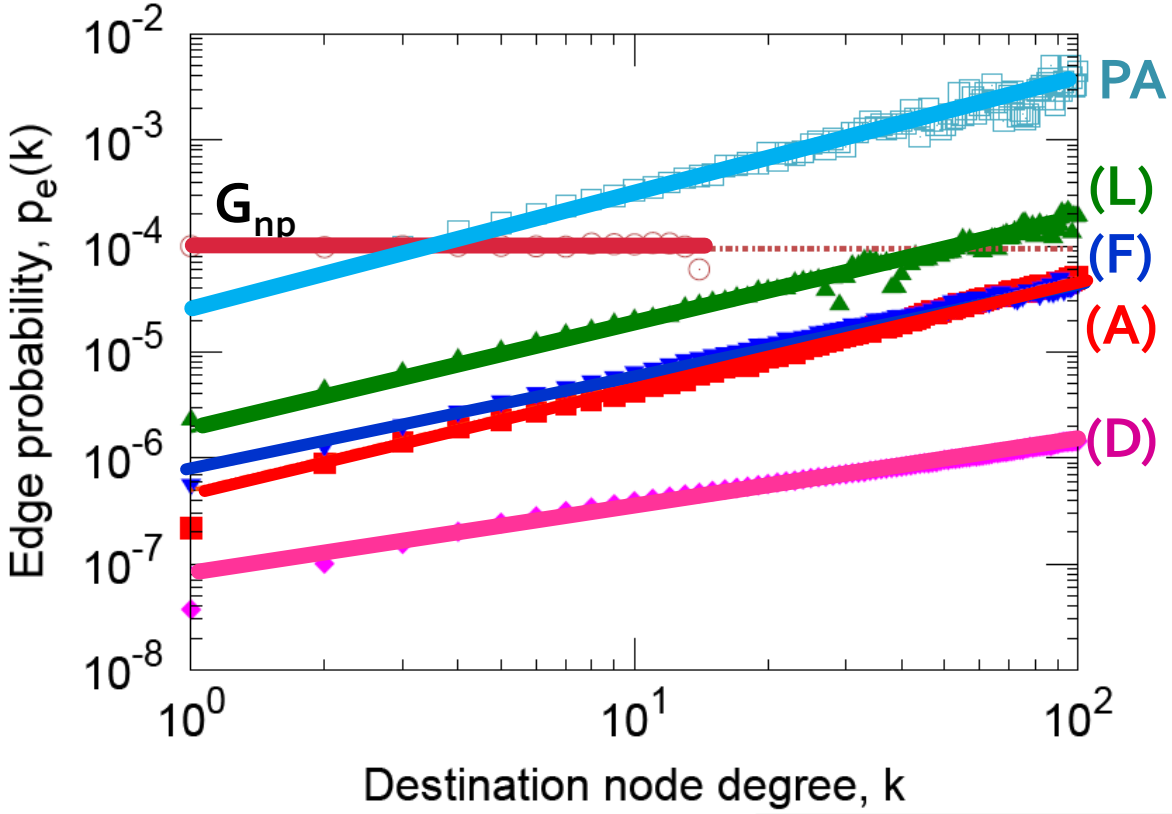
and so on for millions...

	Network	T	N	E
(F)	FLICKR (03/2003–09/2005)	621	584,207	3,554,130
(D)	DELICIOUS (05/2006–02/2007)	292	203,234	430,707
(A)	ANSWERS (03/2007–06/2007)	121	598,314	1,834,217
(L)	LINKEDIN (05/2003–10/2006)	1294	7,550,955	30,682,028

[Leskovec et al. KDD 08]

Does Preferential attachment hold?

- We unroll the true network edge arrivals
- Measure node degrees where edges attach



$$p_e(k) \propto k^{\tau}$$

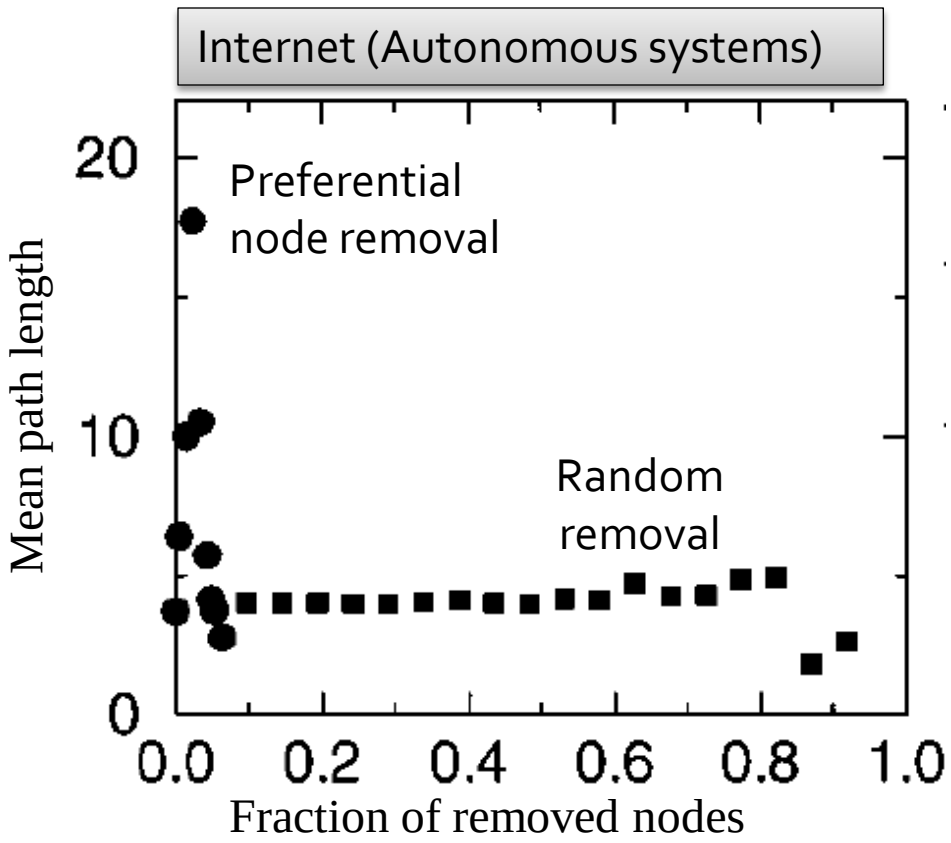
Network	τ
G_{np}	0
PA	1
F	1
D	1
A	0.9
L	0.6

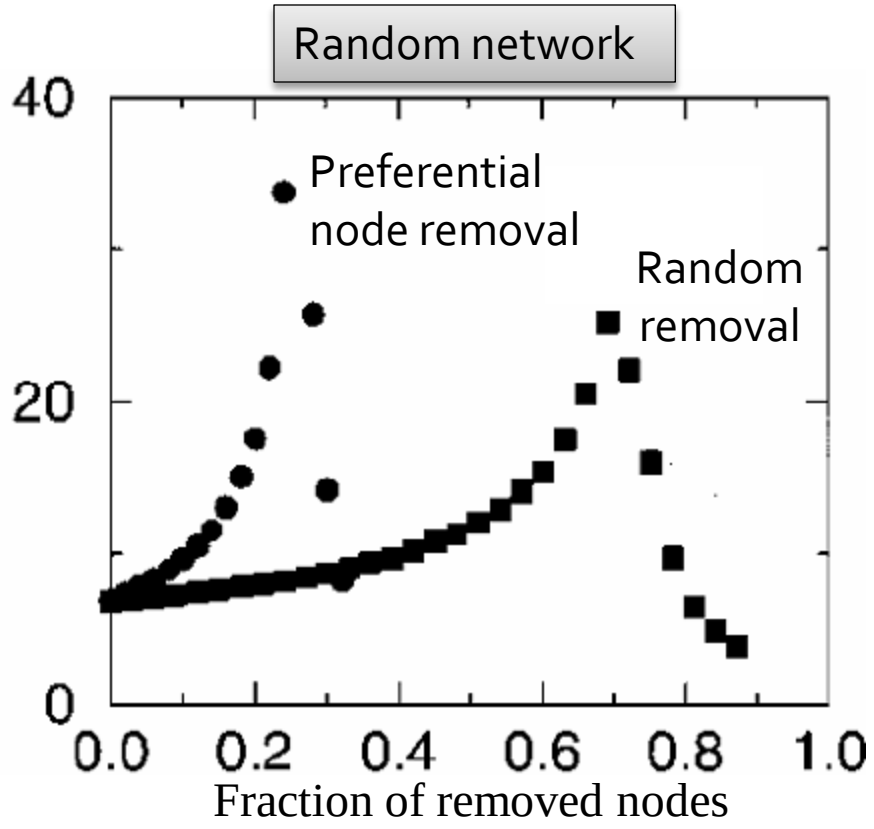
PA holds! (with a little caveat)

[Albert et al. Nature '00]

Consequence 1: network resilience

- Real-world networks are resilient to random node attacks
  - One has to remove all web-pages of degree > 5 to disconnect the web
  - But this is a very small percentage of web pages
- Random network has better resilience to targeted attacks



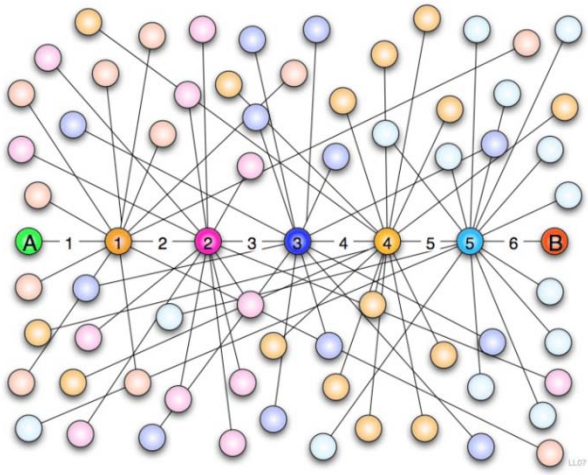


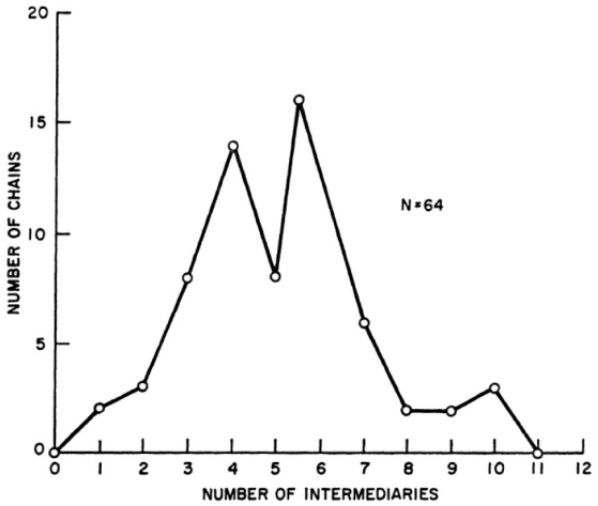
Consequence 2: Web search

- Since web graph is scale-free (and not random) outliers (high-degree webpages) are common
- Thus ranking webpages based on the link structure of the web graph works:
  - PageRank
  - Hubs and Authorities

Properties of social networks

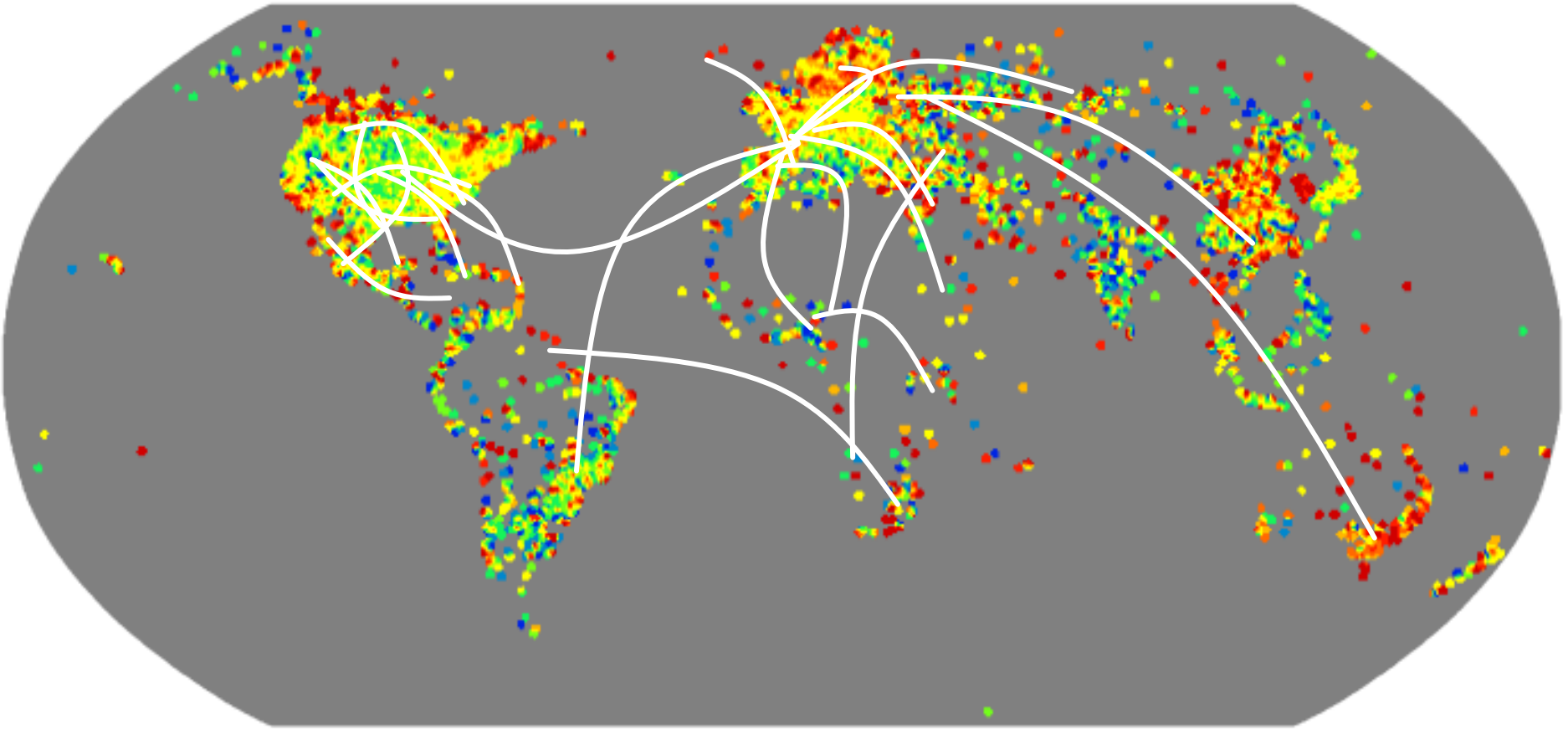
- Six degrees of separation [Milgram 60s]:
  - Random people in Nebraska were asked to send letters to stock brokers in Boston
  - Letters can only be passed to first-name acquaintances
  - On average letters reached the goal in **6 steps**





[Leskovec-Horvitz WWW '08]

World social network

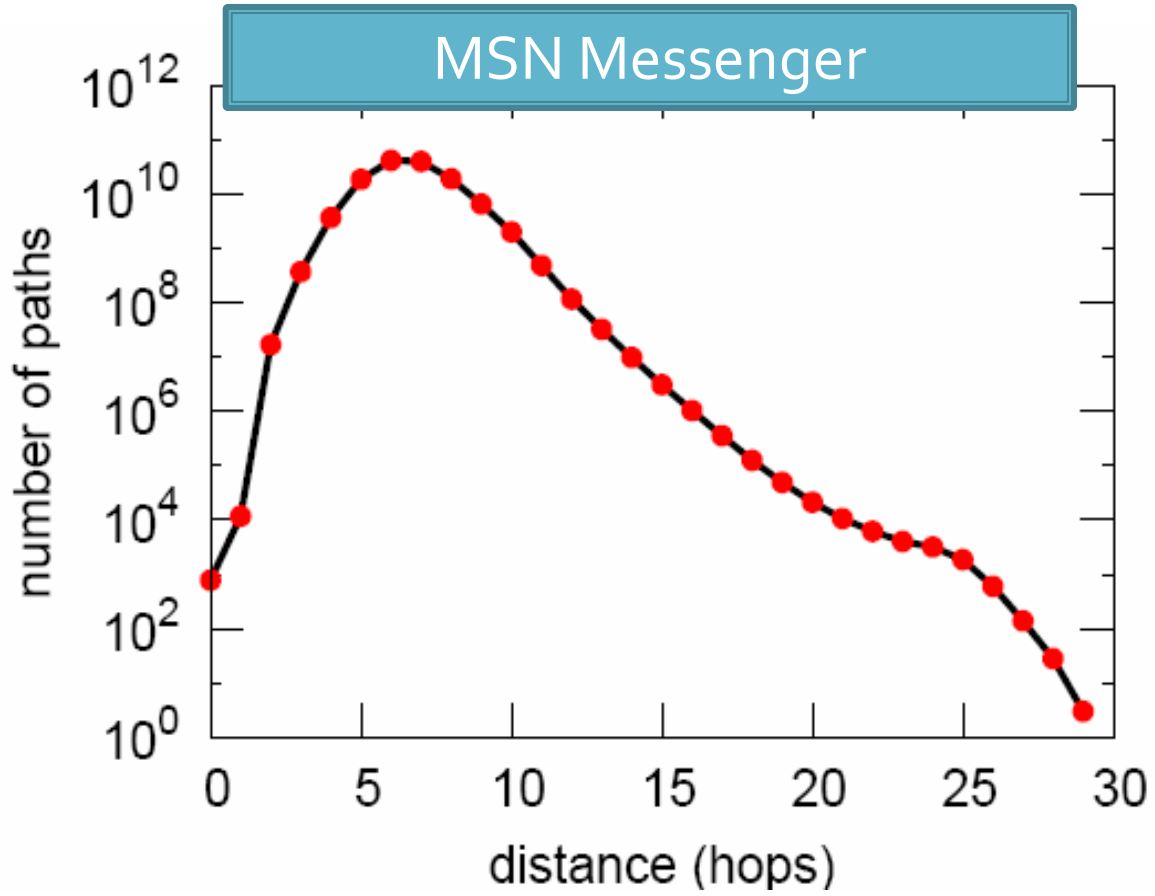


Network: who talks to whom on MSN messenger

240M nodes, 1.3 billion edges

[Leskovec-Horvitz WWW '08]

Small world of messaging

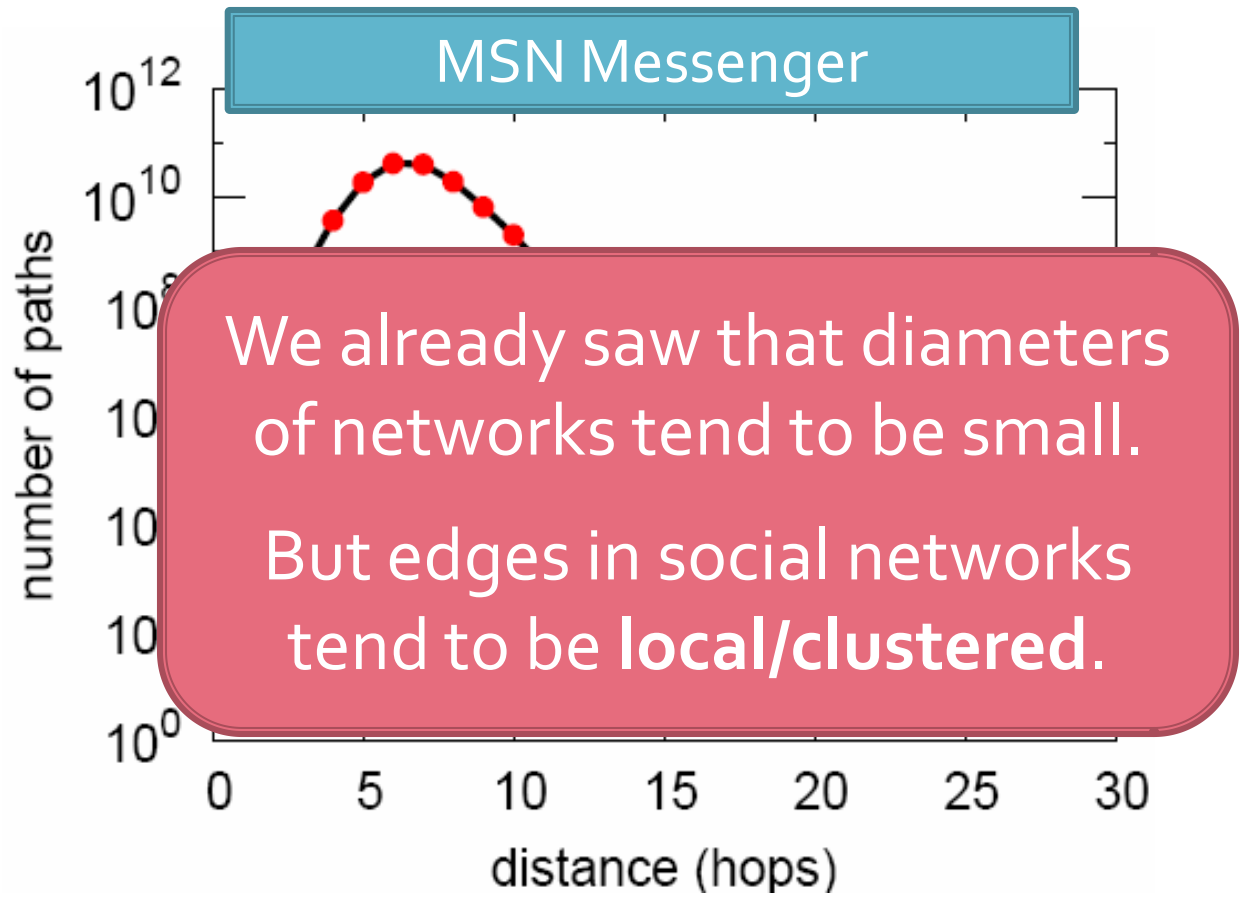


Hops	Nodes
0	1
1	10
2	78
3	3,96
4	8,648
5	3,299,252
6	28,395,849
7	79,059,497
8	52,995,778
9	10,321,008
10	1,955,007
11	518,410
12	149,945
13	44,616
14	13,740
15	4,476
16	1,542
17	536
18	167
19	71
20	29
21	16
22	10
23	3
24	2
25	3

Average path length is **6.6**
90% of nodes is reachable <8 steps

[Leskovec-Horvitz WWW '08]

Small world of messaging



We already saw that diameters of networks tend to be small.

But edges in social networks tend to be **local/clustered**.

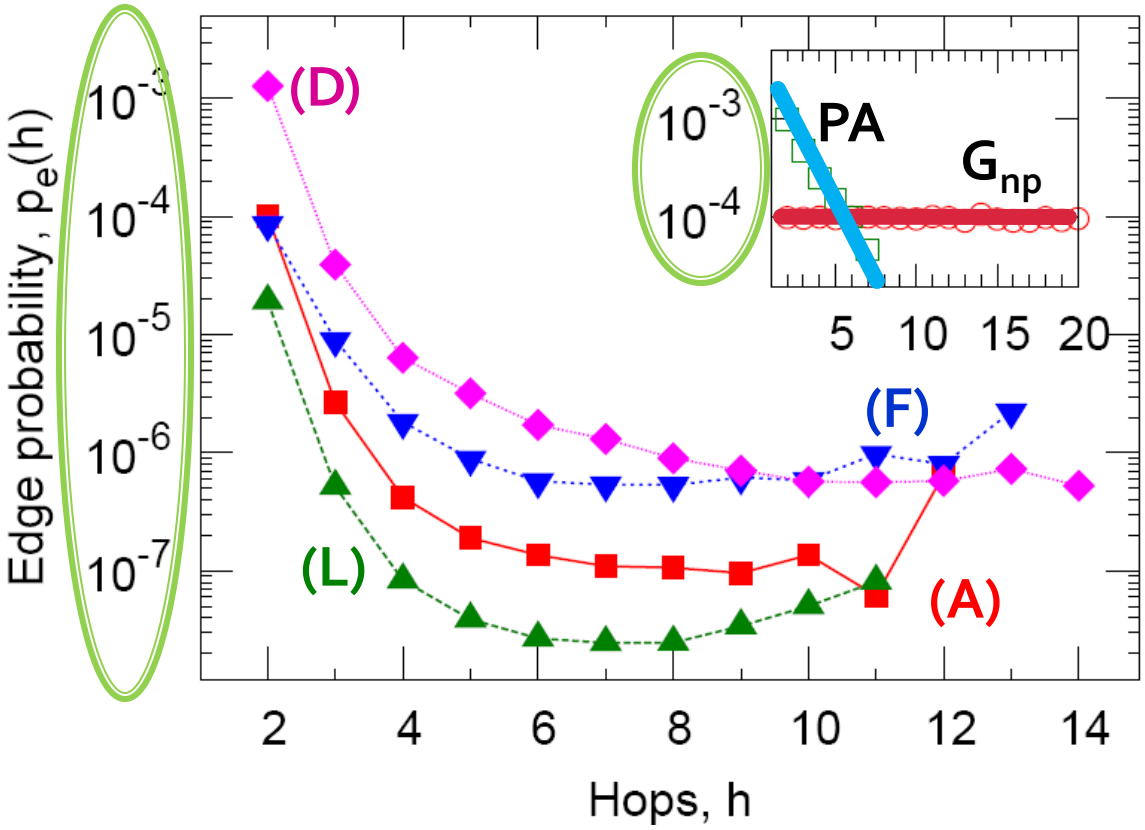
Average path length is **6.6**
90% of nodes is reachable <8 steps

Hops	Nodes
0	1
1	10
2	78
3	3,96
4	8,648
5	3,299,252
6	28,395,849
7	79,059,497
8	52,995,778
9	10,321,008
10	1,955,007
11	518,410
12	149,945
13	44,616
14	13,740
15	4,476
16	1,542
17	536
18	167
19	71
20	29
21	16
22	10
23	3
24	2
25	3

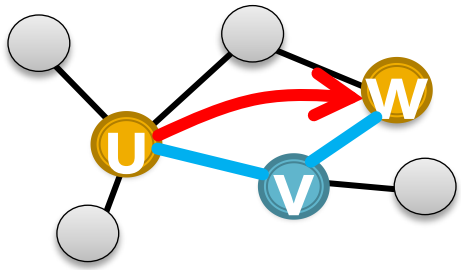
[Leskovec et al. KDD '08]

Edge locality in networks

- Just before the edge *(u,v)* is placed how many hops is between *u* and *v*?



Fraction of triad closing edges	
Network	**% Δ**
Flickr	66%
Delicious	28%
Answers	23%
LinedIn	50%

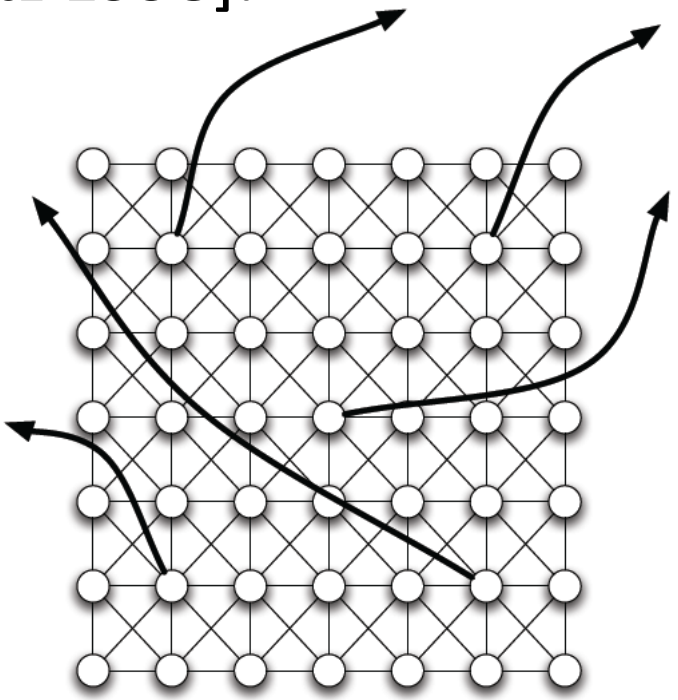


PA holds but edges are local. Most close triangles!

[Watts-Strogatz Nature '98]

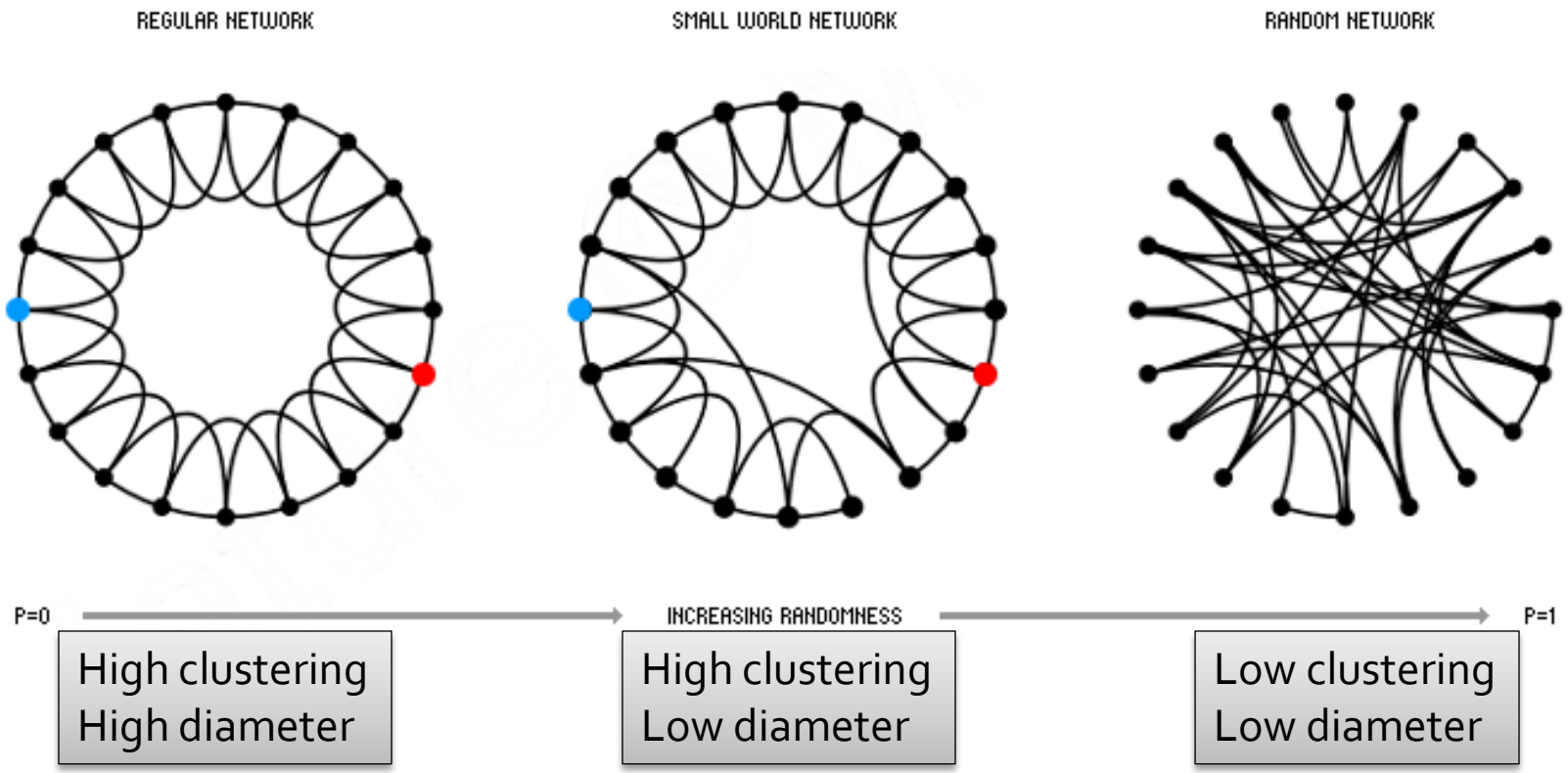
Small-world model

- How to have local edges (lots of triangles) and small diameter?
- Small-world model [Watts-Strogatz 1998]:
  - Start with a low-dimensional regular lattice
  - Rewire:
    - Add/remove edges to create shortcuts to join remote parts of the lattice
    - For each edge with prob. *p* move the other end to a random vertex



[Watts-Strogatz Nature '98]

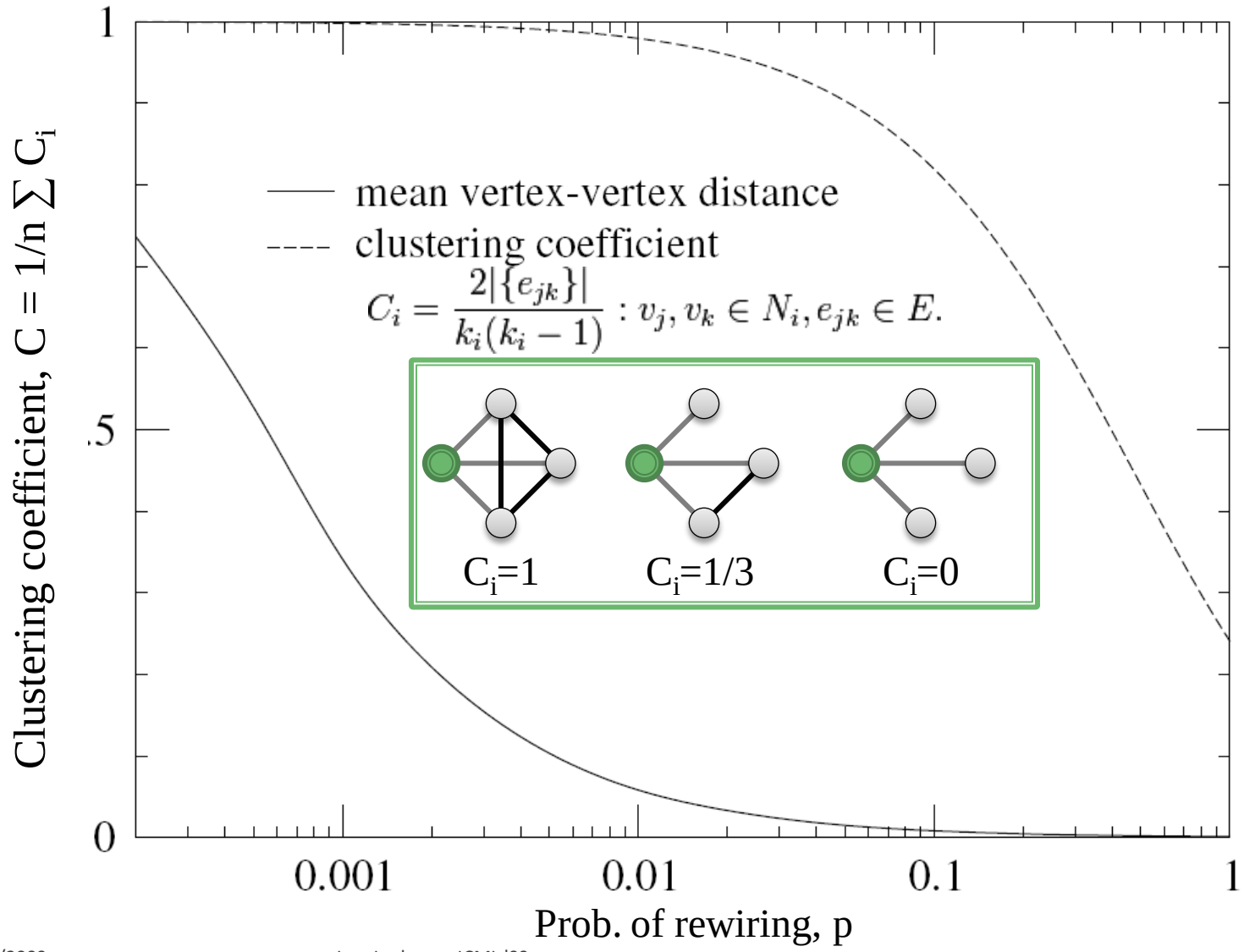
Small-world model



- Rewiring allows to interpolate between regular lattice and a random graph

Clustering and diameter in a Small-world model

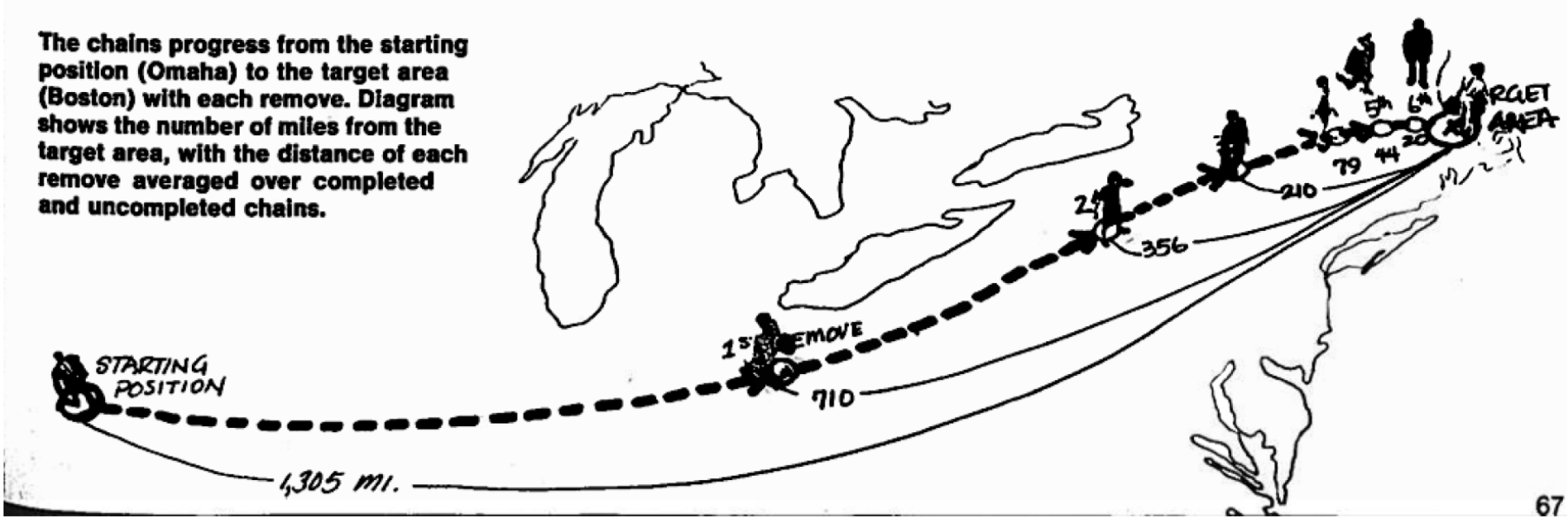
[Watts-Strogatz Nature '98]



[Milgram '67]

Consequence: Navigation (1)

- Conseqences of Milgram's experiment:
  - Short paths exist in networks
  - People are able to find them! How?

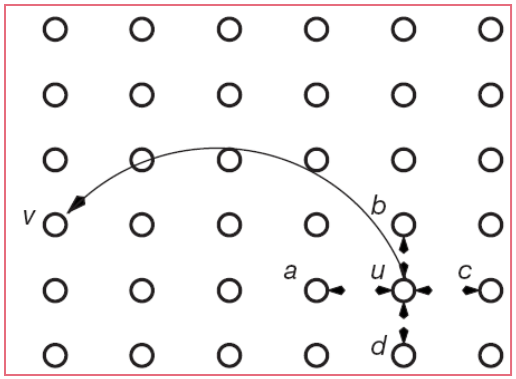


Actual path of the letter traveling from Nebraska to Boston [Milgram '67]

[Kleinberg Nature '01, Dodds et al. Science '03, LibenNovell et al. PNAS '05]

Consequence: Navigation (2)

- Networks are navigable!
  - Model: People on a grid, each node u connects to 4 neighbors and has 1 long range link to node v with prob. $d(u,v)^{-\alpha}$

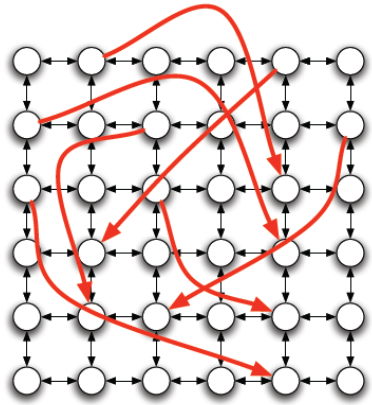


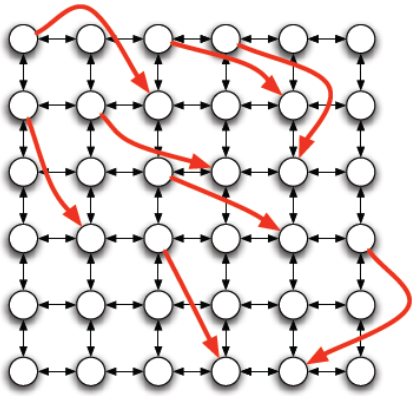
  - Greedy navigation algorithm: given (x,y) location of the target node forward the packet to the neighbor geographically closest to target

[Kleinberg Nature'01]

Consequence: Navigation (3)

- Result: If probability of a long link $d(u,v)^{-\alpha}$, then for $\alpha=2$ greedy navigation will find the target in poly-log time WHP.
- Proof idea:
  - If α too small: too many long links
  - If α too large: too many short links
- Application in P2P networks for search

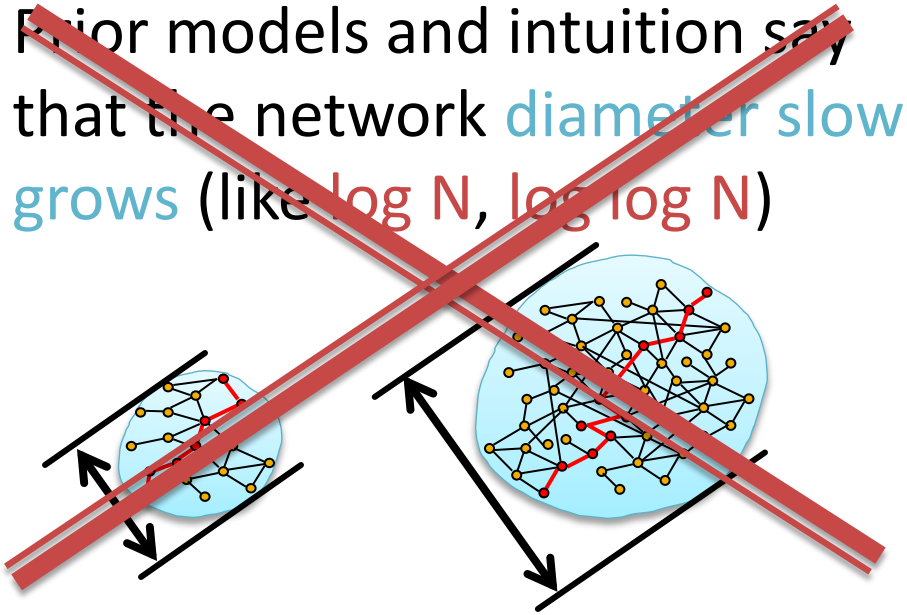




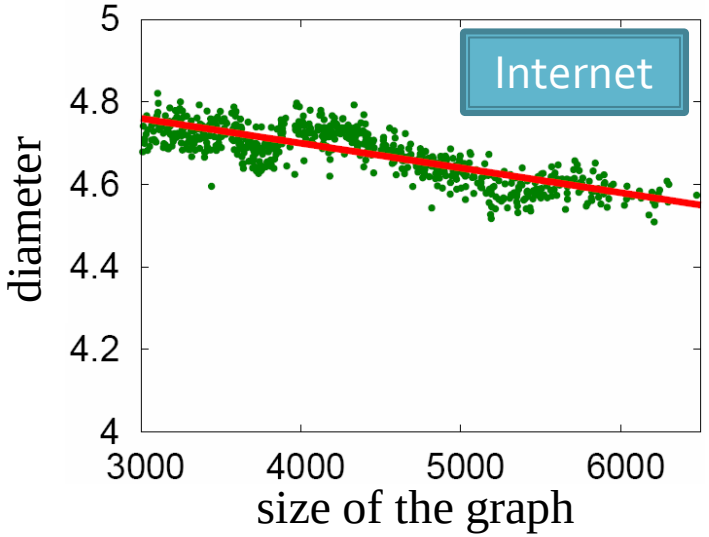
[Leskovec et al. KDD 05]

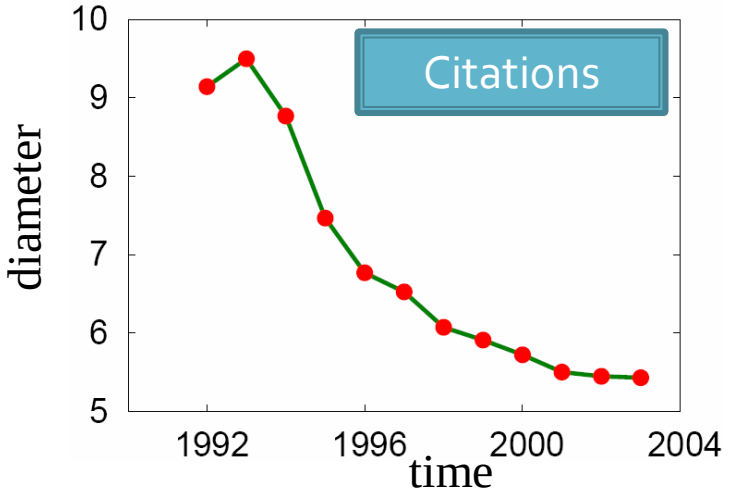
Network evolution

- Prior models and intuition say that the network diameter slowly grows (like log N, log log N)



- **Diameter shrinks over time**
  - as the network grows the distances between the nodes slowly **decrease**





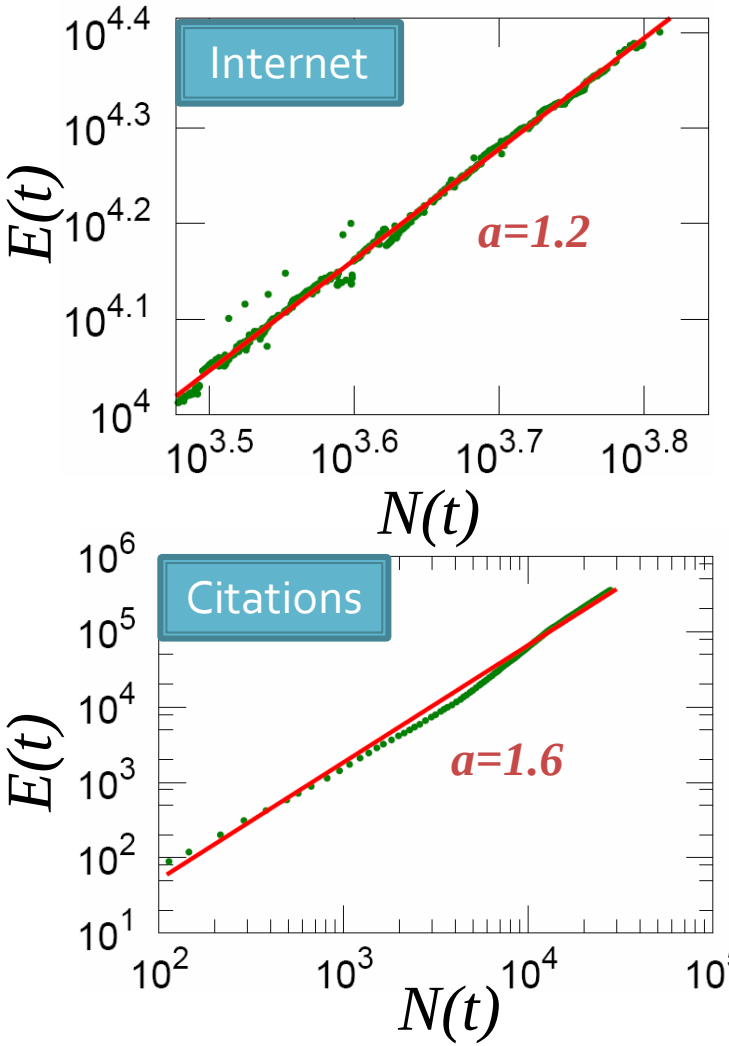
[Leskovec et al. KDD 05]

Network evolution

- What is the relation between the number of nodes and the edges over time?
- ~~Prior models assume: constant average degree over time~~
- Networks are denser over time
- **Densification Power Law:**

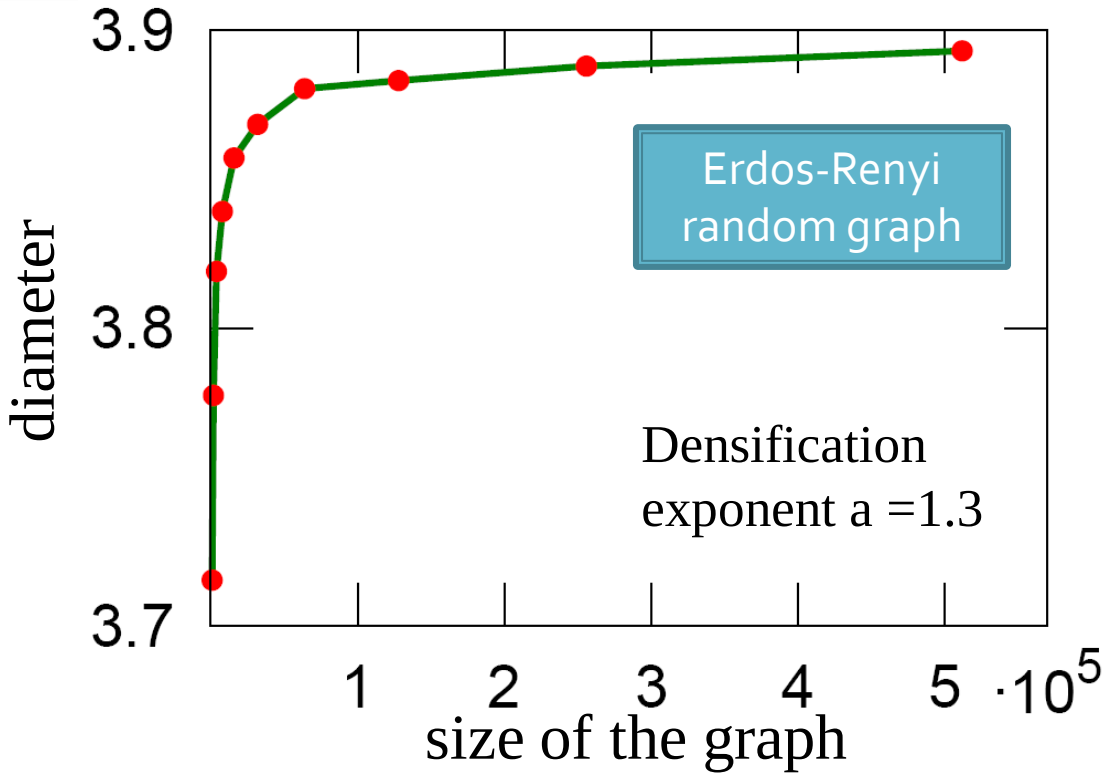
$$E(t) \propto N(t)^a$$

a … densification exponent ($1 \le a \le 2$)



Diameter of a densifying G_{np}

Is shrinking diameter just a consequence of densification?

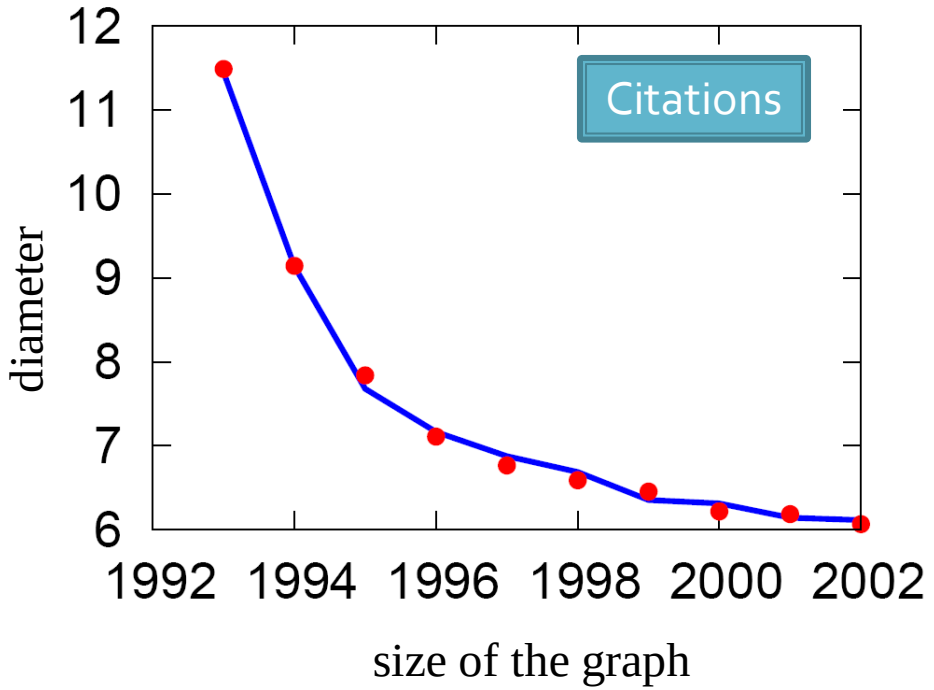


Densifying random graph has increasing diameter⇒ There is more to shrinking diameter than just densification

Diameter of a rewired network

Compare diameter of a:

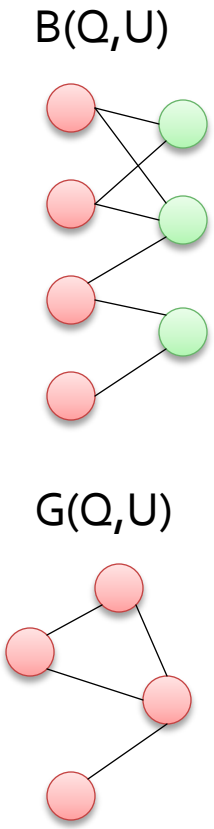
- True network (red)
- Random network with the same degree distribution (blue)



Densification + degree sequence give shrinking diameter

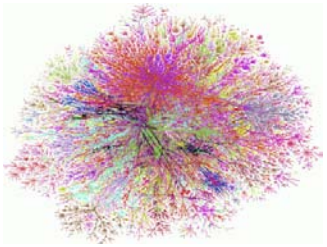
Modeling densification & diameter

- Models:
  - Forest Fire [Leskovec et al. KDD 05]
    - Based on iterative node attachment
  - Kronecker graphs (coming next)
  - Affiliation networks [Lattanzi-Sivakumar STOC 09]
    - Build a scale free bipartite network B(Q,U)
      - Power-law in- and out-degree distribution
    - G(Q, E): fold edges of B:
      - Pair of nodes in G is connected if they share a neighbor in B

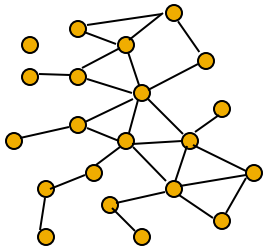


Generating realistic graphs

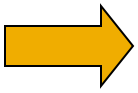
- Want to generate realistic networks:

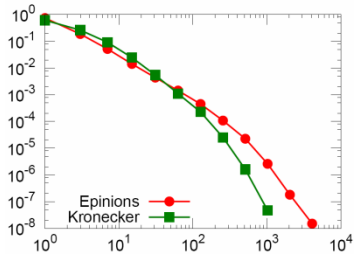


Given a real network



Generate a synthetic network





Compare graphs properties, *e.g.*, degree distribution

- Why synthetic graphs?
  - Anomaly detection, Simulations, Predictions, Null-model, Sharing privacy sensitive graphs, …
- **Q:** Which network properties do we care about?
- **Q:** What is a good model and how do we **fit** it?

Kronecker product: Definition

- Kronecker product of matrices A and B is given by

$$\underset{N\,x\,M}{\mathbf{C}} = \underset{N\,x\,M}{\mathbf{A}} \otimes \underset{K\,x\,L}{\mathbf{B}} \doteq \underset{N*K\,x\,M*L}{\begin{pmatrix} a_{1,1}\mathbf{B} & a_{1,2}\mathbf{B} & \dots & a_{1,m}\mathbf{B} \\ a_{2,1}\mathbf{B} & a_{2,2}\mathbf{B} & \dots & a_{2,m}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}\mathbf{B} & a_{n,2}\mathbf{B} & \dots & a_{n,m}\mathbf{B} \end{pmatrix}}$$

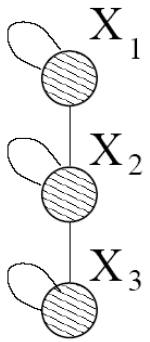
- We define a Kronecker product of two graphs as a Kronecker product of their adjacency matrices

[Leskovec et al. PKDD '05]

Kronecker graphs

- **Kronecker graph:** a growing sequence of graphs by iterating the Kronecker product

$$K_1^{[k]} = K_k = \underbrace{K_1 \otimes K_1 \otimes \ldots K_1}_{k \text{ times}} = K_{k-1} \otimes K_1$$



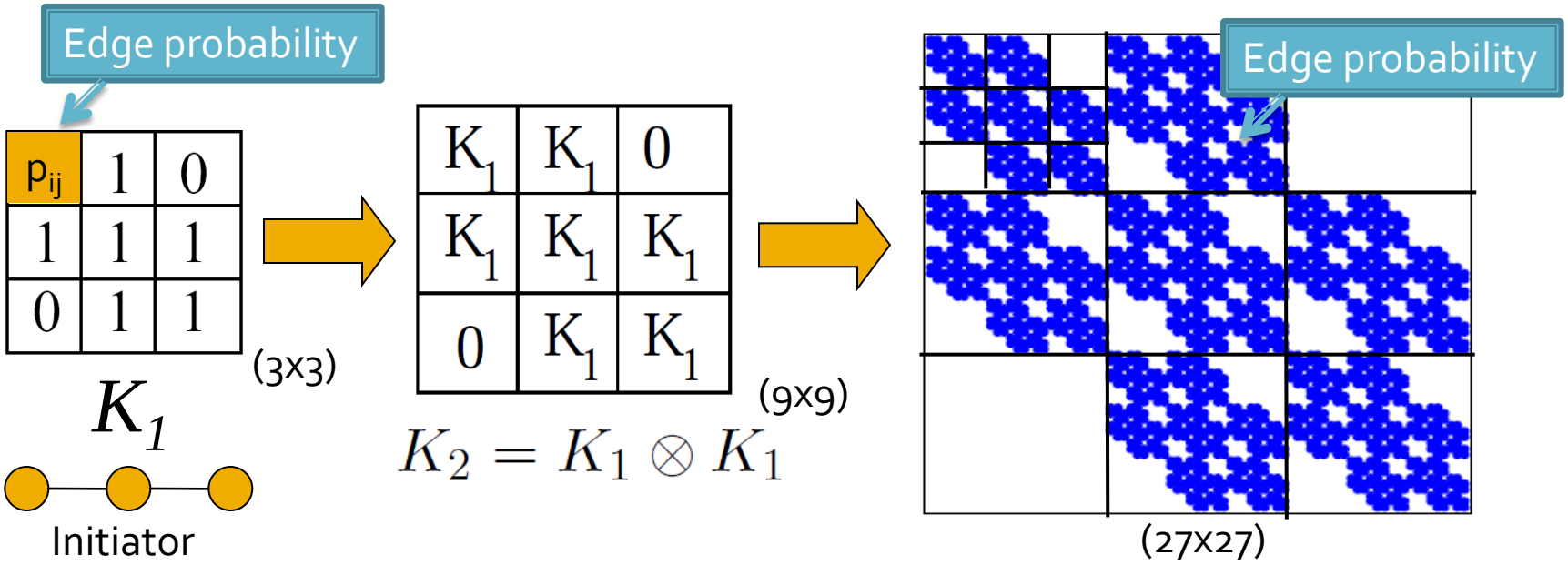
K_1

- Each Kronecker multiplication exponentially increases the size of the graph

- One can easily use multiple initiator matrices (K_1', K_1'', K_1''') that can be of different sizes

[Leskovec et al. PKDD '05]

Kronecker graphs

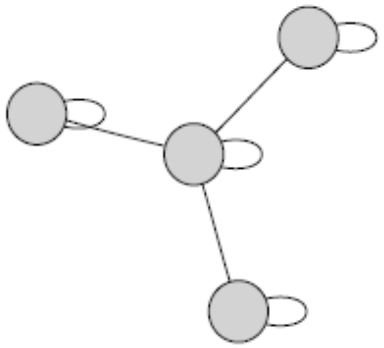


Starting intuition: Recursion & self-similarity

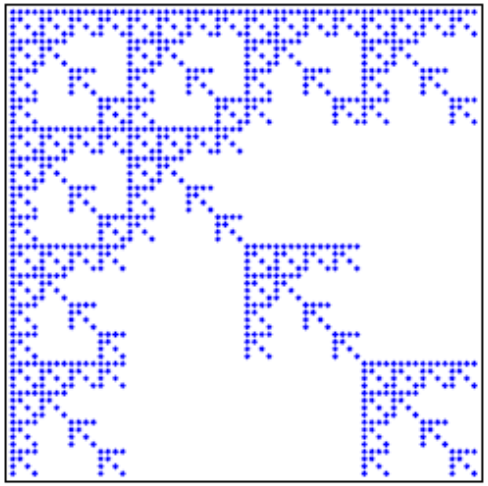
- Kronecker graphs **mimic** real networks:
  - **Theorem:** Power-law degree distribution, Densification, Shrinking/stabilizing diameter, Spectral properties

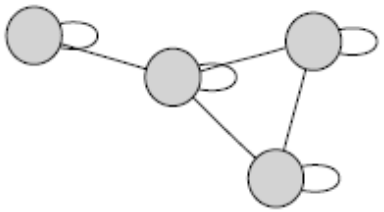
[Leskovec et al. PKDD '05]

Kronecker initiator matrices

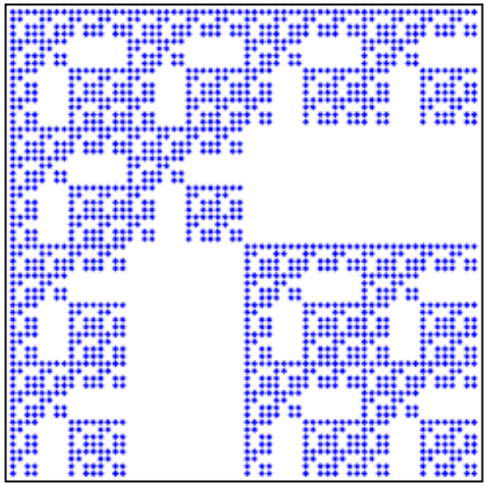


1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	1





1	1	1	1
1	1	0	0
1	0	1	1
1	0	1	1



Initiator K_1 | K_1 adjacency matrix | K_3 adjacency matrix

[Leskovec et al. 09]

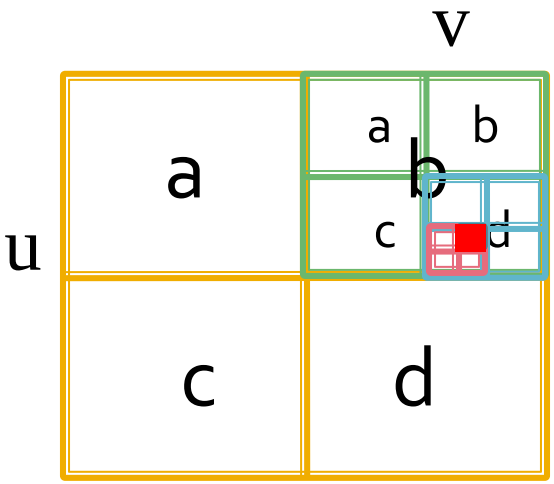
Kronecker graphs: Interpretation

- Initiator matrix K_1 is a similarity matrix
- Node u is described with k binary attributes: $u_1, u_2, \ldots, u_k$
- Probability of a link between nodes u, v:

$$P(u,v) = \prod K_1[u_i, v_i]$$

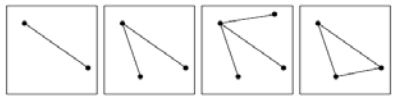
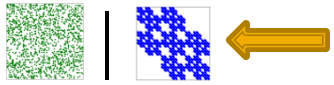
$$K_1 = \begin{array}{c|cc} & 0 & 1 \\ \hline 0 & a & b \\ 1 & c & d \end{array}$$

$$u = (0,1,1,0)$$
$$v = (1,1,0,1)$$
$$P(u,v) = b \cdot d \cdot c \cdot b$$



Estimating Kronecker graphs

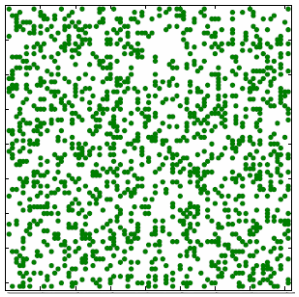
- Given a real network G

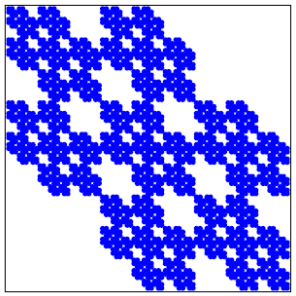
  Want to estimate initiator matrix: $K_1 = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$
- Method of moments [Owen '09]
  - Compare counts of  and solve system of equations.
- Maximum likelihood [Leskovec-Faloutsos ICML '07]
  - arg max P( | G_1)
- SVD [VanLoan-Pitsianis '93]
  - Can solve $\min \left\| G - K_1 \otimes K_1 \right\|_F^2$ using SVD

Kronecker graphs: Estimation

- Maximum likelihood estimation

$$\arg\max_{G_1} P(\quad \mid \quad \text{Kronecker} \Leftarrow K_1)$$





$$K_1 = \begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \hline \end{array}$$

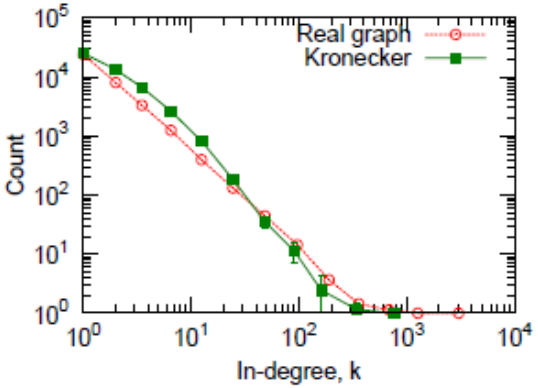
- Naïve estimation takes $O(N!N^2)$:
  - $N!$ for different node labelings:
    - **Our solution:** Metropolis sampling: $N!$ → ***(big) const***
  - N^2 for traversing graph adjacency matrix
    - **Our solution:** Kronecker product ($E << N^2$): N^2 → E
- Do gradient descent

Estimate the model in O(E)

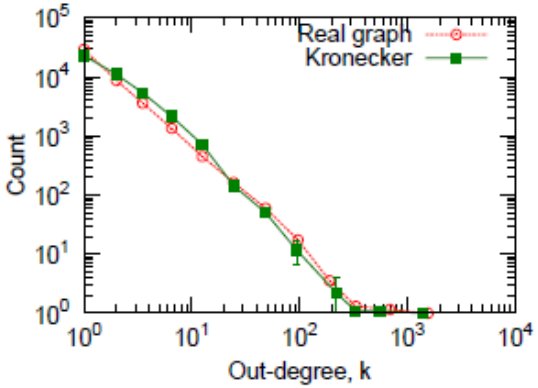
Estimation: Epinions (n=76k, m=510k)

- Real and Kronecker are very close:

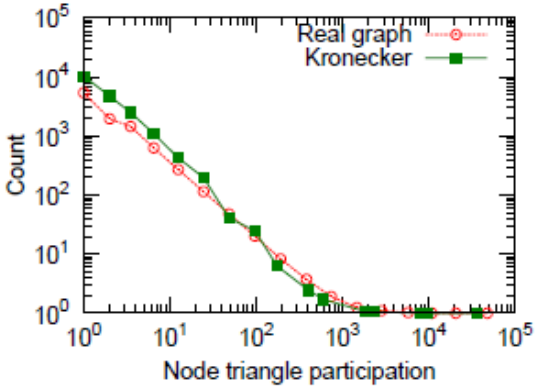
$$K_1 = \begin{bmatrix} 0.99 & 0.54 \\ 0.49 & 0.13 \end{bmatrix}$$



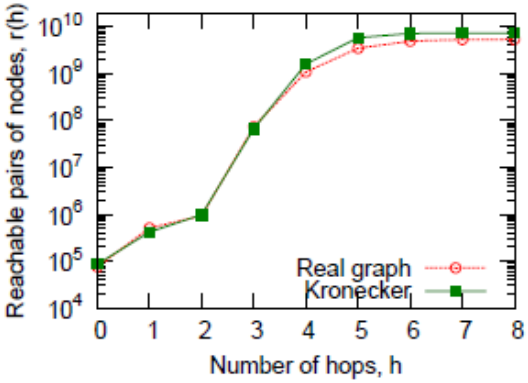
(a) In-Degree



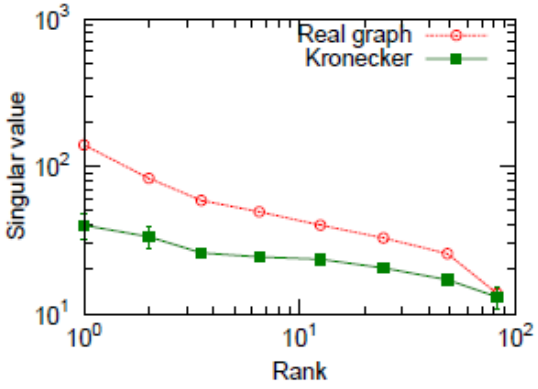
(b) Out-degree



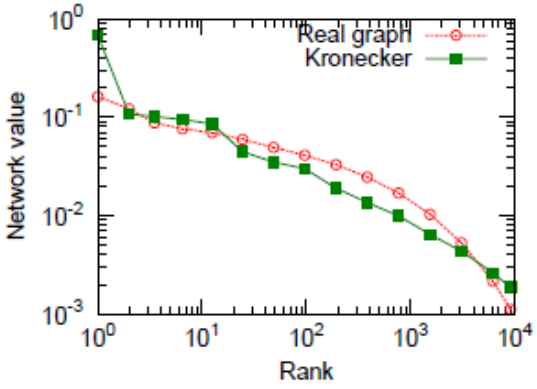
(c) Triangle participation



(d) Hop plot



(e) Scree plot



(f) "Network" value

Kronecker: Consequences

- We can generate realistic looking networks:
  - Simulations of new algorithms where real graphs are hard/impossible to collect
  - Anomaly detection – abnormal behavior, evolution
  - Predictions – predicting future from the past
  - Hypothesis testing
  - Graph sampling – many real world graphs are too large to deal with
  - *"What if"* scenarios

Modeling local network structure (links)

- Link prediction in networks
- Hierarchical random graphs
- Exponential random graphs
- Statistical relational learning

Link prediction in networks

- Network modeling is all about predicting links but so far we have not tackled this problem directly
- Task: predict missing links in a network
  - In a evolving network
  - In a static network
- 2 types of approaches:
  - Node distance approaches:
    - define a distance function, closer nodes are more likely to link
  - Statistical approaches:
    - Design a model of link creation and fit to data

[LibenNowell-Kleinberg CIKM '03]

Link prediction via node distance

- Link prediction in a evolving network:
  - Task: Given $G[t_0,t_0']$ a graph on edges up to time t_0' output a ranked list L of links (not in $G[t_0,t_0']$) that are predicted to appear in $G[t_1,t_1']$
  - Evaluation: $n=|E_{new}|$: # new edges that appear during the test period $[t_1,t_1']$
    Take top n elements of L and count correct edges

[LibenNowell-Kleinberg CIKM '03]

Link prediction via node distance

- Predict links a evolving collaboration network

	training period			Core		
	authors	papers	collaborations[1]	authors	$\|E_{old}\|$	$\|E_{new}\|$
astro-ph	5343	5816	41852	1561	6178	5751
cond-mat	5469	6700	19881	1253	1899	1150
gr-qc	2122	3287	5724	486	519	400
hep-ph	5414	10254	47806	1790	6654	3294
hep-th	5241	9498	15842	1438	2311	1576

- Core: since network data is very sparse
  - Consider only nodes with in-degree and out-degree of at least 3

[LibenNowell-Kleinberg CIKM '03]

Link prediction via node distance

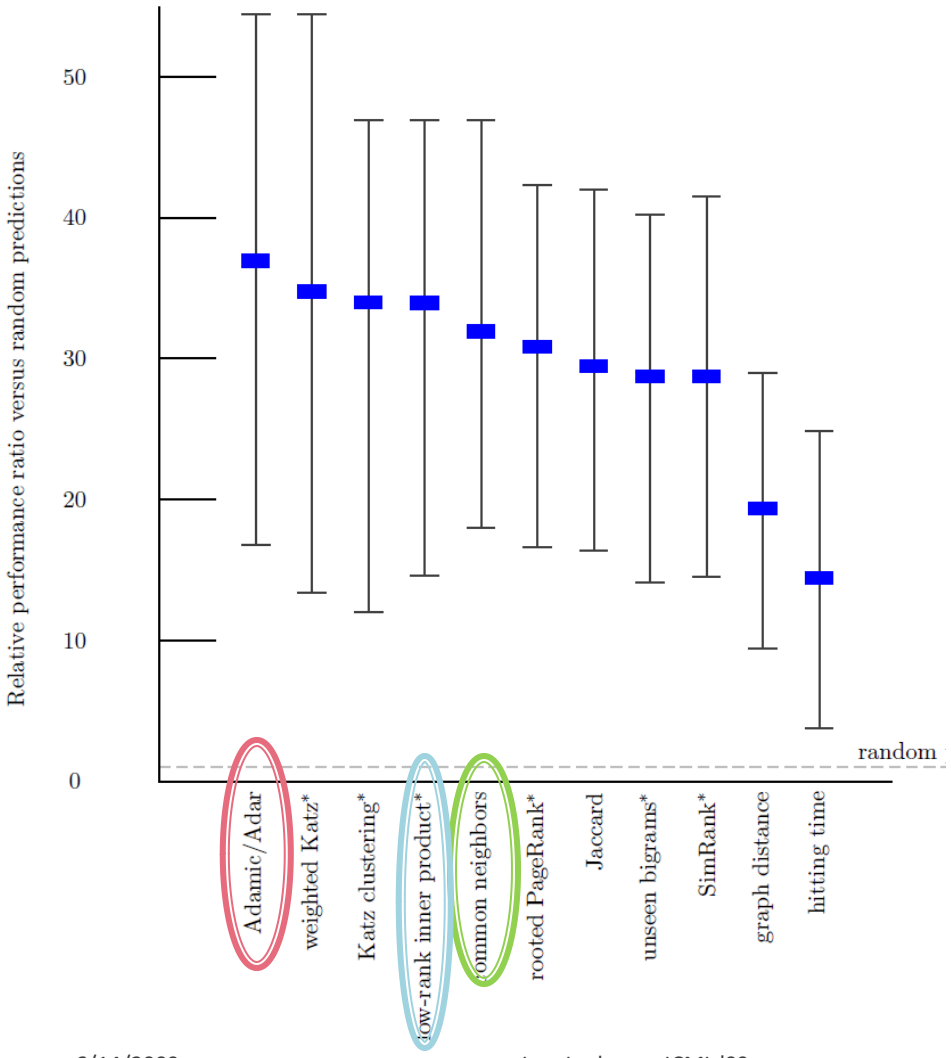
- Rank potential links (x,y) based on:

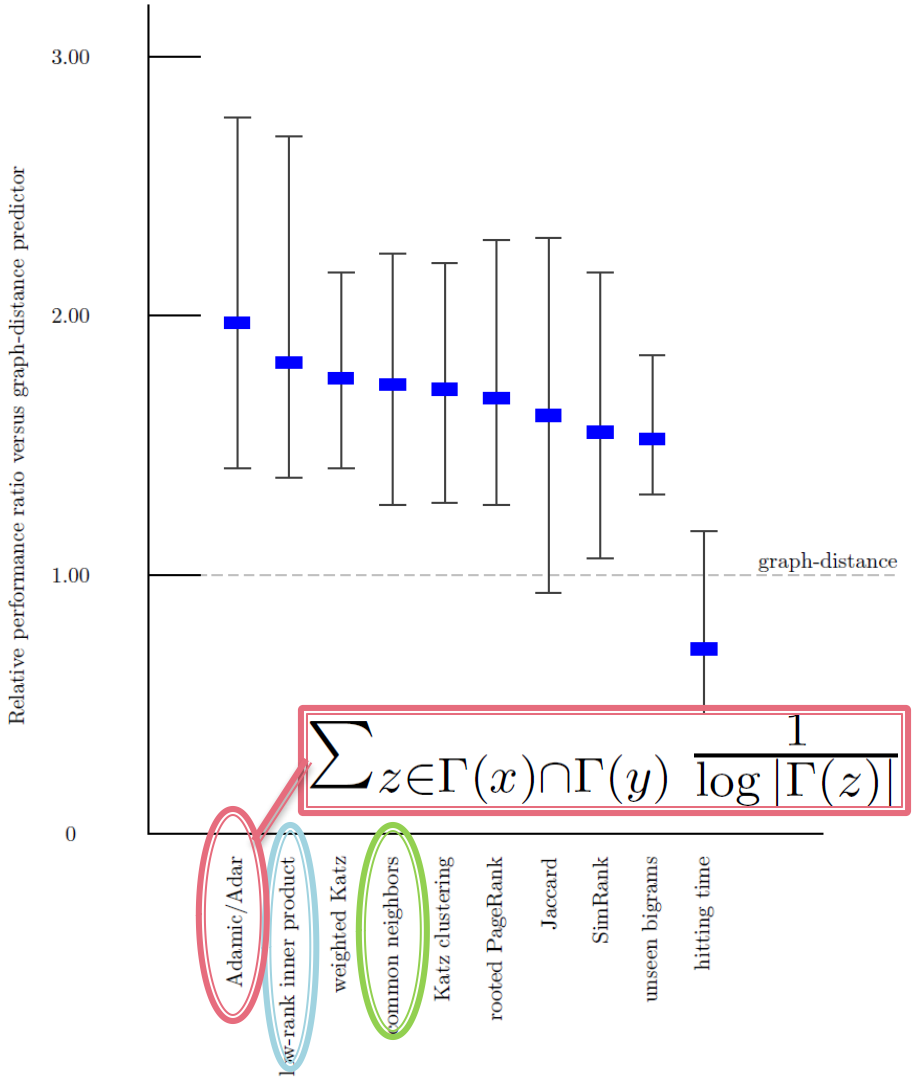
graph distance	(negated) length of shortest path between x and y
common neighbors	$\lvert\Gamma(x) \cap \Gamma(y)\rvert$
Jaccard's coefficient	$\frac{\lvert\Gamma(x)\cap\Gamma(y)\rvert}{\lvert\Gamma(x)\cup\Gamma(y)\rvert}$
Adamic/Adar	$\sum_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log\lvert\Gamma(z)\rvert}$
preferential attachment	$\lvert\Gamma(x)\rvert \cdot \lvert\Gamma(y)\rvert$
Katz$_\beta$	$\sum_{\ell=1}^{\infty} \beta^\ell \cdot \lvert\mathsf{paths}_{x,y}^{\langle\ell\rangle}\rvert$
	where $\mathsf{paths}_{x,y}^{\langle\ell\rangle}$:= {paths of length exactly ℓ from x to y}
	weighted: $\mathsf{paths}_{x,y}^{\langle 1\rangle}$:= number of collaborations between x, y.
	unweighted: $\mathsf{paths}_{x,y}^{\langle 1\rangle}$:= 1 iff x and y collaborate.
hitting time	$-H_{x,y}$
stationary-normed	$-H_{x,y} \cdot \pi_y$
commute time	$-(H_{x,y} + H_{y,x})$
stationary-normed	$-(H_{x,y} \cdot \pi_y + H_{y,x} \cdot \pi_x)$
	where $H_{x,y}$:= expected time for random walk from x to reach y
	π_y := stationary-distribution weight of y (proportion of time the random walk is at node y)

$\Gamma(x)$ … degree of node x

[LibenNowell-Kleinberg CIKM' 03]

Link prediction via node distance

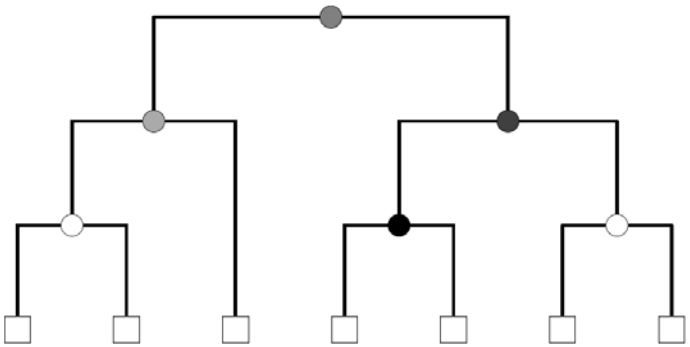




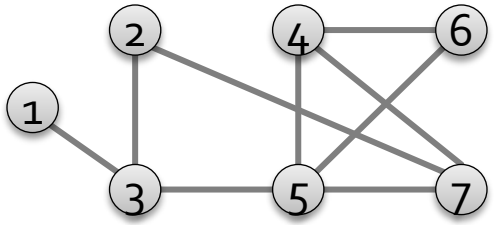
[Clauset et al. Nature '08]

Hierarchical random graphs

- Hierarchical model of network structure
- Tree D:
  - Leaves of D correspond to nodes of the network
  - Internal nodes of D have Bernoulli parameters θ_i associated with them (edge probability)
  - Prob. of edge (u,v) is θ_x where x is the least common ancestor of leaves u and v



Tree: shade corresponds to value of θ_i

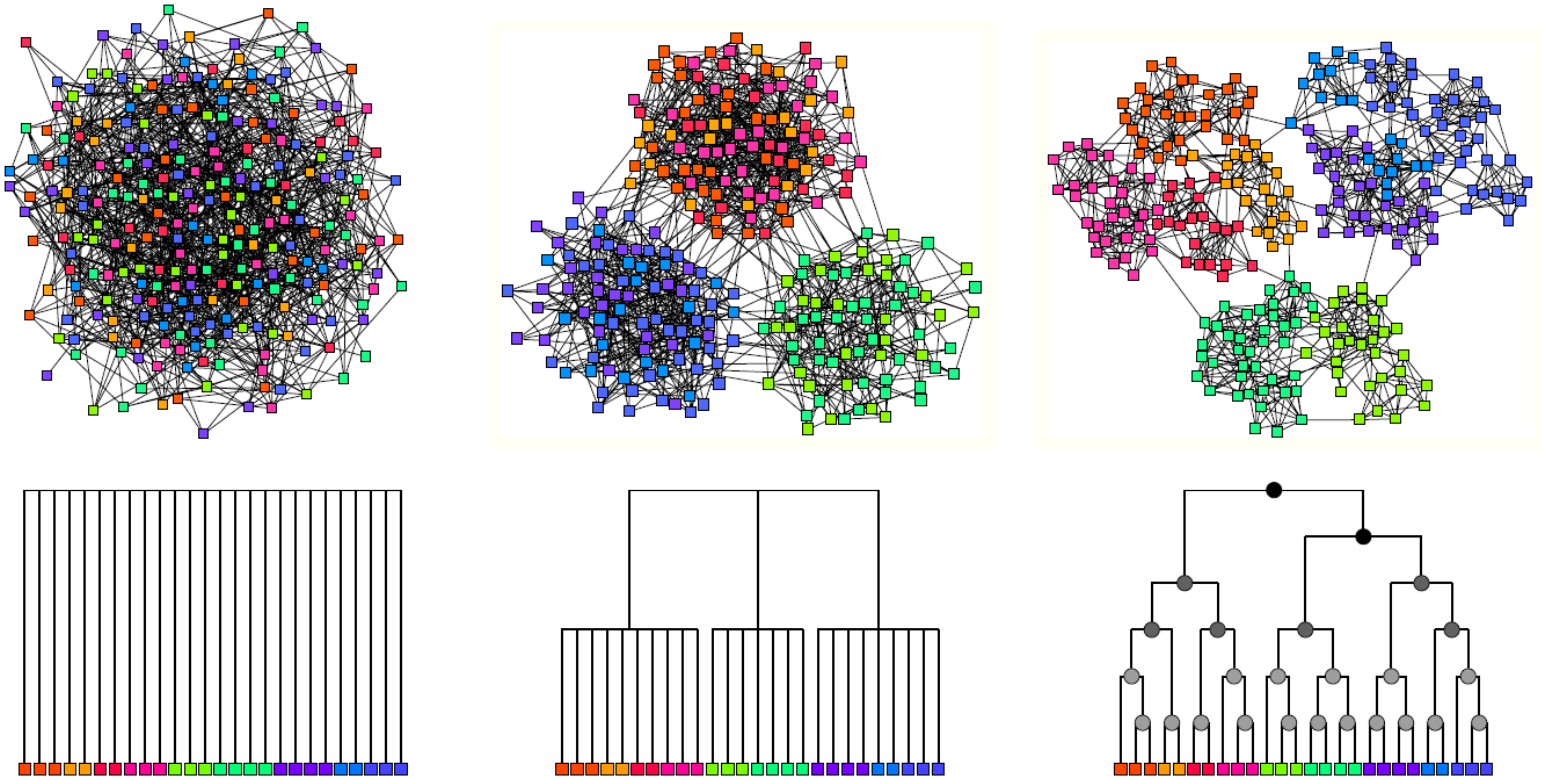


Corresponding graph

[Clauset et al. Nature '08]

HRG: More examples

- Graphs and corresponding hierarchies:



[Clauset et al. Nature '08]

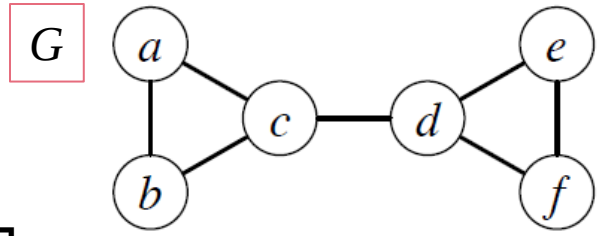
HRG: Model estimation (1)

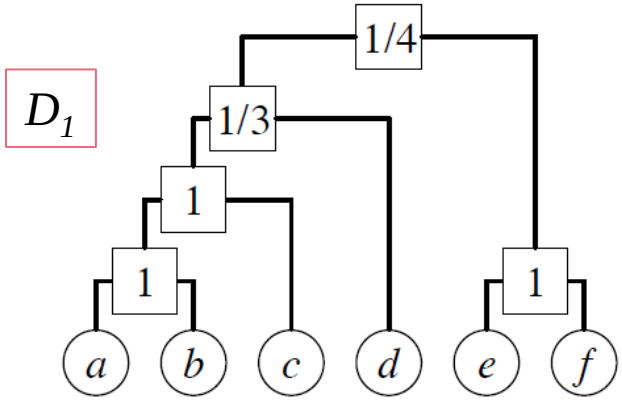
- Given a graph G and a model D
- How do we compute the likelihood $L(D)=P(G|D)$?

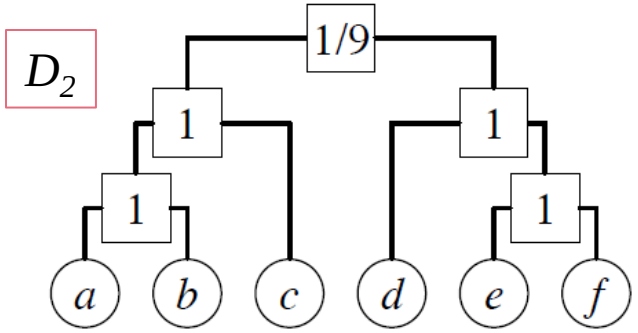
$$\mathcal{L}_{\mathcal{H}}(\mathcal{D},\vec{\theta}) = \prod_{i=1}^{n-1} (\theta_i)^{E_i} (1-\theta_i)^{L_i R_i - E_i}$$

L_i, R_i: # edges in left/right subtree
E_i: # edges between the subtrees

- Example:







$$\mathcal{L}(D_1) = (1/3)(2/3)^2 \cdot (1/4)^2 (3/4)^6 = 0.00165$$

$$\mathcal{L}(D_2) = (1/9)(8/9)^8 = 0.0433$$

[Clauset et al. Nature '08]

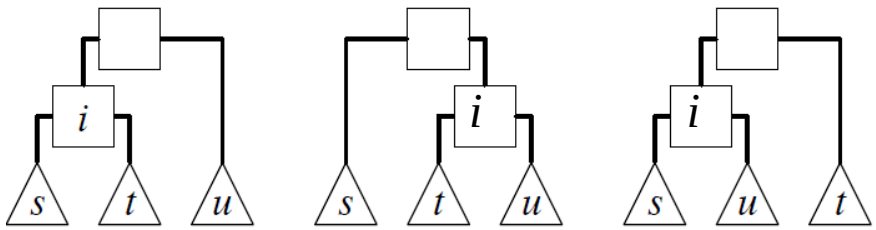
HRG: Model estimation (2)

- How estimate model parameters θ_i?
  - Just count number of edges between the subtrees:

$$\theta_i = E_i / L_i R_i$$

L_i, R_i: # edges in left/right subtree
E_i: # edges between the subtrees

- How to learn the tree?
  - Markov Chain Monte Carlo to the rescue to stochastically search over the network structures
  - Each internal node i can be in one of the 3 configurations:



Algorithm:
1. Randomly pick internal node *i*
2. Randomly pick one of the two alternative configurations
3. Accept the change based on likelihood ratio

[Clauset et al. Nature '08]

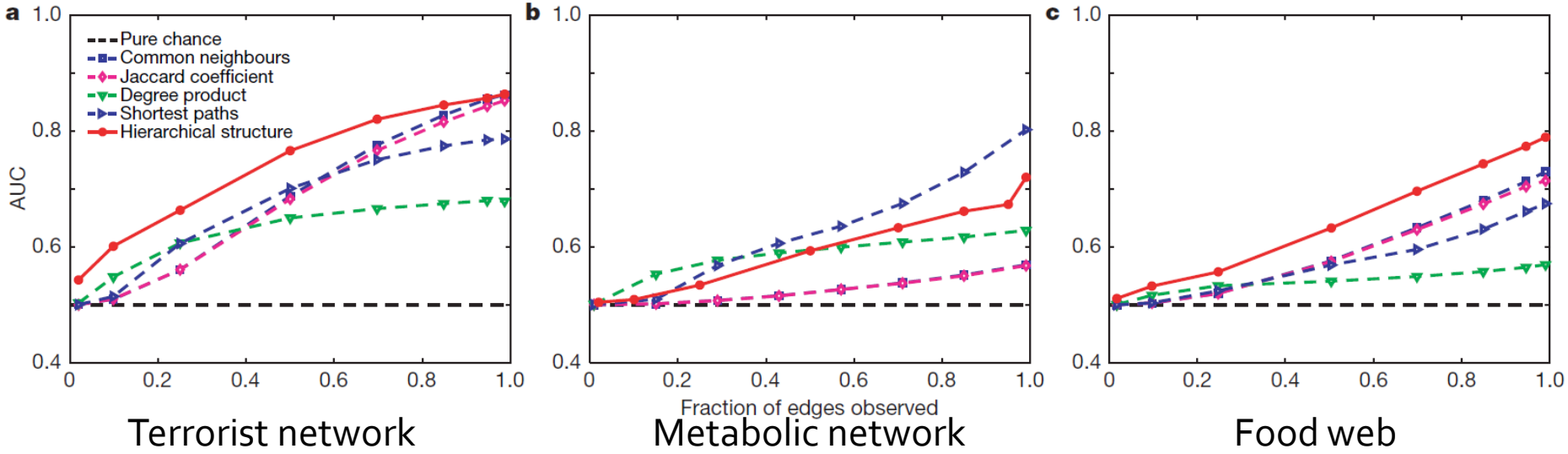
HRG: Consensus hierarchies

- The model has linear number (n) parameters
  - Possible problem due to overfitting
  - Solution: Model averaging
    - 1: do MCMC so that it converges to stationary distribution
    - 2: sample models from stationary distribution and then out Majority Consensus model (tree):
      - Each model D_i has a score (likelihood $L(D_i)$)
      - Use tree consensus procedure

[Clauset et al. Nature '08]

HRG: Link prediction

- Setting:
  - Given a static network *G* on *m* edges
  - Create graph G' on a random subset of *x* edges
  - Estimate the model on G'
  - Output *m*-*x* most likely edges
- Results AUC: 3 networks with tens of nodes

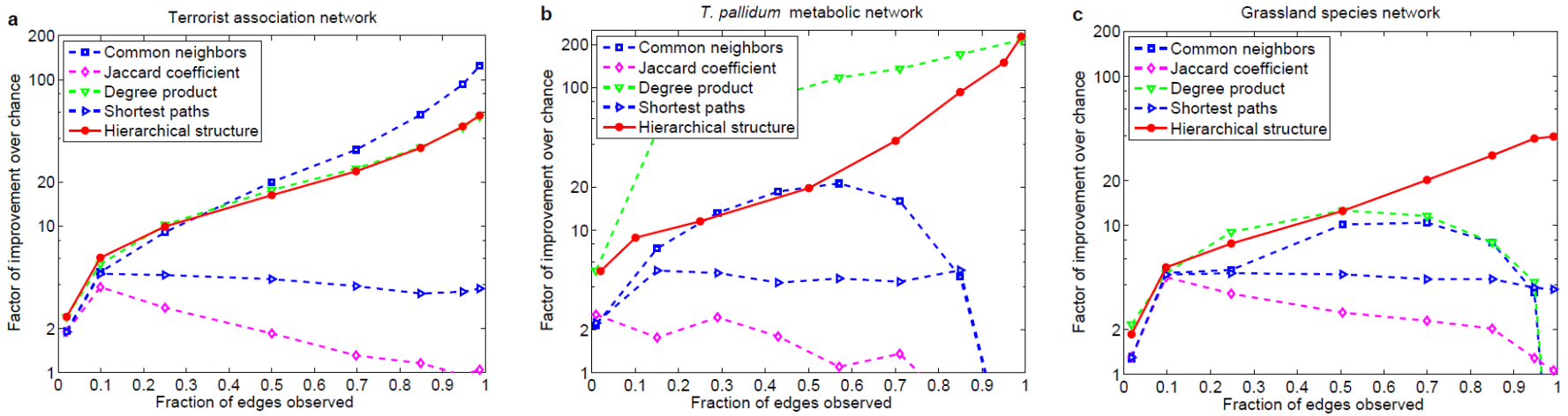


Terrorist network | Metabolic network | Food web

[Clauset et al. Nature' 08]

HRG: Link prediction (2)

- Results:
  - Improvement over random



Exponential random graphs (p*)

- Mainly used by statistics and traditional social network analysis community
- Descriptive model: numerical summary measures
  - Nodal level: centrality, node attributes
  - Configuration level: cycles, triads, reciprocity
  - Network level: clustering, core-periphery
- Generative:
  - Test alternative hypotheses
  - Extrapolate and simulate from the model

p*: Definition

- Log-linear model over graph configurations:
  - Unit of analysis: an edge (dyad)
  - Observations (edges) are dependent
- In the most general case:

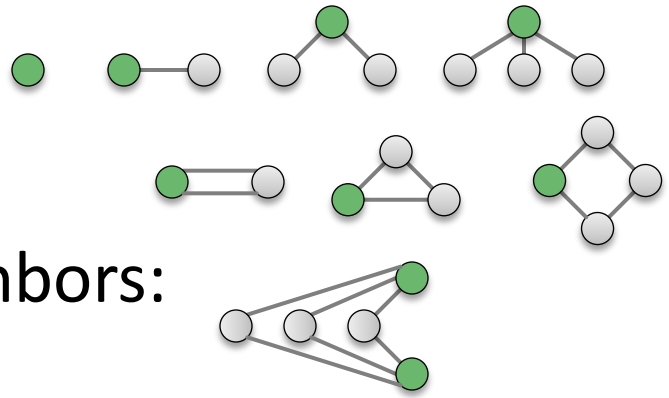
$$P(Y = y) = \frac{1}{Z}\exp\left(\sum_A \theta_A g_A(y)\right)$$

  - where :
    - *A:* (labeled) configurations
    - θ_A: parameter for configuration *A*
    - $g_A(y)$*:* if configuration *A* is present $g_A(y)=1$ else $g_A(y)=0$
    - *Z*: normalizing constant (sum over all possible graphs!)

(We usually replace $g(y)$ with $g(y)$-$g(y^{obs})$)

p*: What kinds of features?

- Attributes of nodes:
  - Characteristics of a group: activity
  - Individual's characteristics
- Attributes of links:
  - Characteristics of links: duration, type
- Configurations:
  - Node degree:
  - Cycles:
  - Common neighbors:



Edge independent terms

Edge dependent terms

[Holland-Leinhardt '81]

Example: p1 (earliest ERG model)

- Edge independence model:

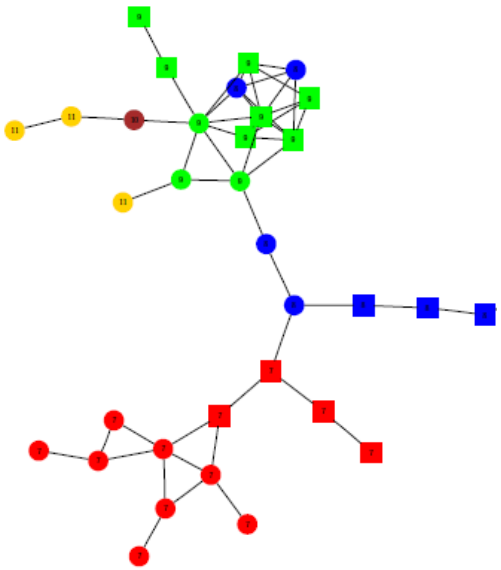
$$P_\eta(Y = y) = \frac{\exp\{\rho \sum_{i<j} y_{ij} y_{ji} + \phi y_{++} + \sum_i \alpha_i y_{i+} + \sum_j \beta_j y_{+j}\}}{\kappa(\rho, \alpha, \beta, \phi)}$$

- where:
  - y: observed graph adjacency matrix
  - ϕ: the expected number of edges
  - ρ: tendency toward reciprocation
  - α_i: productivity of a node (out-degree)
  - β_i: attractiveness of a node (in-degree)

p*: Parameter estimation

- Problem: normalizing constant

$$Z = \sum_{\substack{\text{all possible} \\ \text{graphs } y}} \exp\left(\sum_A \theta_A g_A(y)\right)$$



For the graph on left we have to sum over 7,547,924,849,643,082,704,483,109,161,976,537,781,833,842,440,832,880,856,752,412,600,491,248,324,784,297,704,172,253,450,355,317,535,082,936,750,061,527,689,799,541,169,259,849,585,265,122,868,502,865,392,087,298,790,653,952 terms (graphs) ☺

[Hunter '06]

p*: Maximizing likelihood

- Suppose we fix θ_0 then log-likelihood:

$$log\ E[exp((\theta_0 - \theta)g(Y))] = l(\theta) - l(\theta_0)$$

- Law of large numbers says we can approximate true mean by a sample mean
- Thus: $l(\theta) - l(\theta_0) \approx \frac{1}{m}\sum_{i=1}^{m} \exp((\theta_0 - \theta)g(Y_i))$

where $Y_1, Y_2, \ldots, Y_m$ is a random sample of networks from the distribution defined by the model with parameters θ_0

[Hunter '06]

p*: Simulating random graphs

- Goal: simulate random networks Y from the p* model
- Use Markov Chain Monte Carlo:
  - Repeat for a long time:
    - Select a pair of nodes (i,j) at random
    - Calculate likelihood ratio:

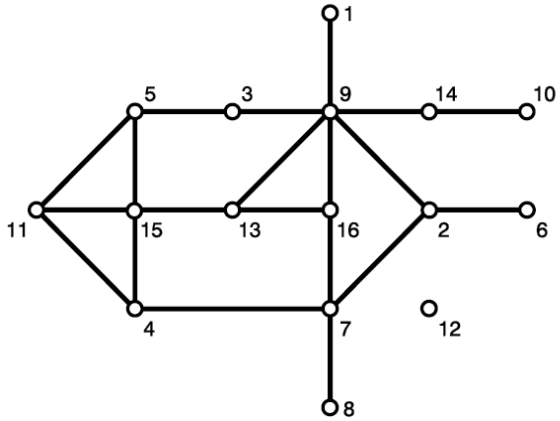
      π = P(Y_{ij} changes) / P(Y_{ij} does not change)

      accept the change with prob. $\min\{1, \pi\}$
- Convergence is agonizingly slow

[Robins et al. '06]

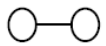
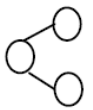
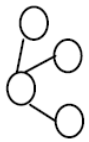
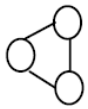
p*: Example

- Graph of relations between Florentine families (n=16,m=19)



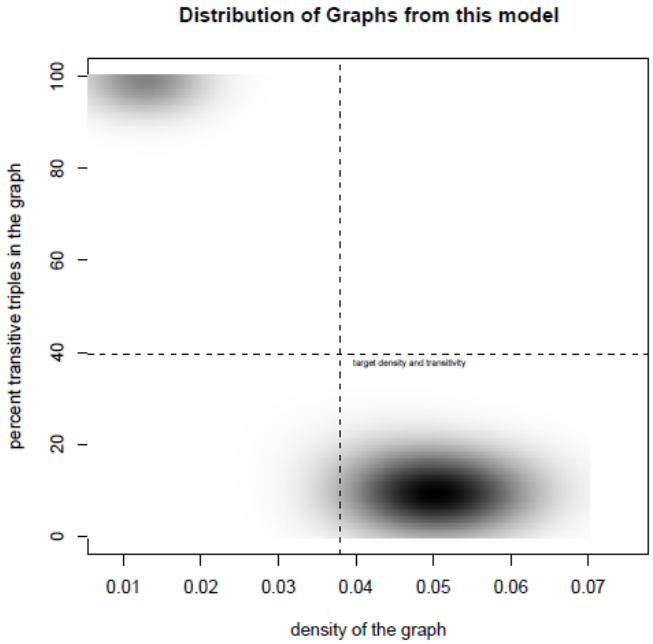
- Decide on the features:
  - Density, Two-star, Three-star, Triangle

- Parameter estimates:
  - $\theta < 0$: edges occur relatively rarely, especially if they are not part of higher order structures
  - $\tau > 0$: business tries tend to occur in triangular structures

Parameter	Configuration	Estimate (stan
θ		-4.27 (1.13)
σ_2		1.09 (0.65)
σ_3		-0.67 (0.41)
τ		1.32 (0.65)

p*: Issues

- Scalability: for graphs up to 1,000 nodes
  - MCMC converges very slowly
  - Computation of features can be expensive
- Model degeneracy:
  - Very small number of graphs has high probability
  - This is a problem for networks with high transitivity (i.e., social networks) as the model clumps triangles together.



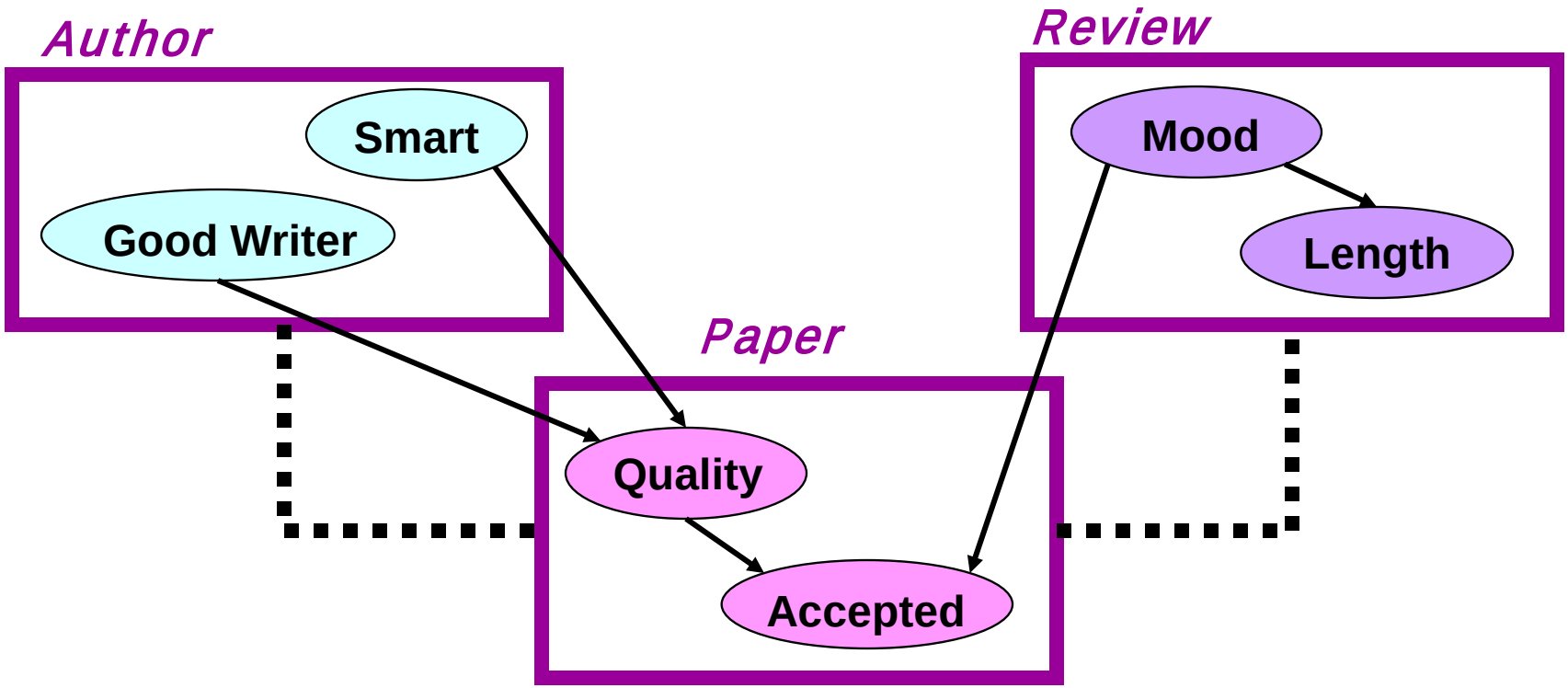
Statistical Relational Learning

- Types of predictive tasks addressed by SRL:
  - Object classification: predict category of an object based on its attributes and links
  - Link classification: predict type of a link
  - Link existence: predict whether a link exists or not
  - Link cardinality estimation: predict the number of links of a node
- Approaches use directed and undirected graphical models
- See Introduction to statistical relational learning by Taskar and Getoor

[Getoor et al.]

SLR: Directed graphical model

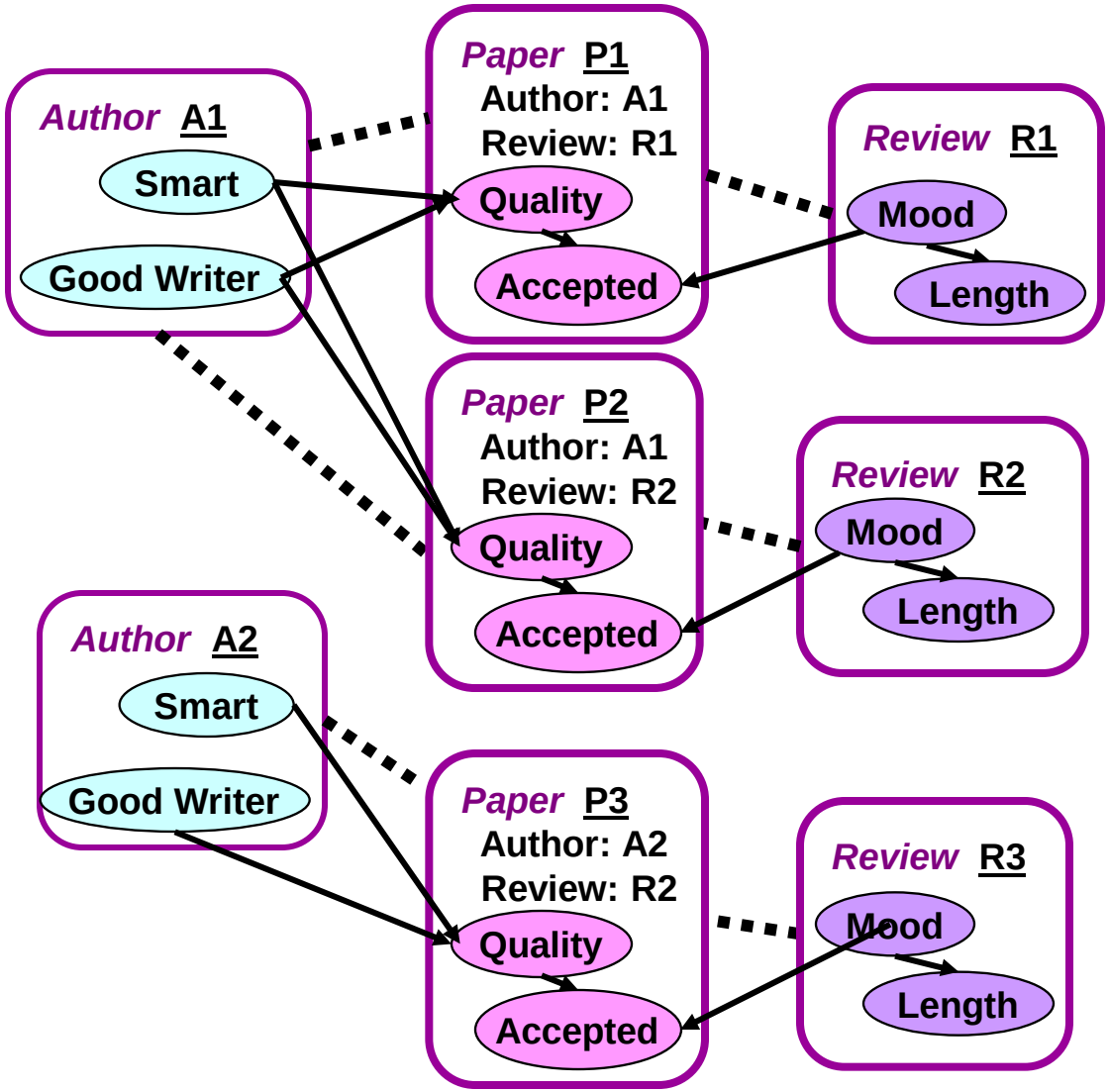
- Will a paper get accepted?
- Templated Bayes network: each entity defines a little graphical model



[Getoor et al.]

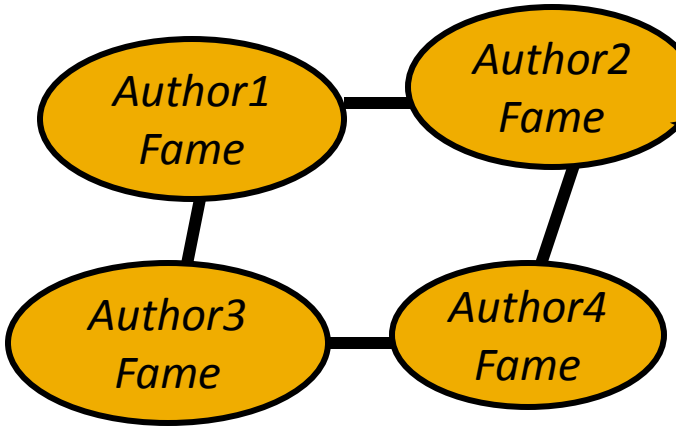
SLR: Directed graphical model

- Data on papers, authors & reviews instantiates a **big** Bayes net
- Some labels are given
- Infer the missing labels
- Collective classification



[Taskar et al. NIPS '01]

SLR: Undirected graphical model



nodes = domain variables
edges = mutual influence

F2	F4	ϕ(F2,F4)
f2	f4	0.6
f2	$\bar{f}4$	0.3
$\bar{f}2$	f4	0.3
$\bar{f}2$	$\bar{f}4$	1.5

Potentials measure compatibility

$$P(f1,\bar{f}2,f3,\bar{f}4) = \frac{1}{Z}\phi_{12}(f1,\bar{f}2)\phi_{13}(f1,f3)\phi_{24}(\bar{f}2,\bar{f}4)\phi_{34}(f3,\bar{f}4)$$

Good news: no acyclicity constraints

Bad news: global normalization (1/Z)

[Taskar et al. NIPS '01]

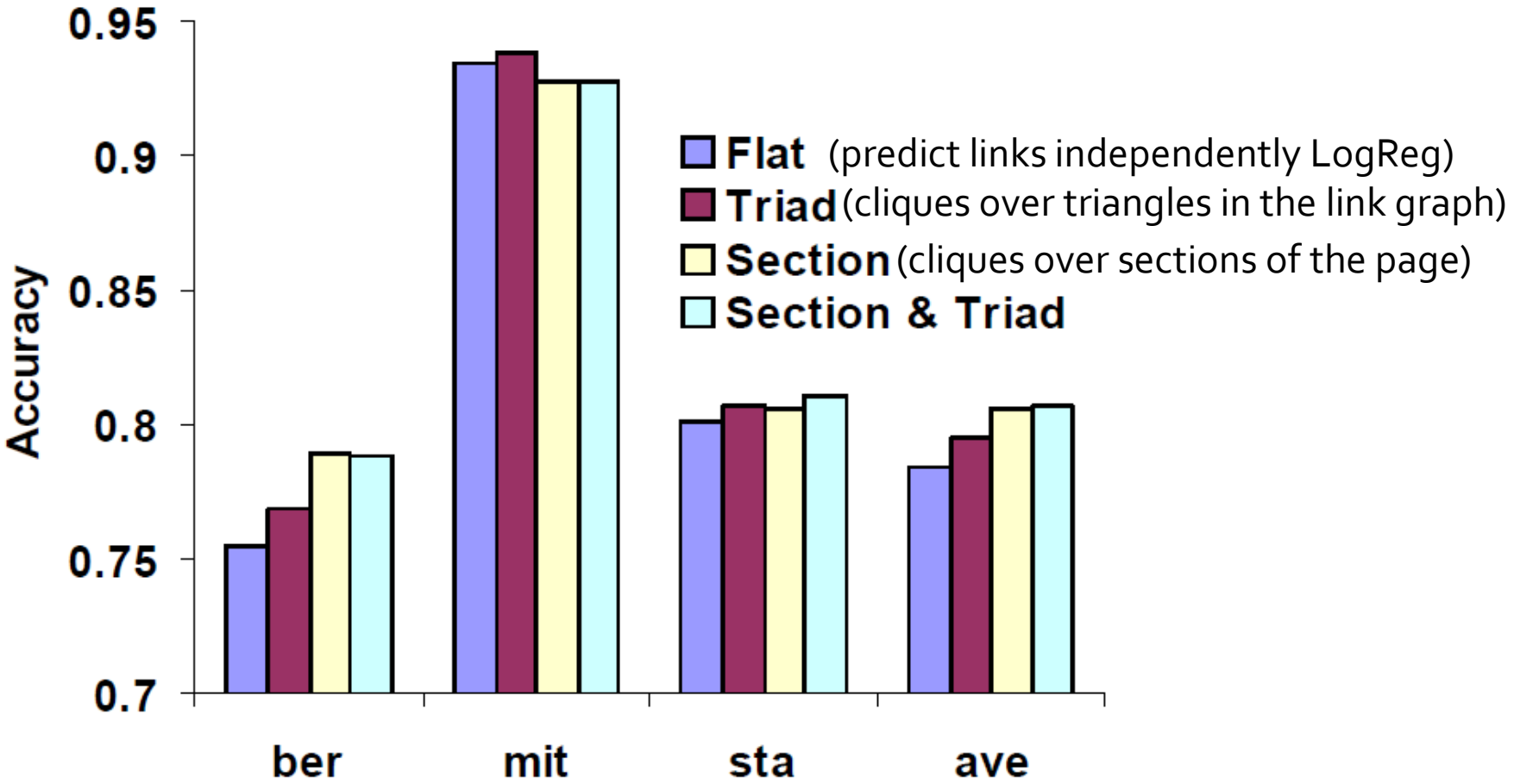
Undirected RL: Web-KB

- Web-KB: university webpages
  - 2954 pages from Stanford, Berkeley, MIT
  - Webpage classes: organization, student, research group, faculty, course, research project, research scientist, staff
  - Predict relation type: Advisor, Member, Teach, TA
  - Train on two universities, predict on one
- Does link structure help in classifying the type of a webpage?

[Taskar et al. NIPS '01]

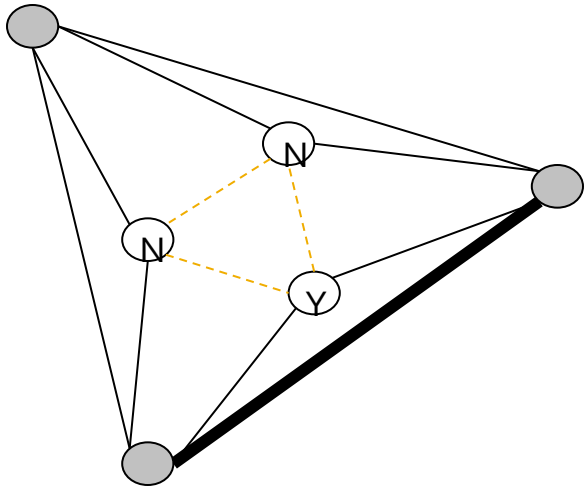
Web-KB: Predicting link type

- Link structure helps with prediction



SLR: Summary

- SLR in a nutshell:
  - Inside each node/edge we have a small graphical model
  - Link structure defines additional dependencies between the variables
  - A big graphical model
- Very good for collective classification
  - predicting node/edge types



- Inference is hard but many clever ideas on exploiting templated structure of the graphical model
- For modeling network structure one has to consider all possible edges and then for each infer its presence/absence.

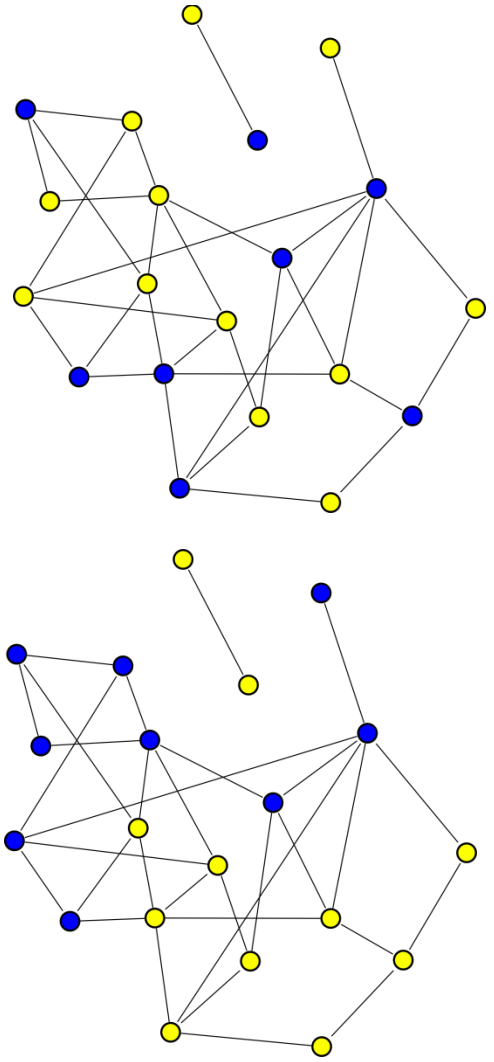
Network structure at the level of groups of nodes

- Group formation
- Finding groups/communities/clusters
- Modular structure in networks
- Consequences

[Backstrom et al. KDD '06]

Group formation in networks

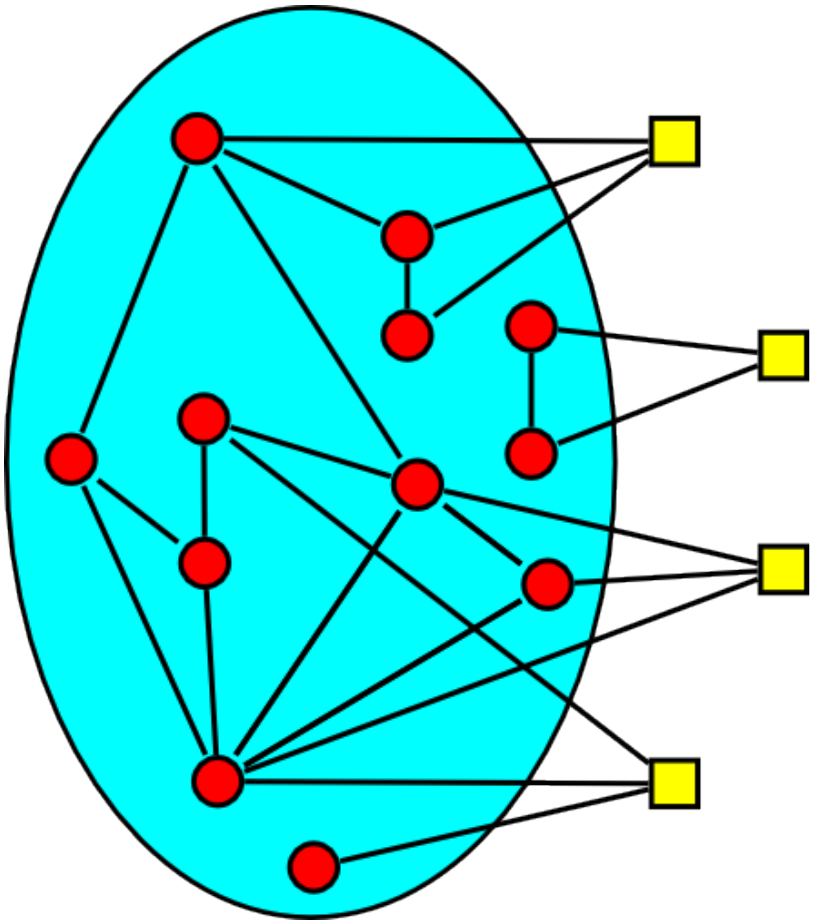
- In a social network nodes explicitly declare group membership:
  - Facebook groups
  - Publication venue
- Can think of groups as node colors
- Gives insights into social dynamics:
  - Recruits friends? Memberships spread along edges
  - Doesn't recruit? Spread randomly
- What factors influence a person's decision to join a group?
- What factors indicate that a group will grow in membership?



[Backstrom et al. KDD '06]

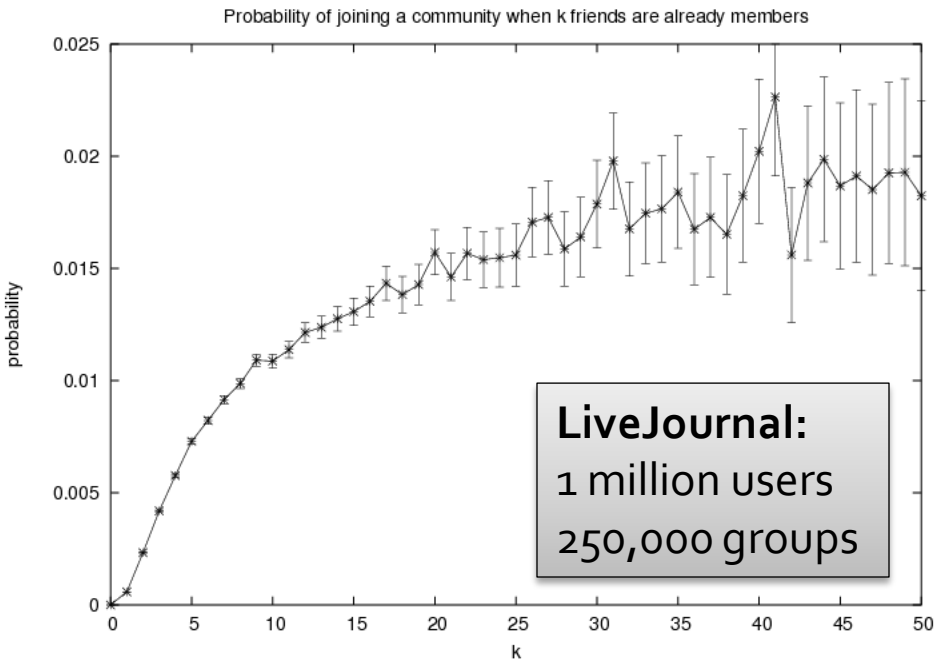
Group growth as diffusion

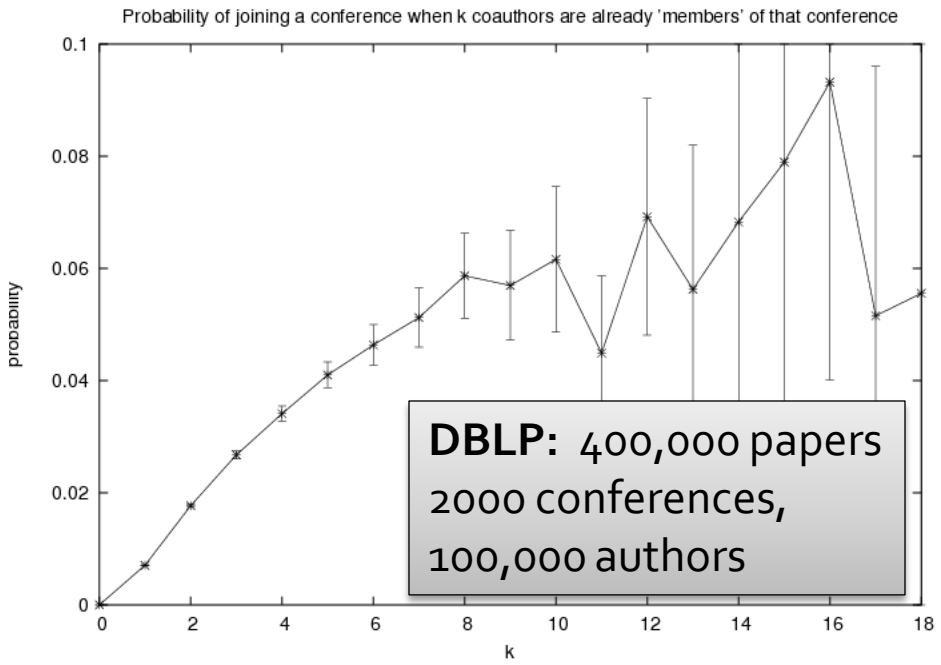
- Analogous to diffusion:
  Group memberships spread over the network:
  - Red circles represent existing group members
  - Yellow squares may join
- Question:
  - How does prob. of joining a group depend on the number of friends already in the group?



[Backstrom et al. KDD '06]

P(join) vs. # friends in the group





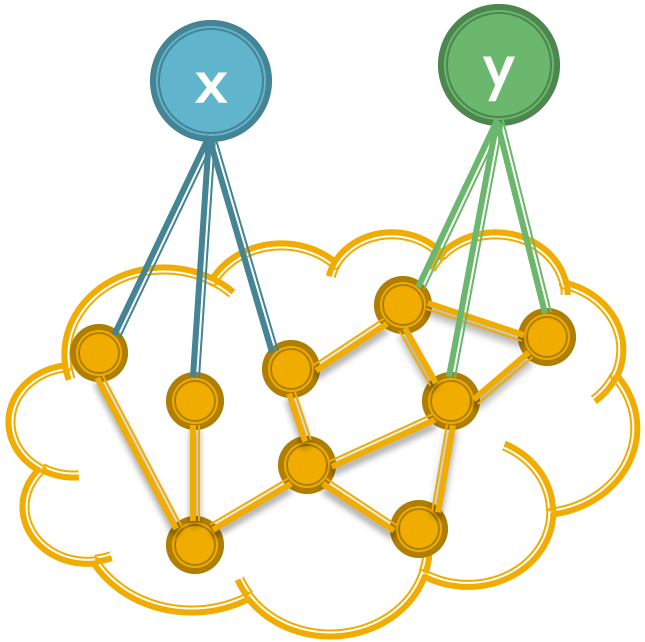
- Diminishing returns:
  - Probability of joining increases with the number of friends in the group
  - But increases get smaller and smaller

[Backstrom et al. KDD '06]

Groups: More subtle features

- Connectedness of friends:
  - x and y have three friends in the group
  - x's friends are independent
  - y's friends are all connected

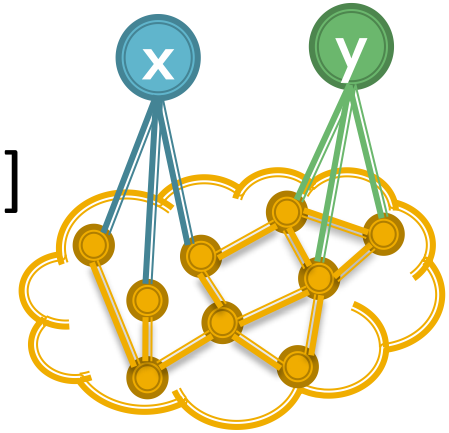
Who is more likely to join?



[Backstrom et al. KDD '06]

Connectedness of Friends

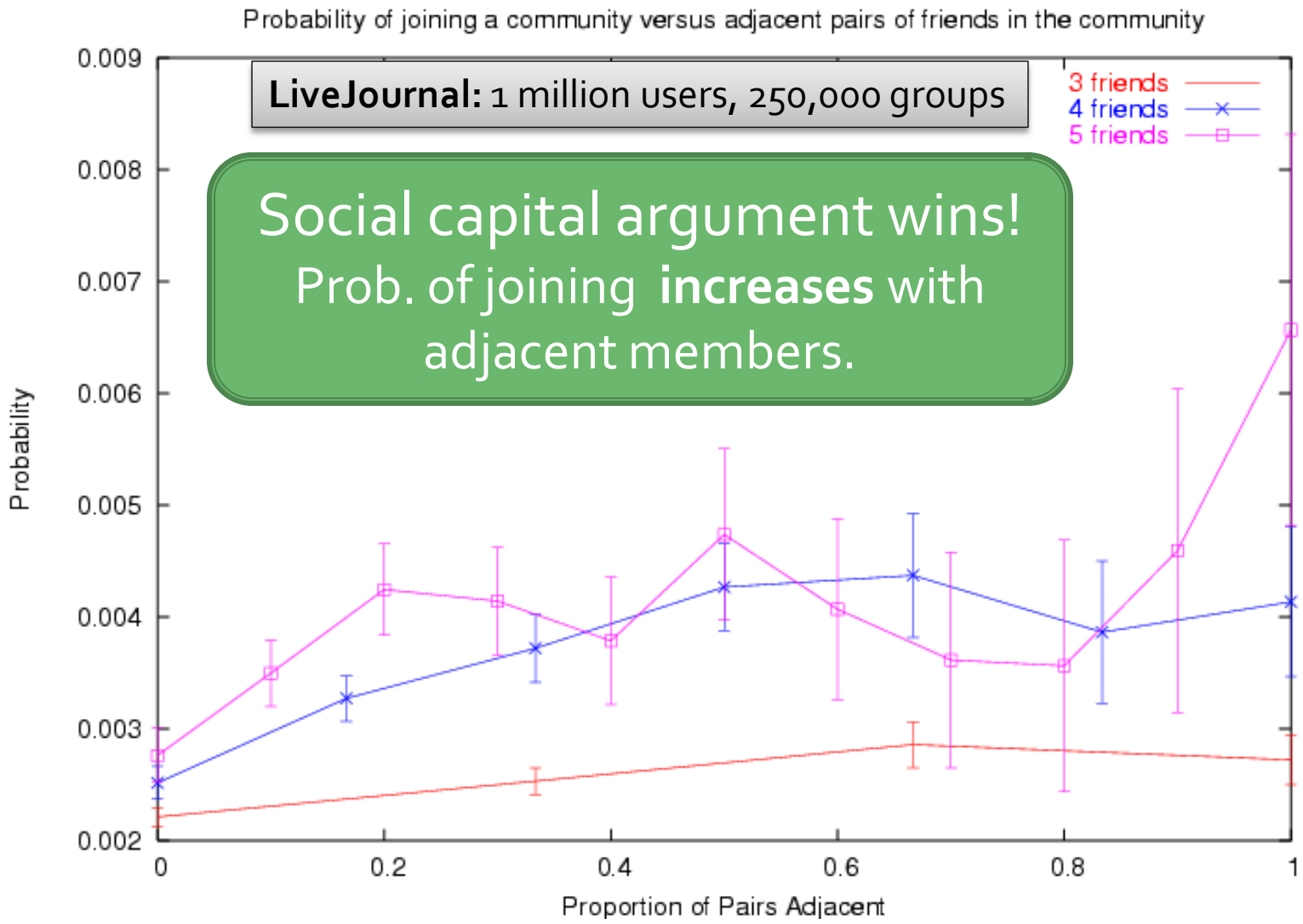
- Competing sociological theories
  - Information argument [Granovetter '73]
  - Social capital argument [Coleman '88]



- Information argument:
  - Unconnected friends give independent support
- Social capital argument:
  - Safety/truest advantage in having friends who know each other

[Backstrom et al. KDD '06]

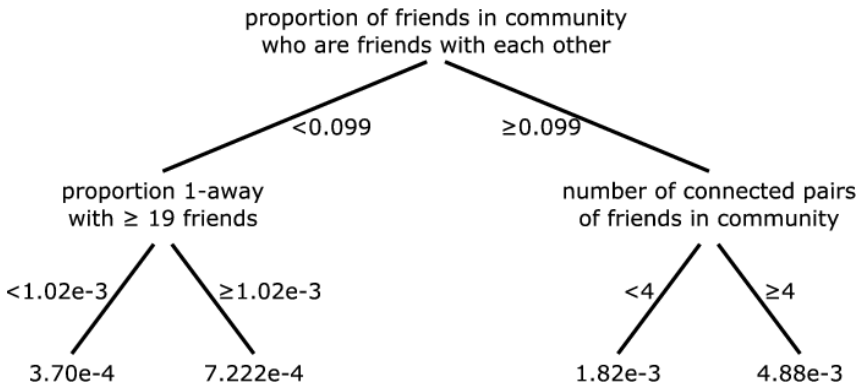
Connectedness of friends

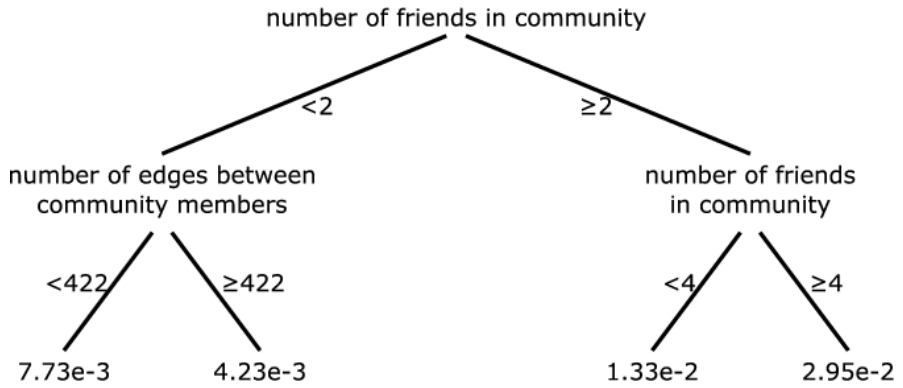


[Backstrom et al. KDD '06]

Groups: More subtle features

- Predict whether a user will join a group
- Important features:
  - Group activity level in LiveJournal (# posts).
  - Internal connectedness of friends.
  - Other topological features of the friendship graphs

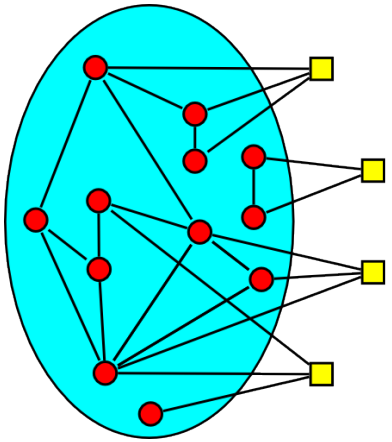




[Backstrom et al. KDD '06]

Predicting group growth

- Predict whether group will grow significantly:
  - Less than 9% vs. greater than 18%
- Predicting based on fringe size does not do well
- Using more sophisticated features gives good performance
  - Number of closed triads
  - Number of people with at least 10 friends in the group
  - Total number of friendships

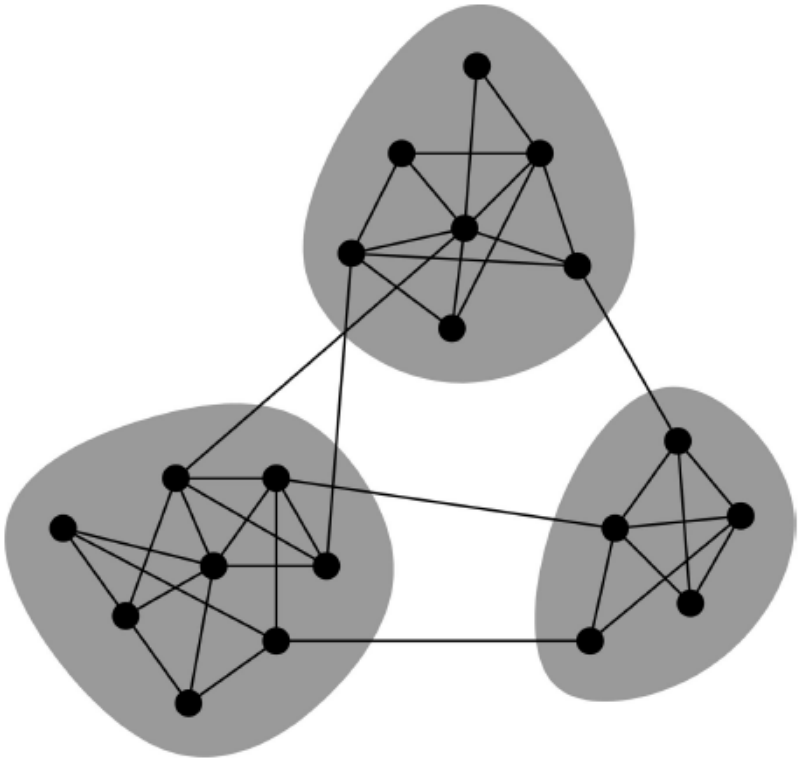


Feature	AUC
Fringe Size	0.559
Group Size	0.521
Fringe/Group	0.562
Above Three	0.601
All network features	0.771

Network communities

- Findings so far suggest that network groups are **tightly connected**

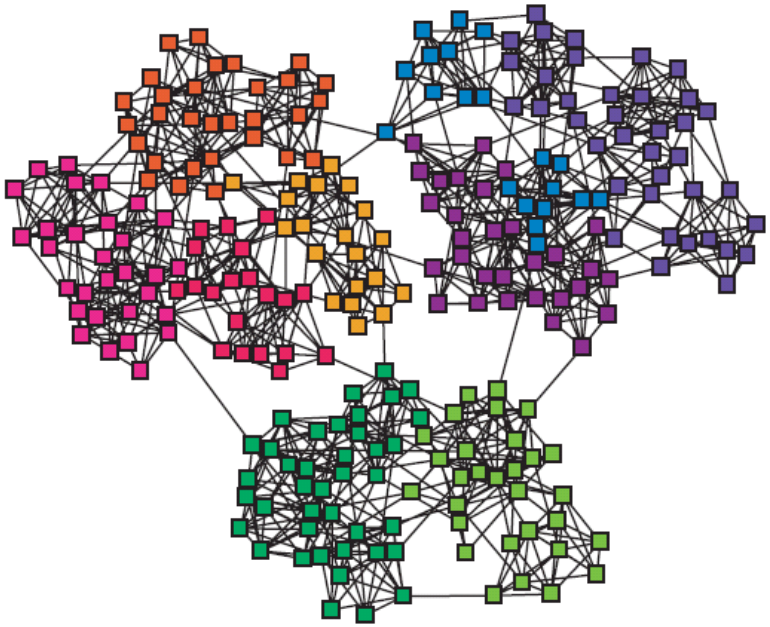
- **Network communities:**
  - Sets of nodes with **lots** of connections **inside** and **few** to **outside** (the rest of the network)



Communities, clusters, groups, modules

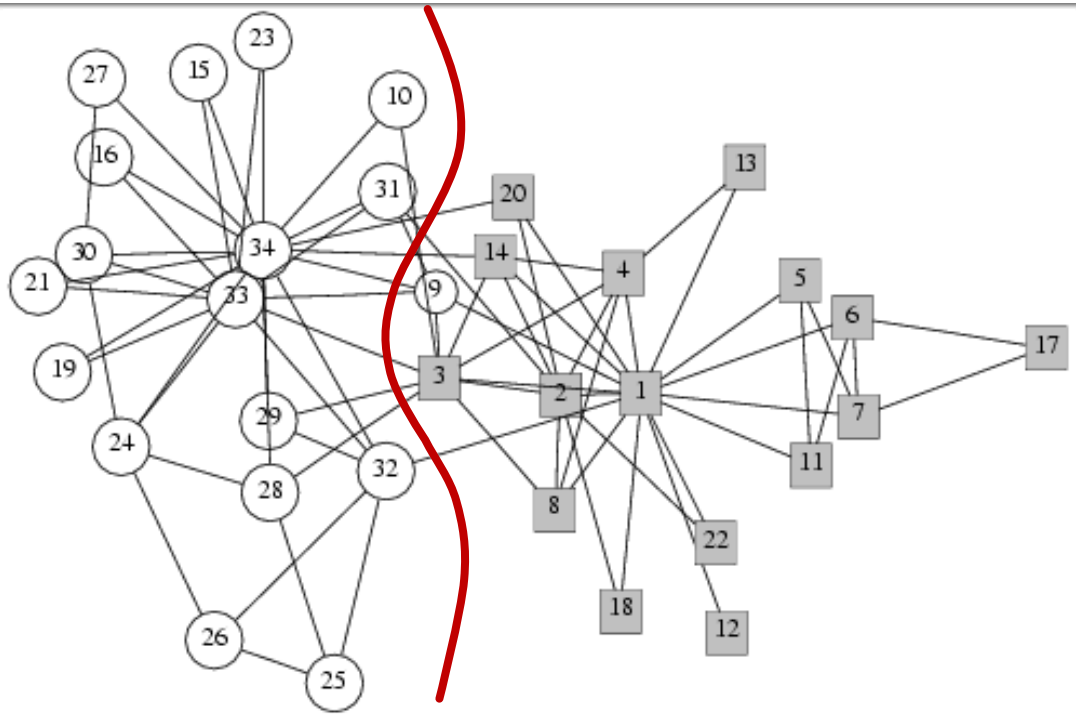
Finding network communities

- How to automatically find such densely connected groups of nodes?
- Ideally such automatically detected clusters would then correspond to real groups
- For example:



Communities, clusters, groups, modules

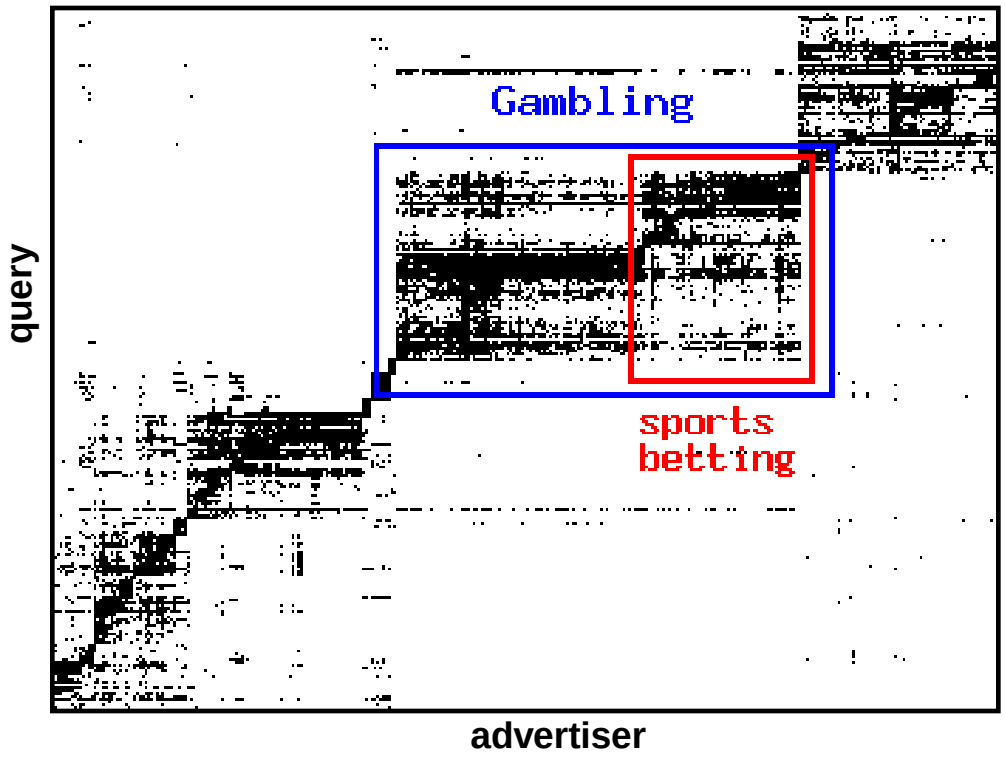
Social Network Data



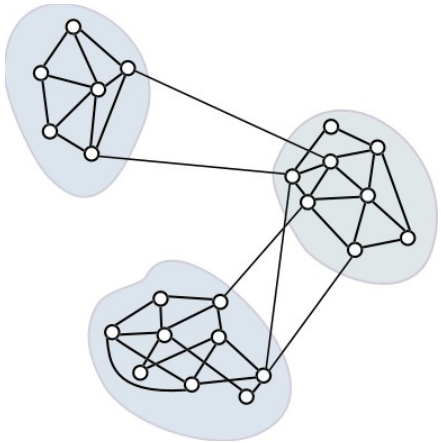
- Zachary's Karate club network:
  - Observe social ties and rivalries in a university karate club
  - During his observation, conflicts led the group to split
  - Split could be explained by a minimum cut in the network

Micro-markets in sponsored search

Find micro-markets by partitioning the "query x advertiser" graph:



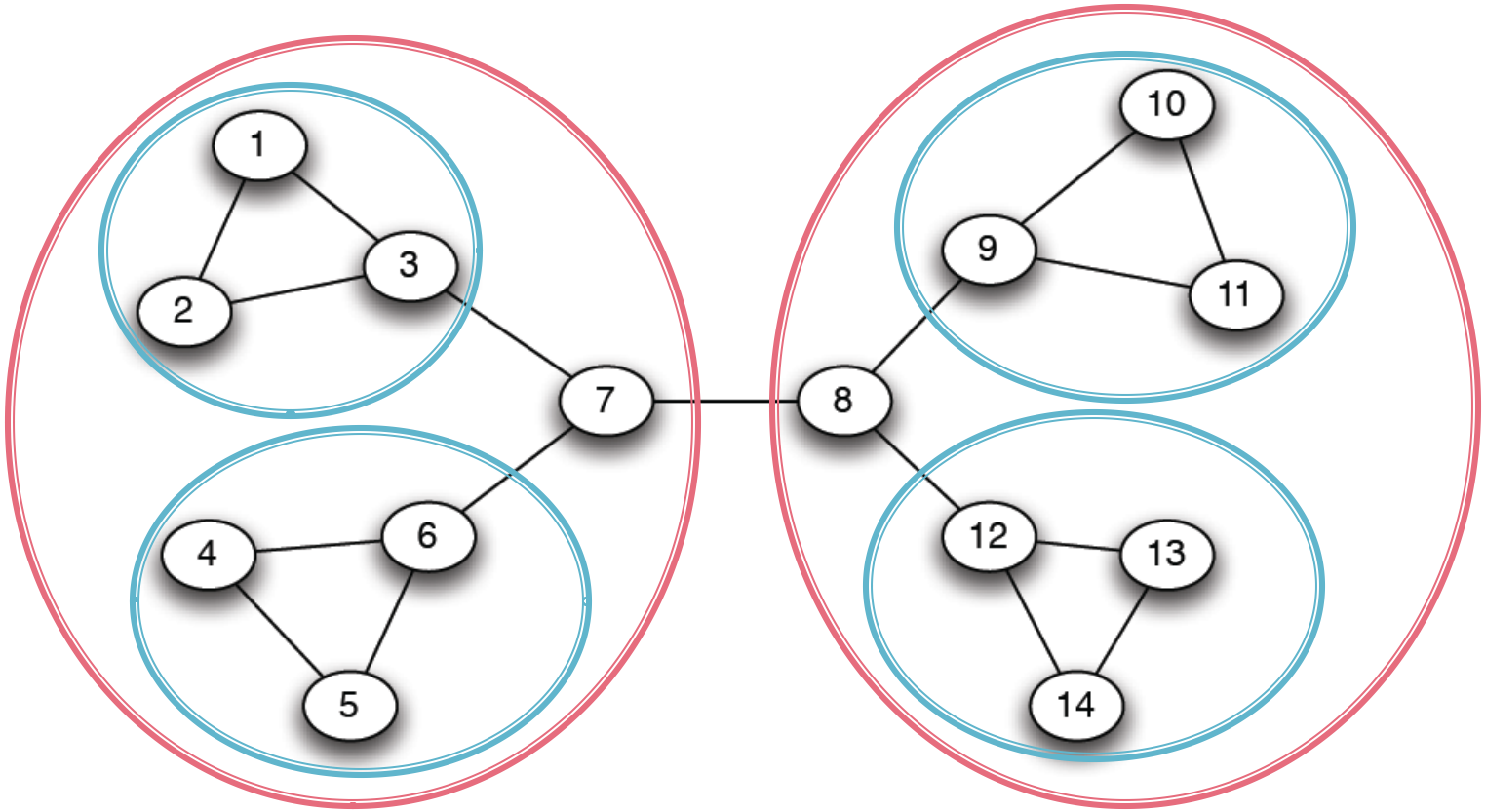
Clustering and Community Finding



Many methods:

- Linear (low-rank) methods:
  - If Gaussian, then low-rank space is good
- Kernel (non-linear) methods:
  - If low-dimensional manifold, then kernels are good
- Hierarchical methods:
  - Top-down and bottom-up – common in social sciences
- Graph partitioning methods:
  - Define "edge counting" metric – conductance, expansion, modularity, etc. – and optimize!

Hierarchically nested communites



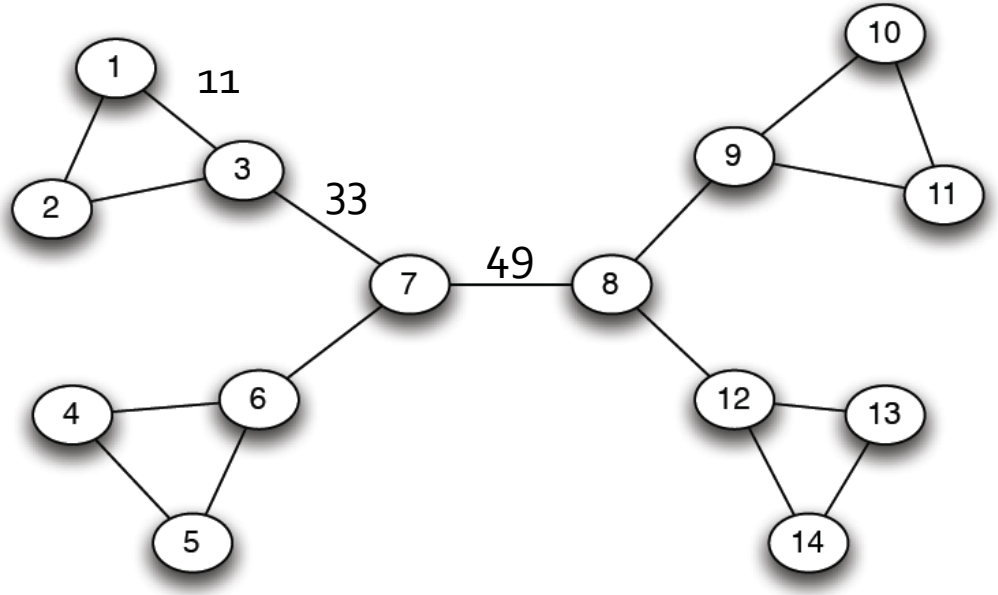
What is a good notion that would extract such clusters?

Algorithm of Girvan-Newman

- Divisive hierarchical clustering based on the notion of edge betweenness:

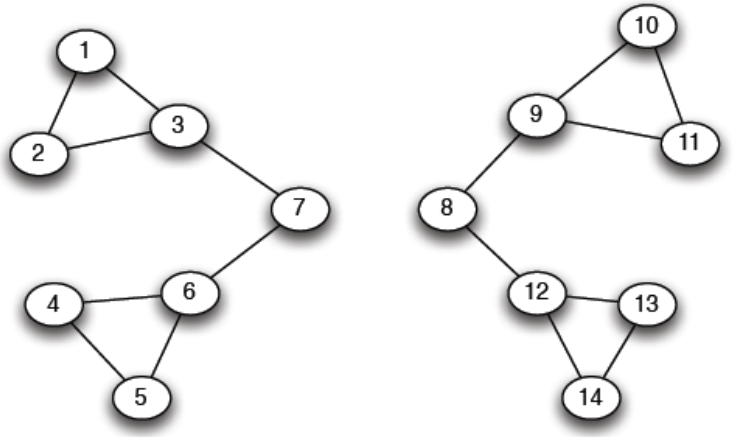
  Number of shortest paths passing through the edge

- Remove edges in decreasing betweenness

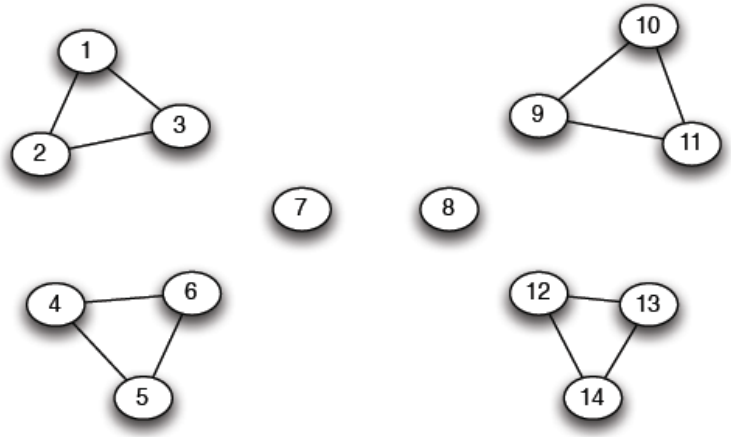


[Girvan-Newman PNAS '02]

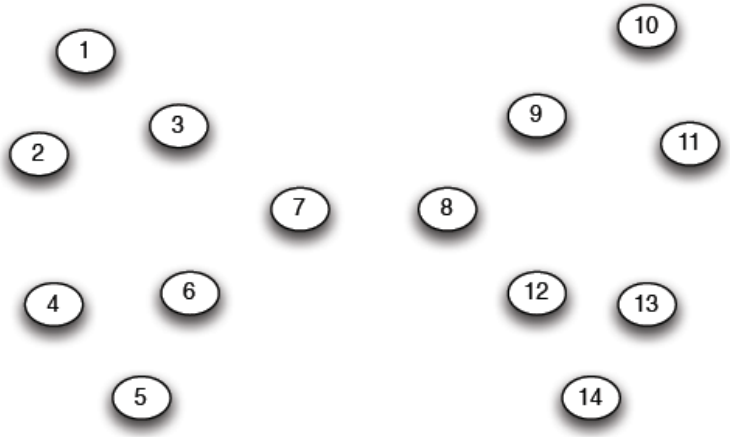
Algorithm of Girvan-Newman



(a) *Step 1*



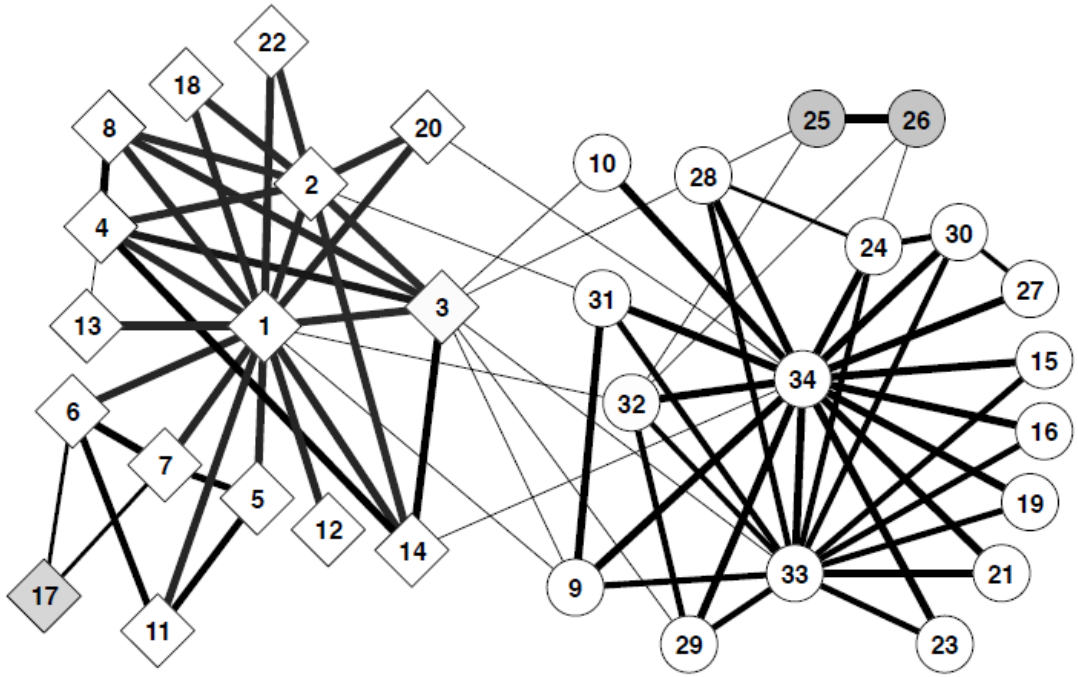
(b) *Step 2*

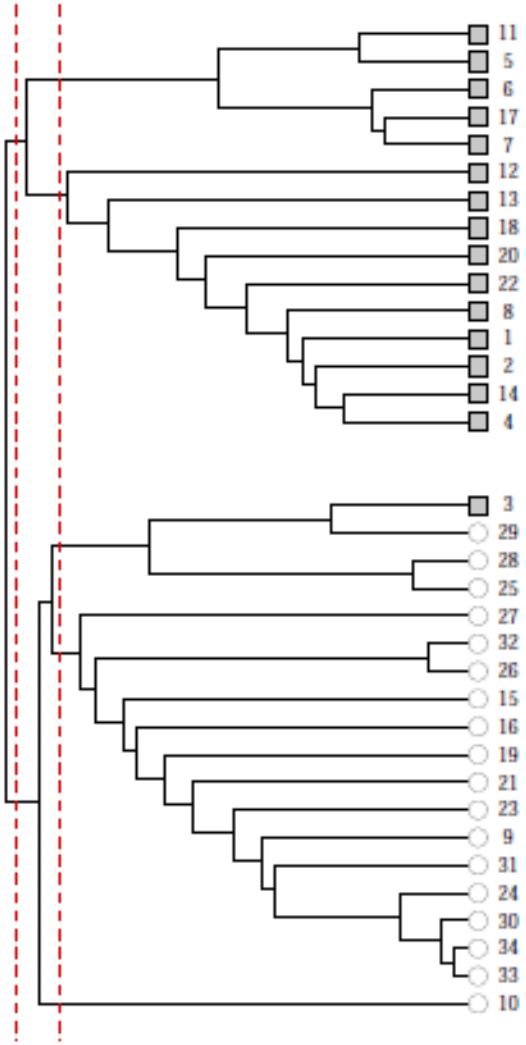


(c) *Step 3*

Girvan-Newman: Results

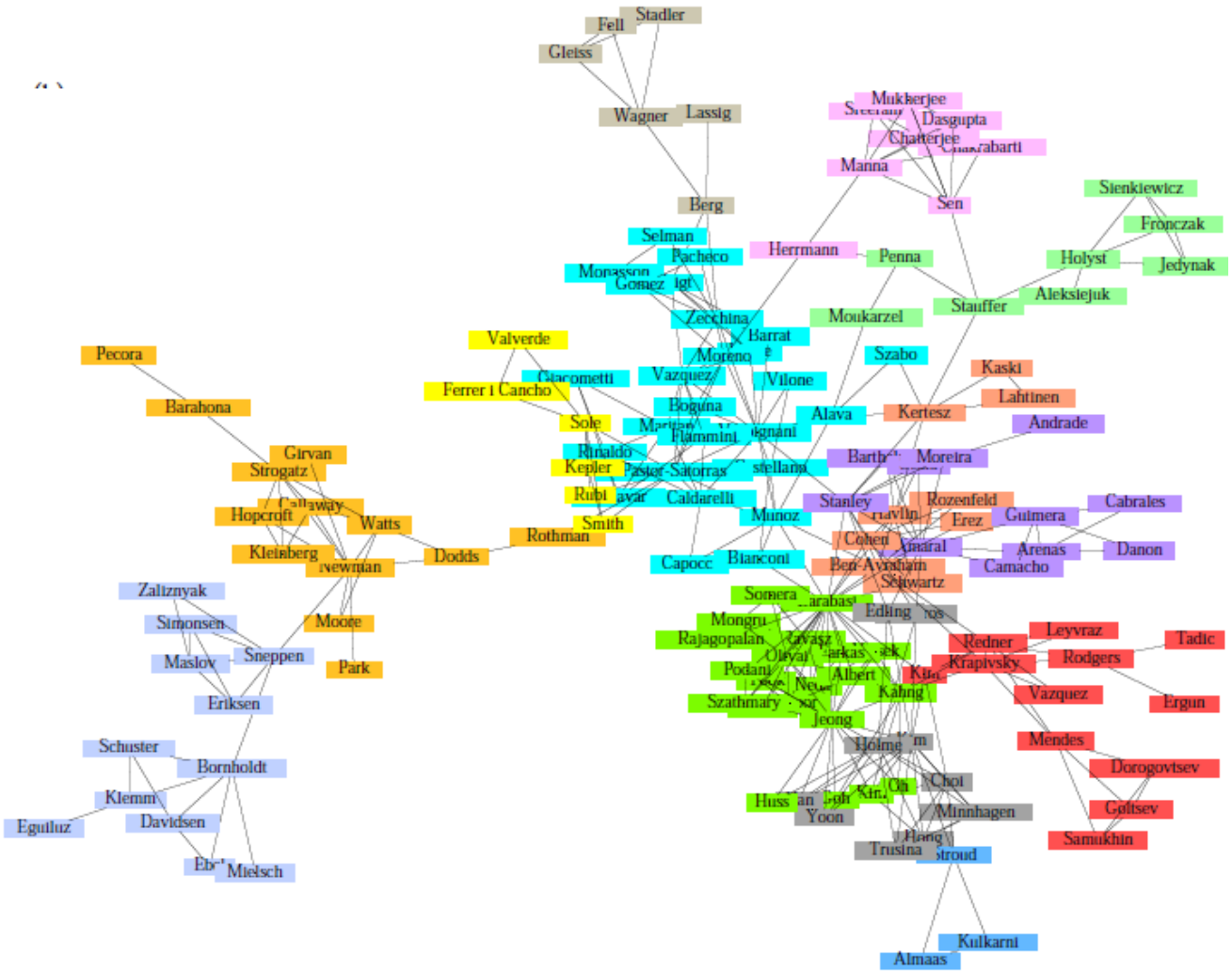
- Zachary's Karate club: hierarchic decomposition





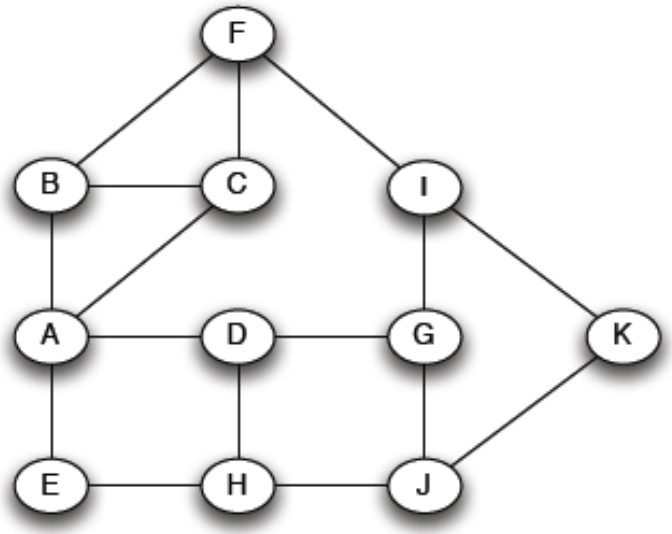
[Newman-Girvan PhysRevE '03]

Girvan-Newman: Results



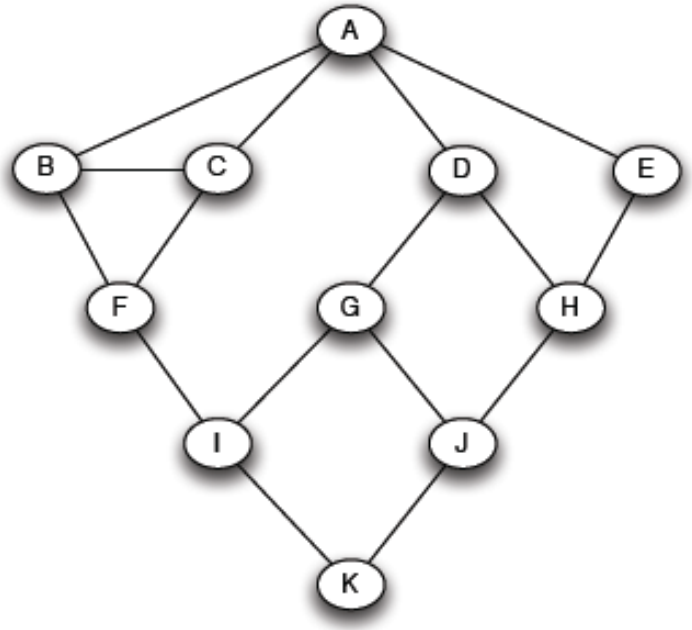
Communities in physics collaborations

How to compute betweenness (1)



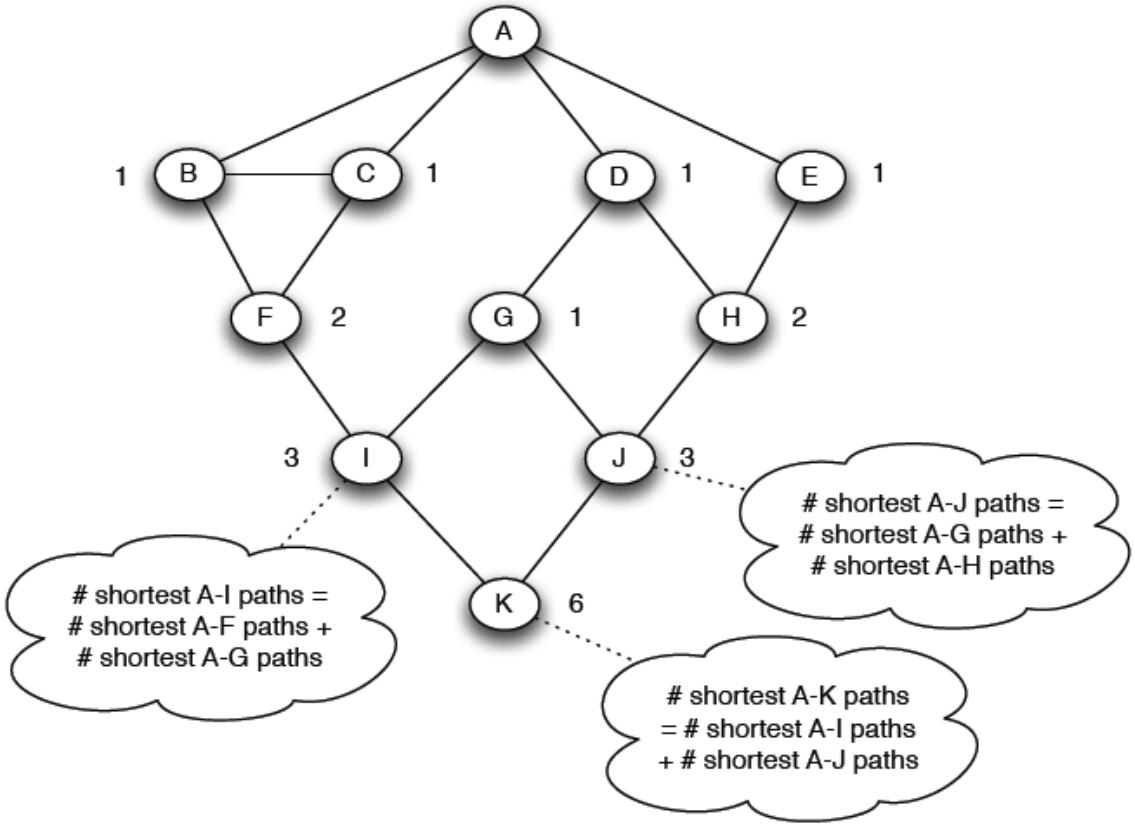
- Want to compute betweenness of paths starting at node A

- Breath first search starting from A:



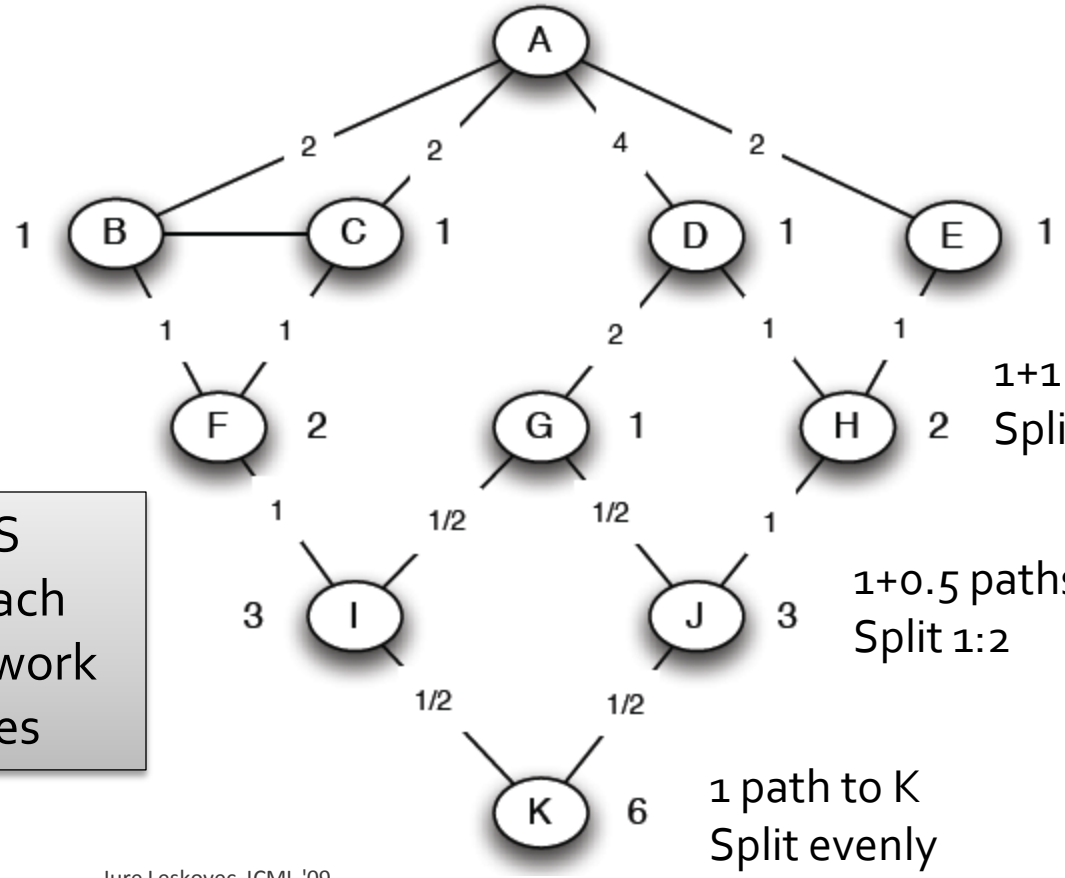
How to compute betweenness (2)

- Count the number of shortest paths from A to all other nodes of the network:



How to compute betweenness (3)

- Compute betweenness by working up the tree: If there are multiple paths count them fractionally

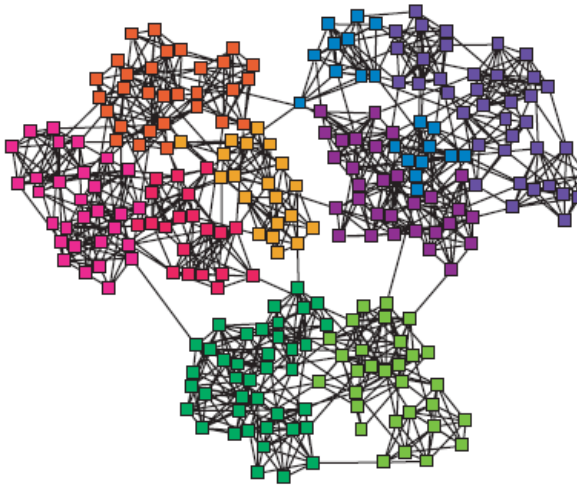


- Repeat the BFS procedure for each node of the network
- Add edge scores

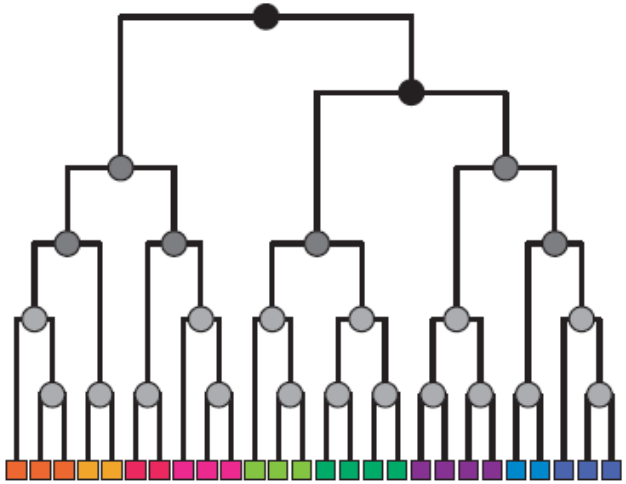
[Clauset et al. Nature '08]

Hierarchical random graphs

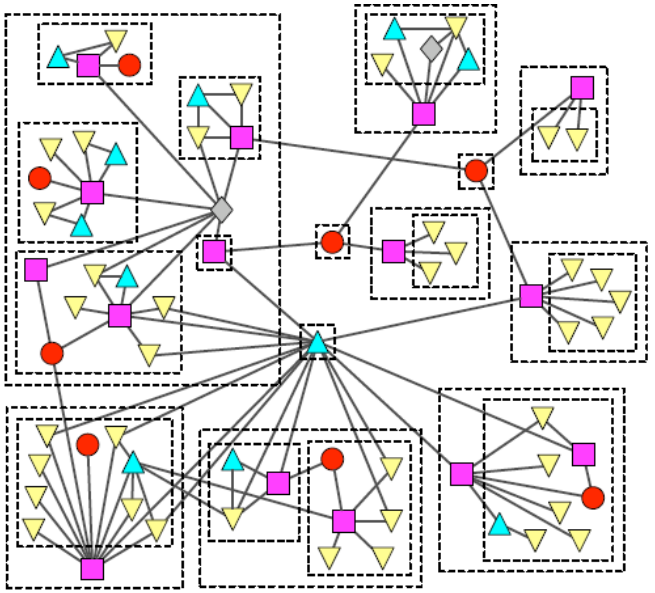
- Hierarchical random graphs can be used to extract hierarchical community structure



Simple network



Corresponding dendrogram

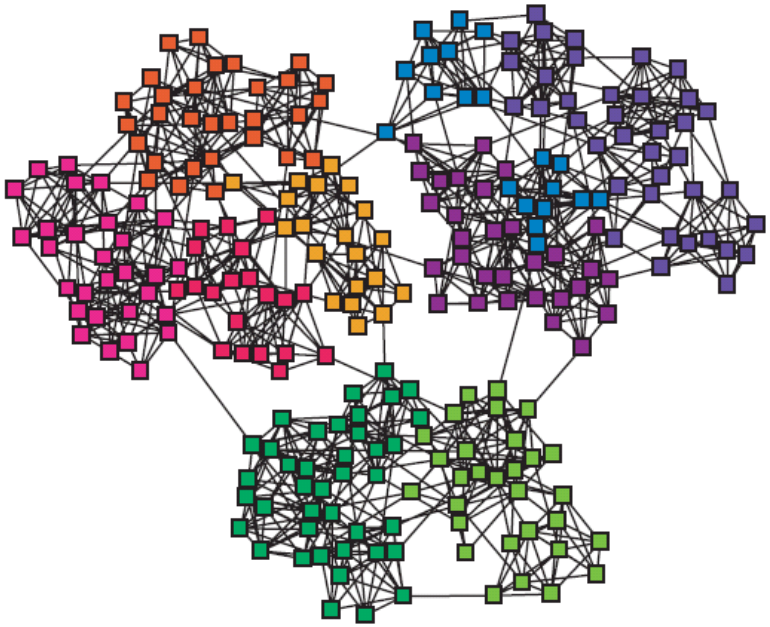


Grassland species network

Node shapes: plants, herbivores, parasitoids and hyperparasitoids

Network communities

- **Communities:**
  - Sets of nodes with **lots** of connections **inside** and **few** to **outside** (the rest of the network)

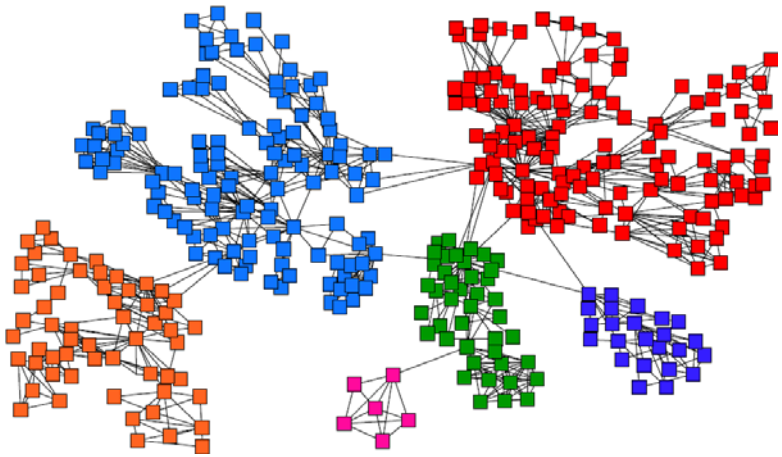


Hierarchical community structure

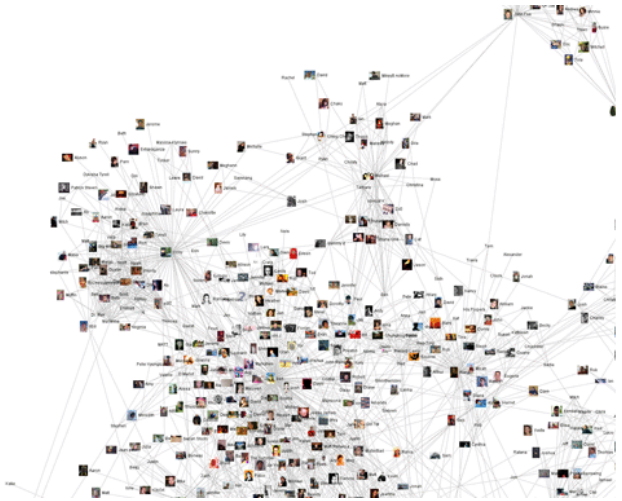
Question:
Are large networks really like this?

Small vs. Large networks

- Community (cluster) structure of networks



Physics collaborations



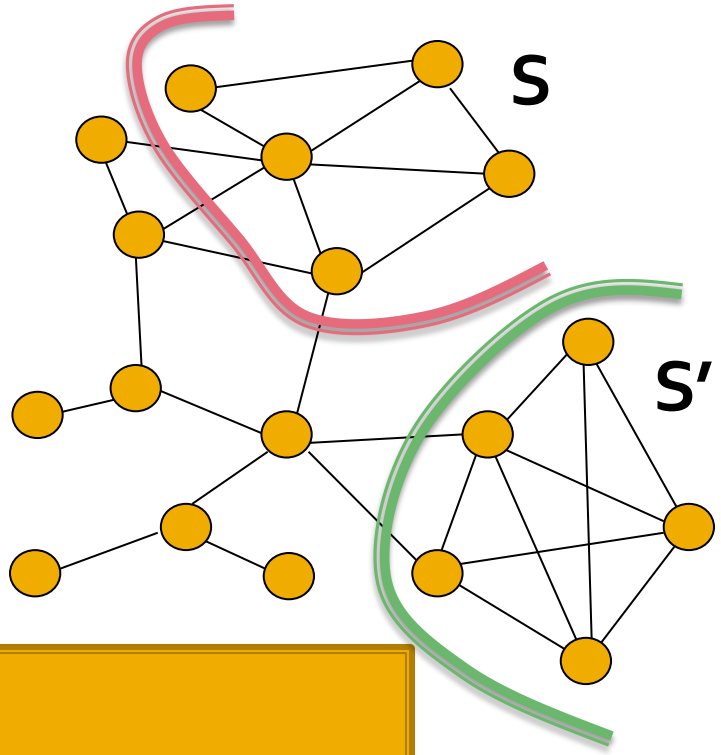
Tiny part of a large social network

How does community structure scale from small to large networks?

[Leskovec et al. WWW '08]

How expressed are communities?

- How community-like is a set of nodes?
- How good of a community is a set of nodes?



Conductance (normalized cut):

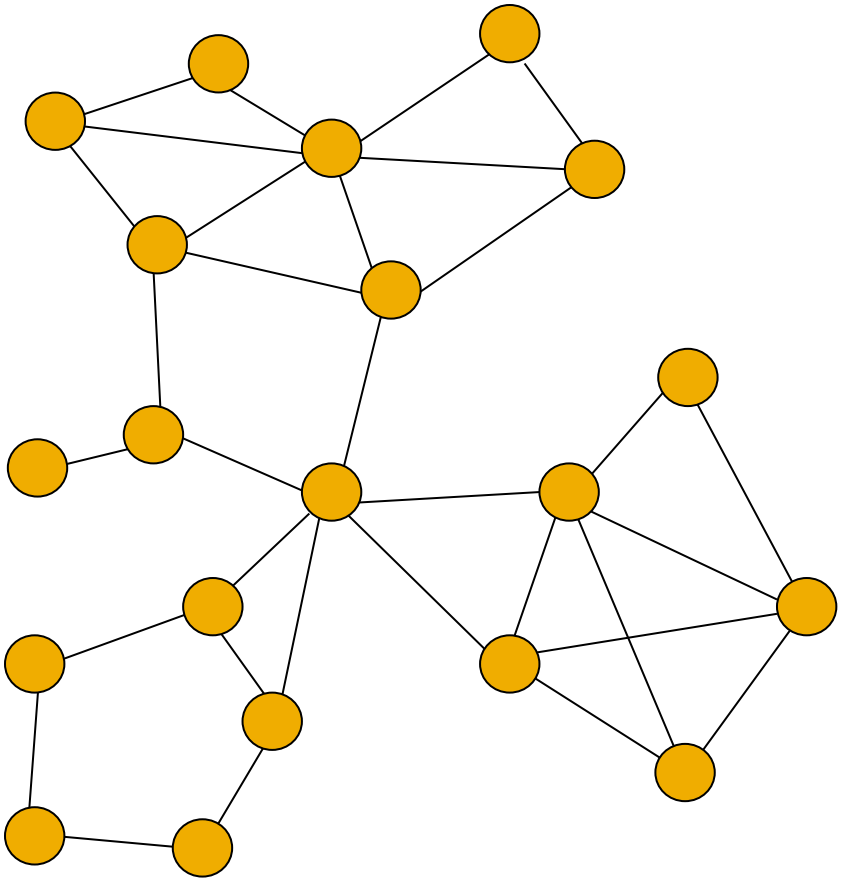
$$\phi(S) = \frac{\sum_{i \in S, j \notin S} A_{ij}}{\min\{A(S), A(\overline{S})\}}$$

$$A(S) = \sum_{i \in S} \sum_{j \in V} A_{ij}$$

- Small **Φ(S) ==** more community-like sets of nodes

Community score (quality)

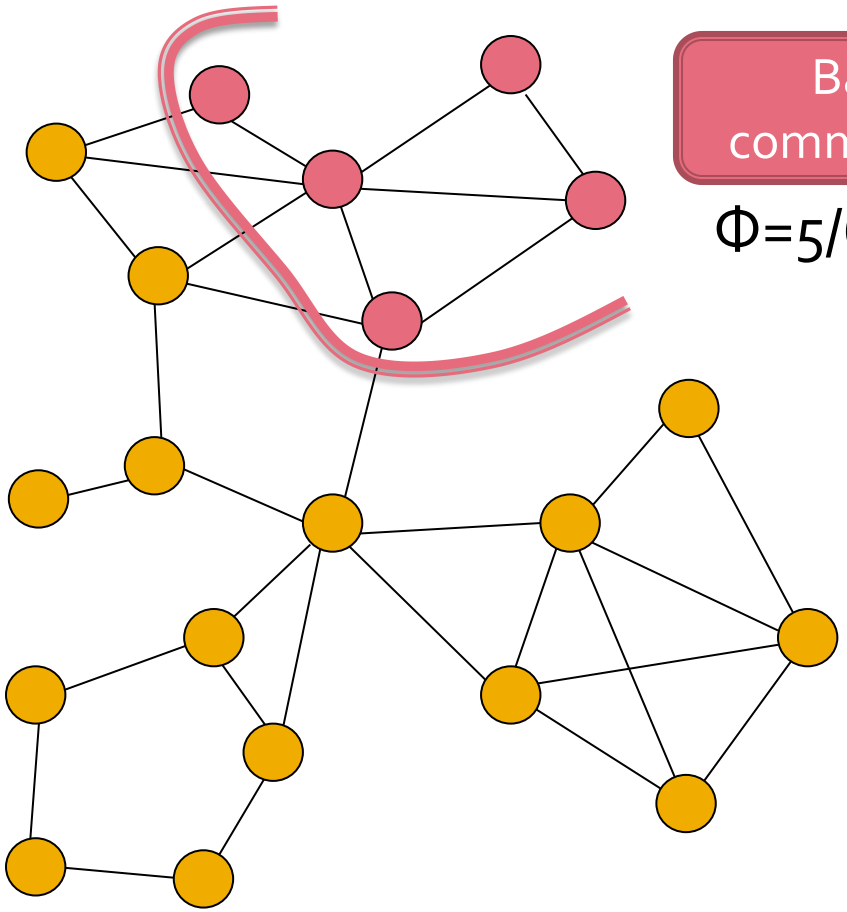
What is "best" community of 5 nodes?



Score: $\Phi(S)$ = # edges cut / # edges inside

Community score (quality)

What is "best" community of 5 nodes?



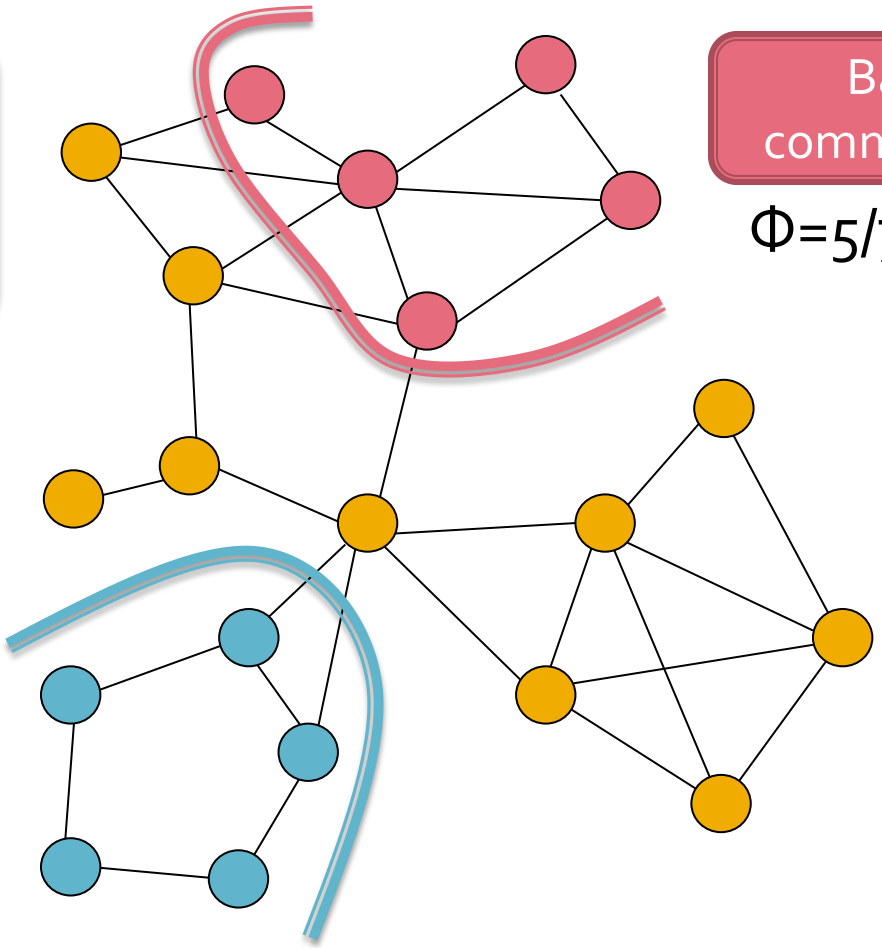
Bad community

Φ=5/6 = 0.83

Score: Φ(S) = # edges cut / # edges inside

Community score (quality)

What is "best" community of 5 nodes?



Bad community

Φ=5/7 = 0.7

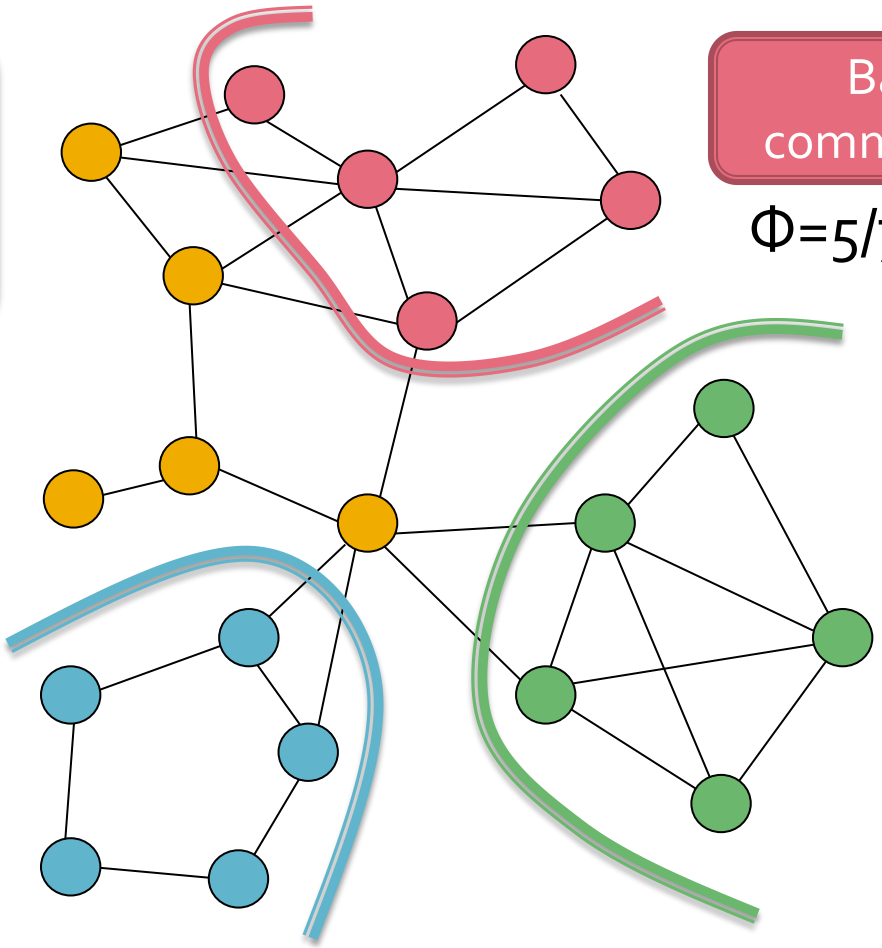
Better community

Φ=2/5 = 0.4

Score: Φ(S) = # edges cut / # edges inside

Community score (quality)

What is "best" community of 5 nodes?



Bad community

Φ=5/7 = 0.7

Best community

Φ=2/8 = 0.25

Better community

Φ=2/5 = 0.4

Score: Φ(S) = # edges cut / # edges inside

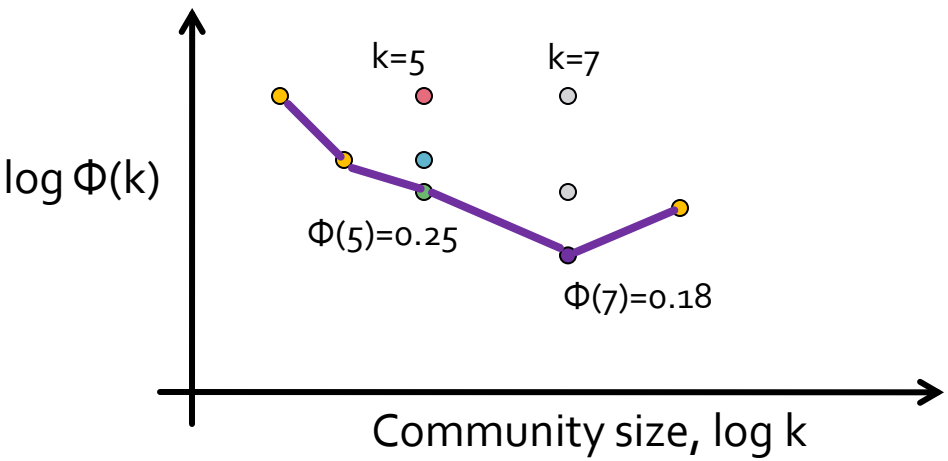
[Leskovec et al. 08]

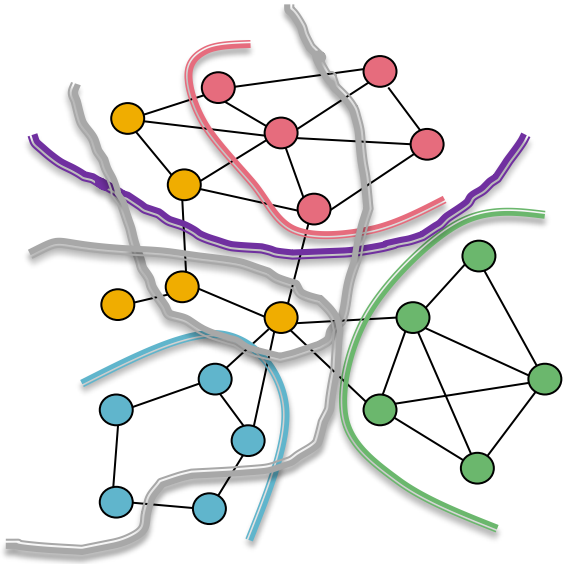
Network Community Profile Plot

- Define:
  Network community profile (**NCP**) plot

  Plot the score of best community of size *k*

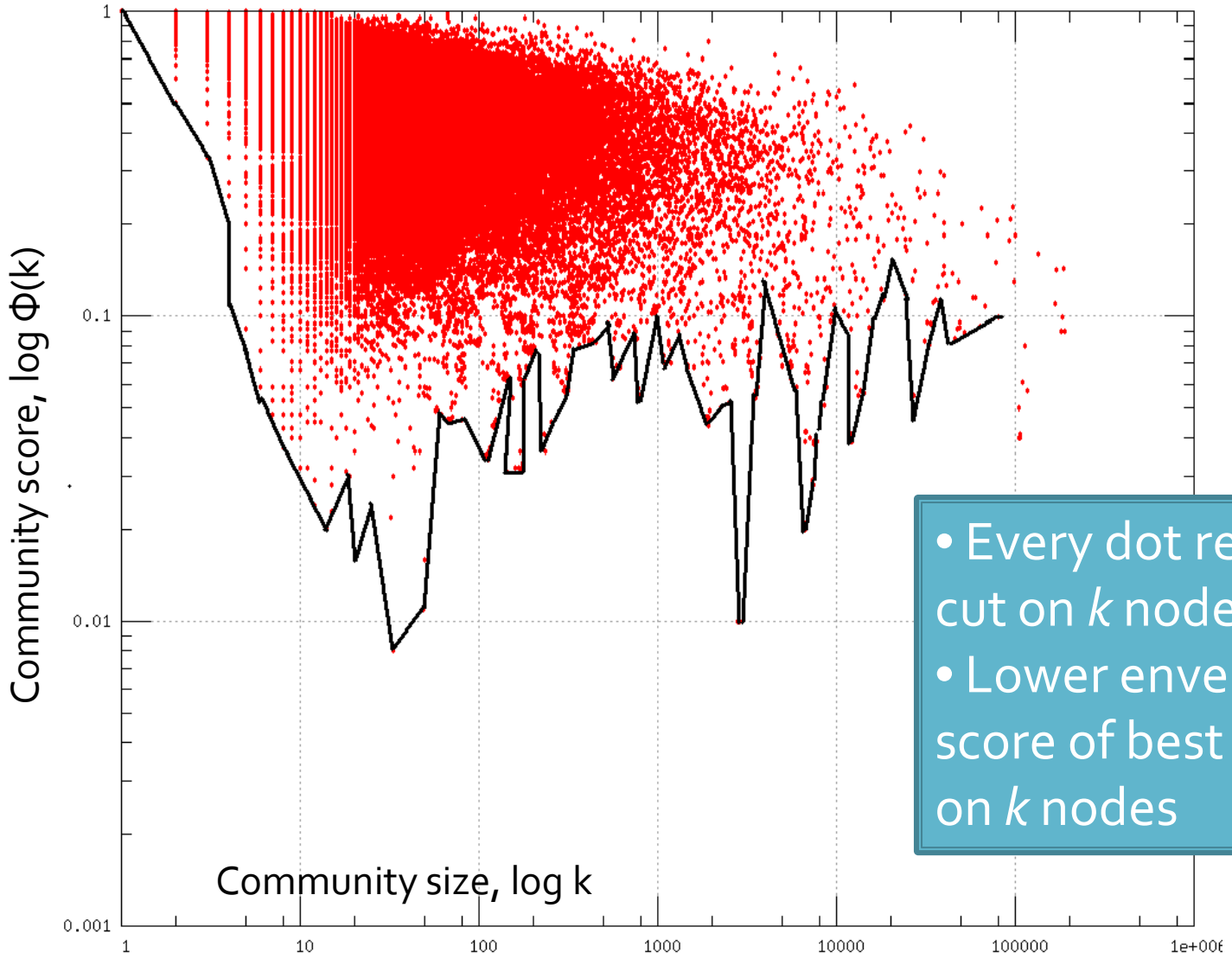
$$\Phi(k) = \min_{S \subset V, |S| = k} \phi(S)$$





[Leskovec et al. WWW '08]

Network Community Profile Plot



- Every dot represents a cut on *k* nodes
- Lower envelope gives score of best community on *k* nodes

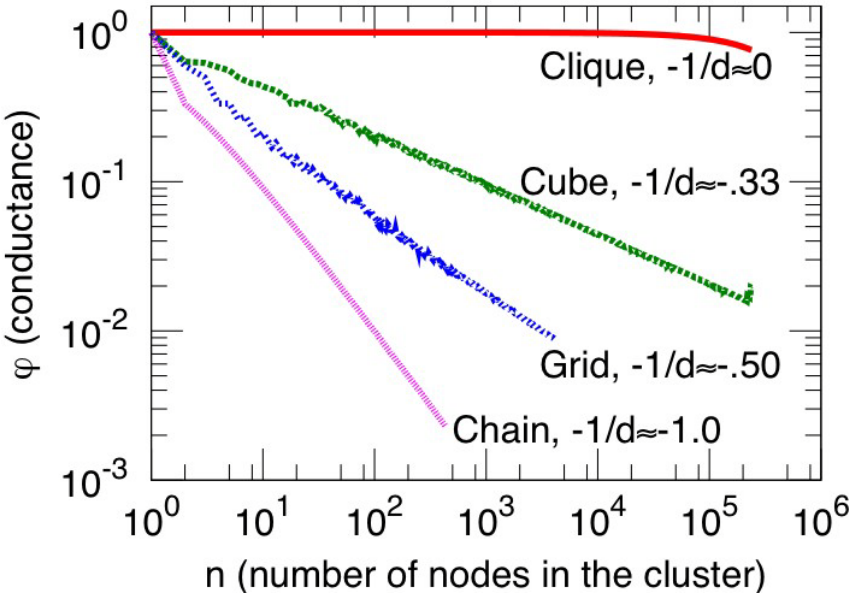
[Leskovec et al. WWW '08]

Probing networks

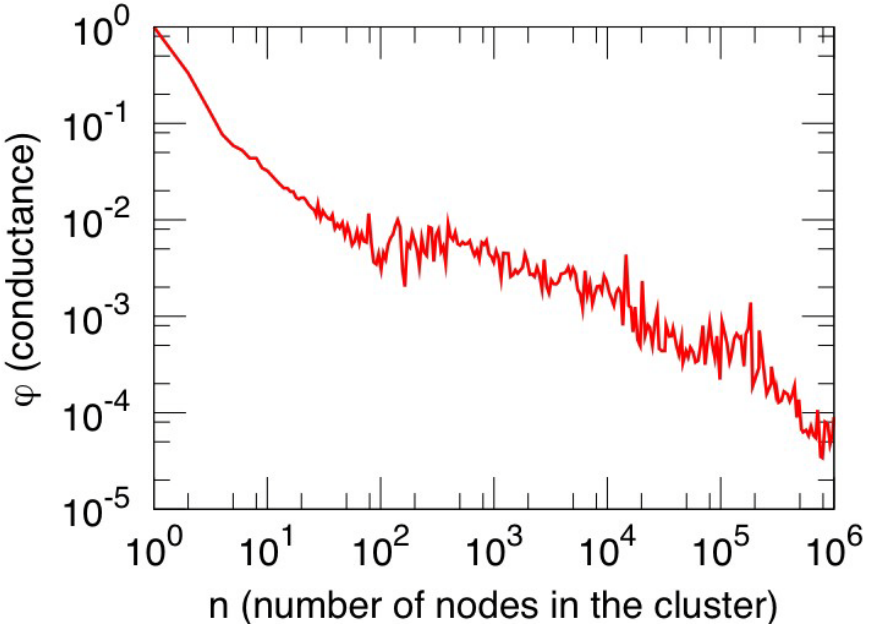
- **Idea:** Use approximation algorithms for NP-hard graph partitioning problems as experimental probes of network structure.
  - Spectral (quadratic approx): confuses "long paths" with "deep cuts"
  - Multi-commodity flow (*log(n)* approx): difficulty with expanders
  - SDP (*sqrt(log(n))* approx): best in theory
  - Metis (multi-resolution heuristic): common in practice
  - X+MQI: post-processing step on, *e.g.*, MQI of Metis
- Local Spectral - connected and tighter sets (empirically)
- Metis+MQI - best conductance (empirically)

[Leskovec et al. WWW '08]

NCP plot: Meshes



d-dimensional meshes

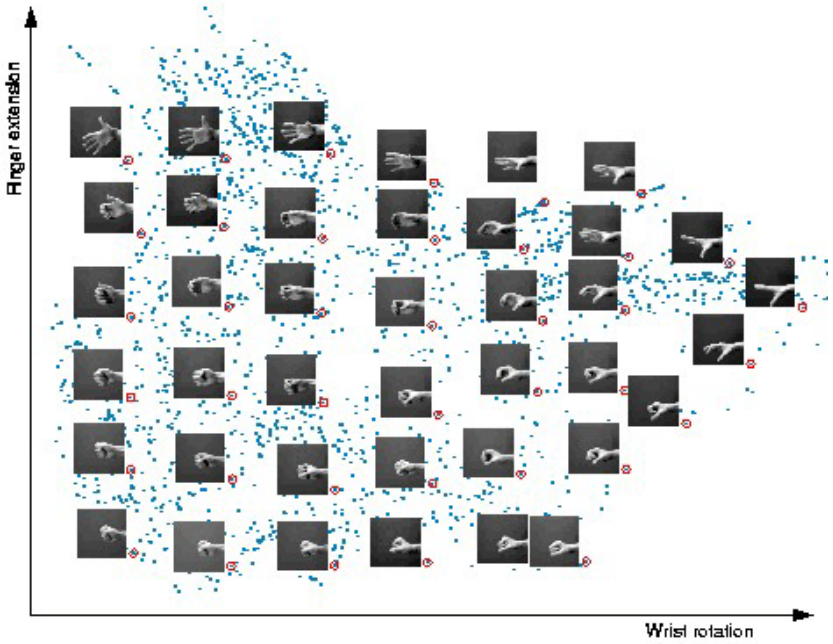


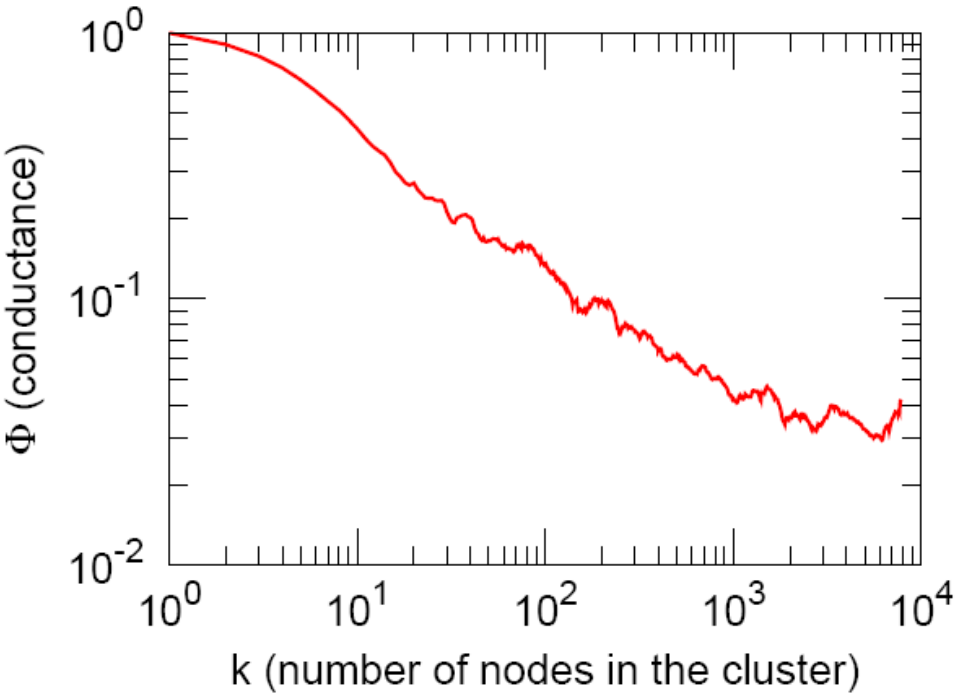
California road network

[Leskovec et al. WWW '08]

NCP plot: Manifold dataset

- Manifold learning dataset (Hands)

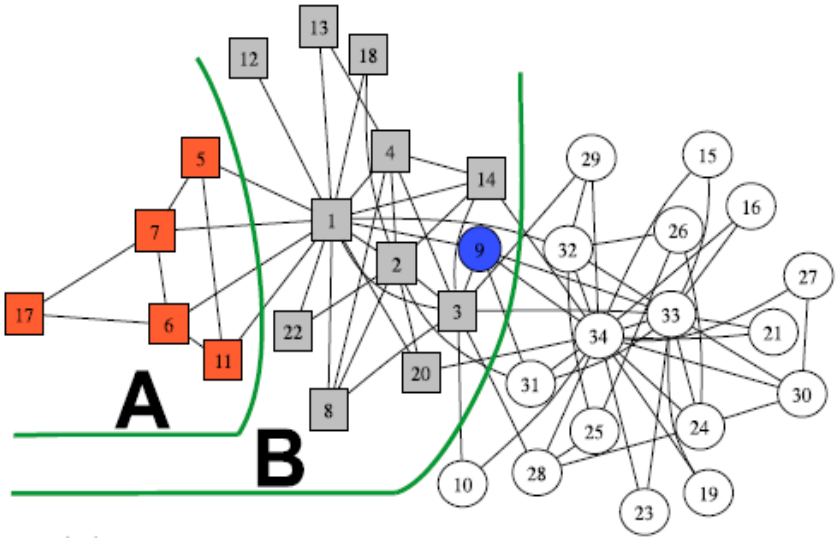


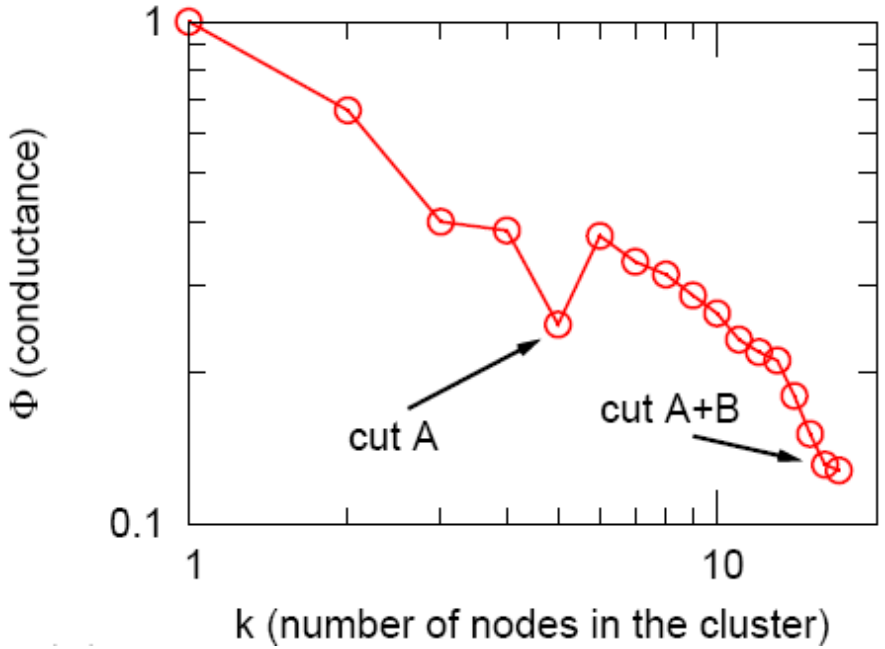


[Leskovec et al. WWW '08]

NCP plot: Zachary's karate club

- Zachary's university karate club social network

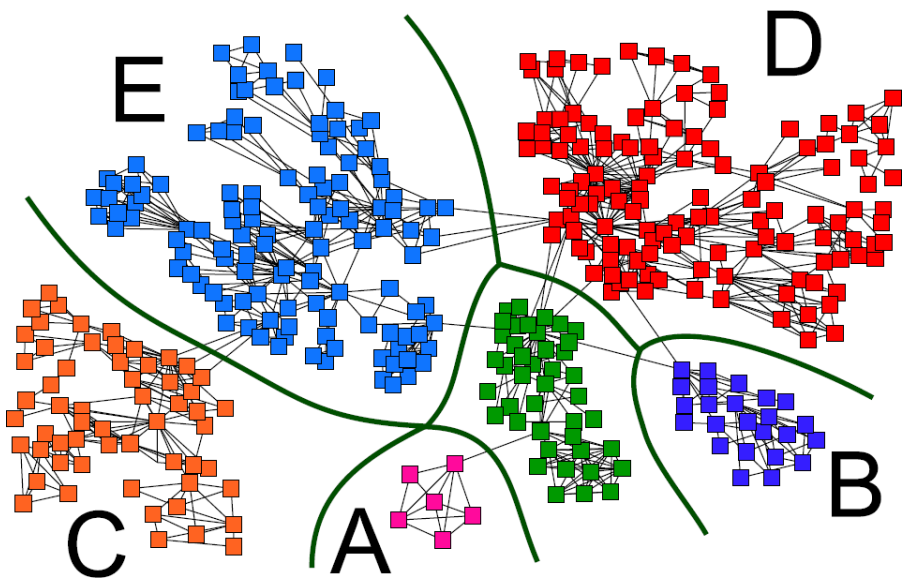


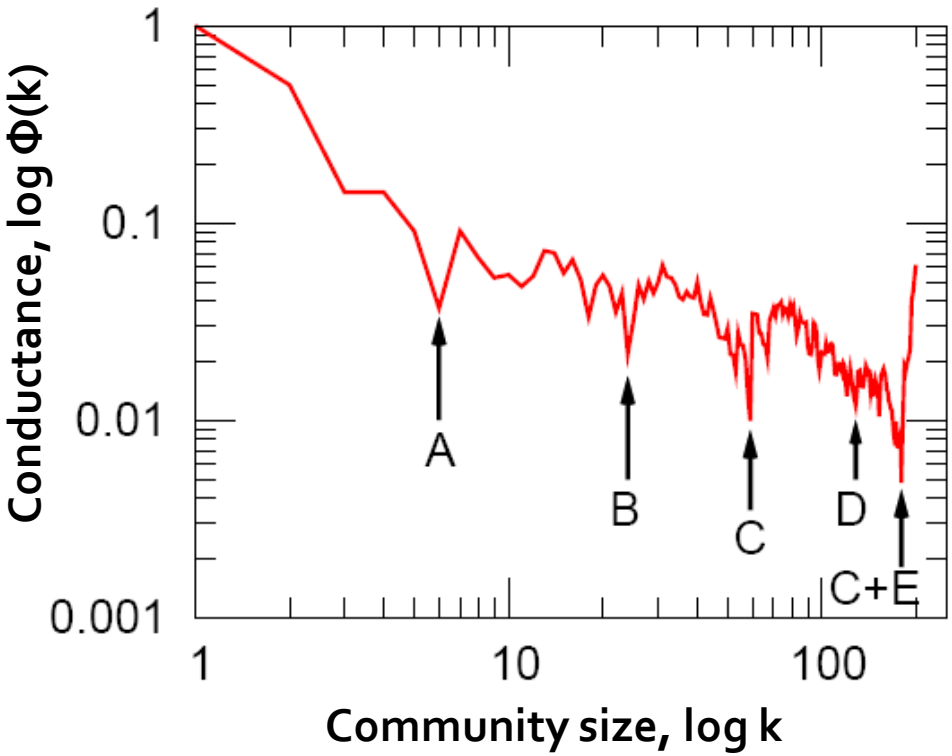


[Leskovec et al. WWW '08]

NCP plot: Network Science

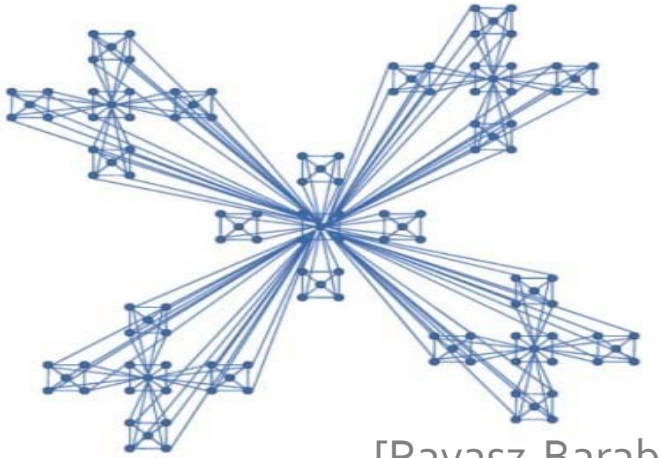
- Collaborations between scientists in Networks
  [Newman, 2005]



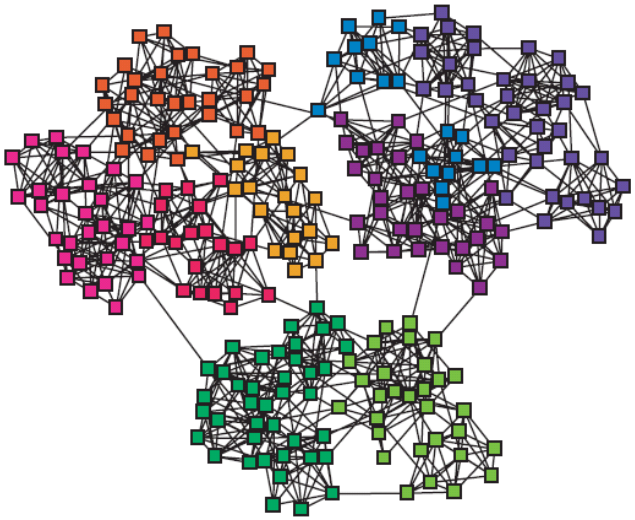


[Leskovec et al. WWW '08]

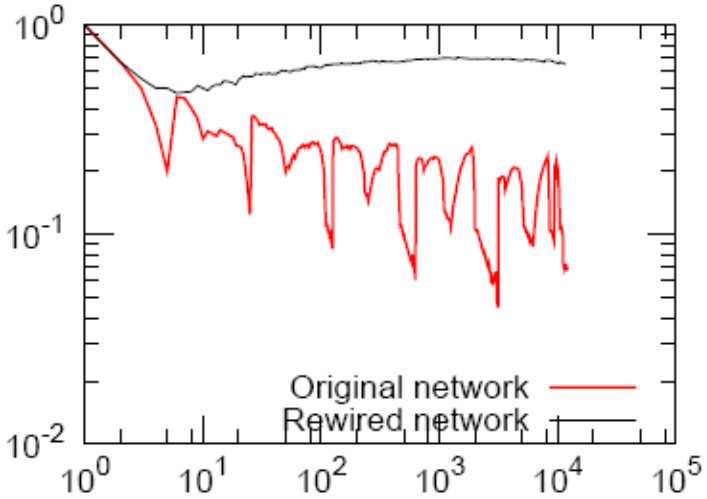
NCP plot: Hierarchical networks

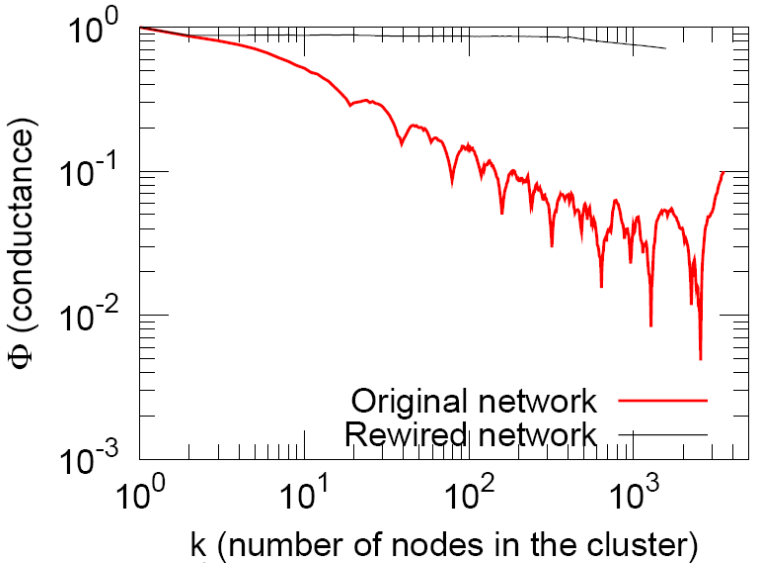


[Ravasz-Barabasi 03]



[Clauset-Moore-Newman 08]



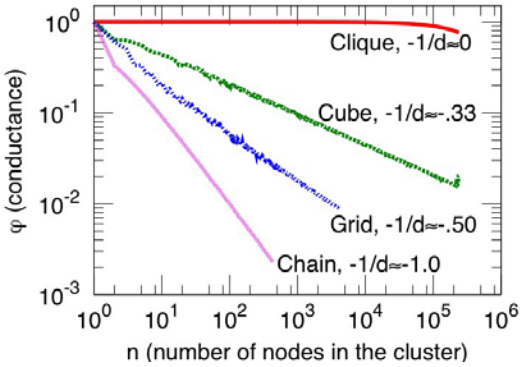


[Leskovec et al. WWW '08]

Natural hypothesis

Natural hypothesis about NCP:

- NCP of real networks slopes downward
- Slope of the NCP corresponds to the dimensionality of the network



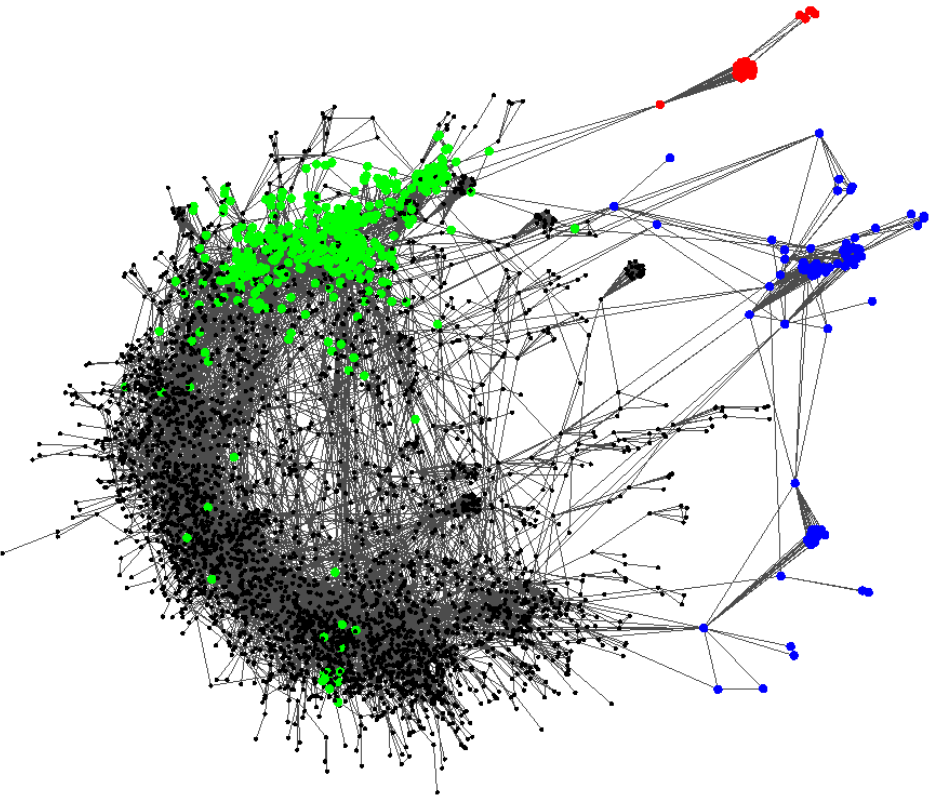
What about large networks?

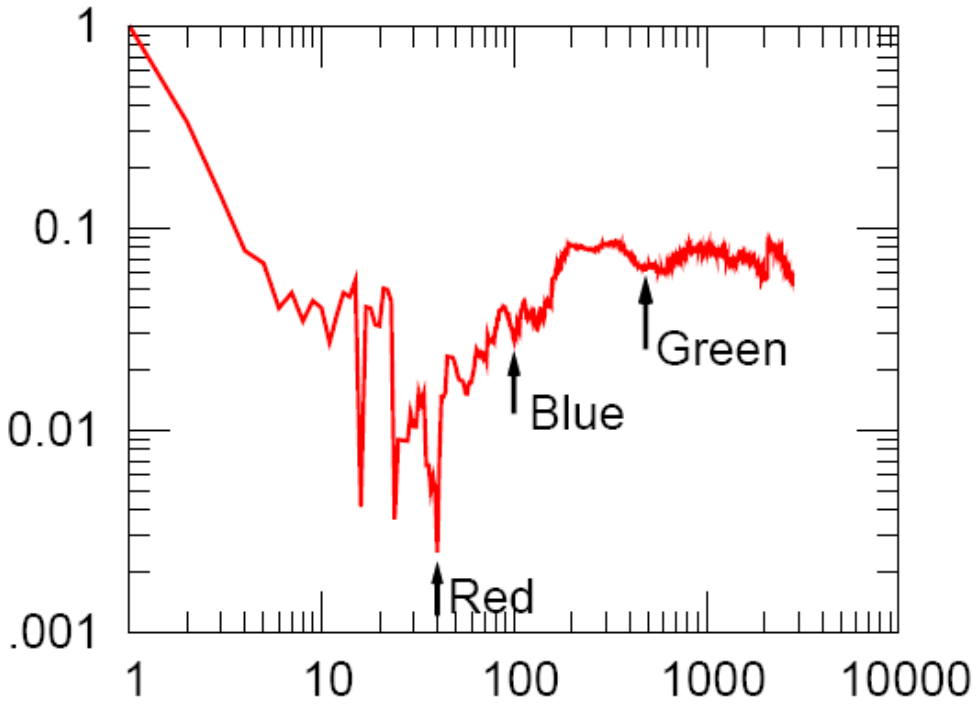
• Social nets	Nodes	Edges	Description
LIVEJOURNAL	4,843,953	42,845,684	Blog friendships [5]
EPINIONS	75,877	405,739	Trust network [28]
CA-DBLP	317,080	1,049,866	Co-authorship [5]
• Information (citation) networks			
CIT-HEP-TH	27,400	352,021	Arxiv hep-th [14]
AMAZONPROD	524,371	1,491,793	Amazon products [8]
• Web graphs			
WEB-GOOGLE	855,802	4,291,352	Google web graph
WEB-WT10G	1,458,316	6,225,033	TREC WT10G
• Bipartite affiliation (authors-to-papers) networks			
ATP-DBLP	615,678	944,456	DBLP [21]
ATM-IMDB	2,076,978	5,847,693	Actors-to-movies
• Internet networks			
ASSKITTER	1,719,037	12,814,089	Autonom. sys.
GNUTELLA	62,561	147,878	P2P network [29]

[Leskovec et al. WWW '08]

Large networks: Very different

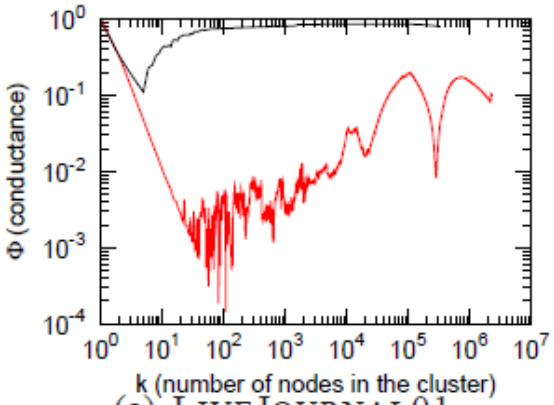
Typical example: General Relativity collaborations (n=4,158, m=13,422)



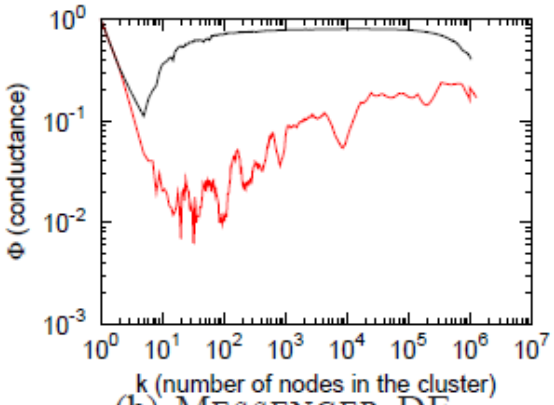


[Leskovec et al. WWW '08]

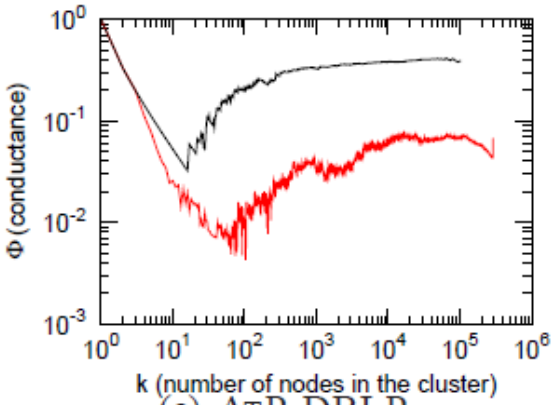
More NCP plots of networks



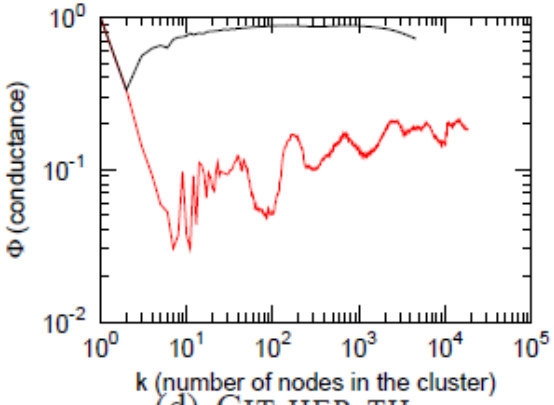
(a) LIVEJOURNAL01



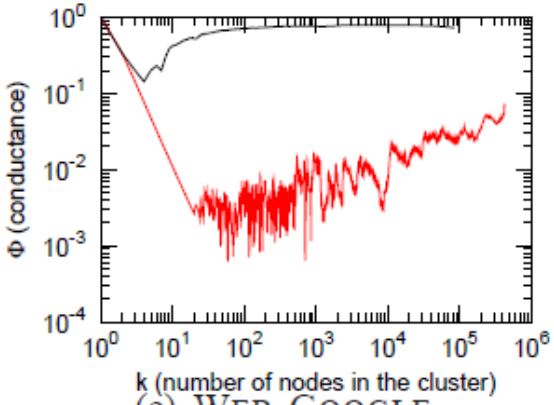
(b) MESSENGER-DE



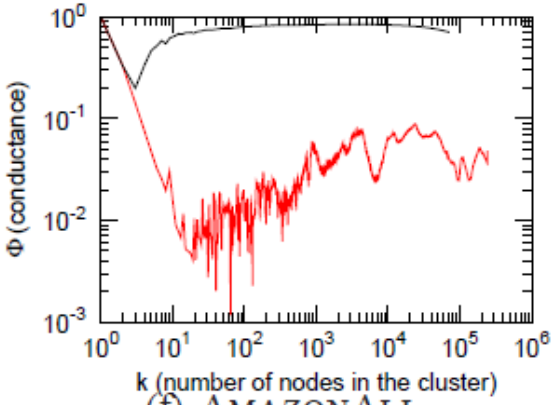
(c) ATP-DBLP



(d) CIT-HEP-TH



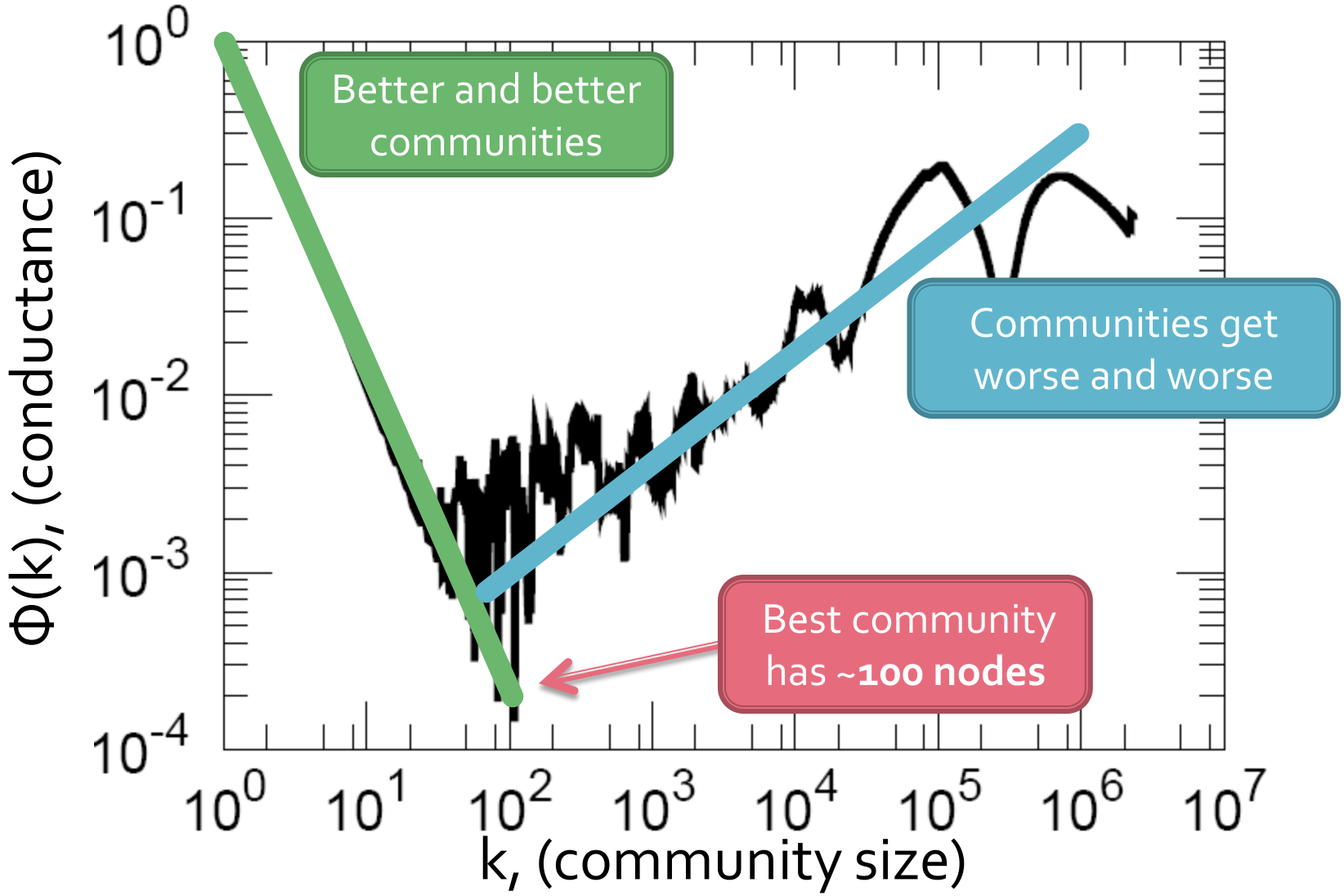
(e) WEB-GOOGLE



(f) AMAZONALL

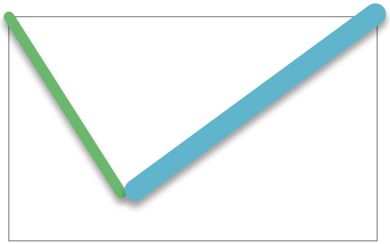
[Leskovec et al. WWW '08]

NCP: LiveJournal (n=5m, e=42m)

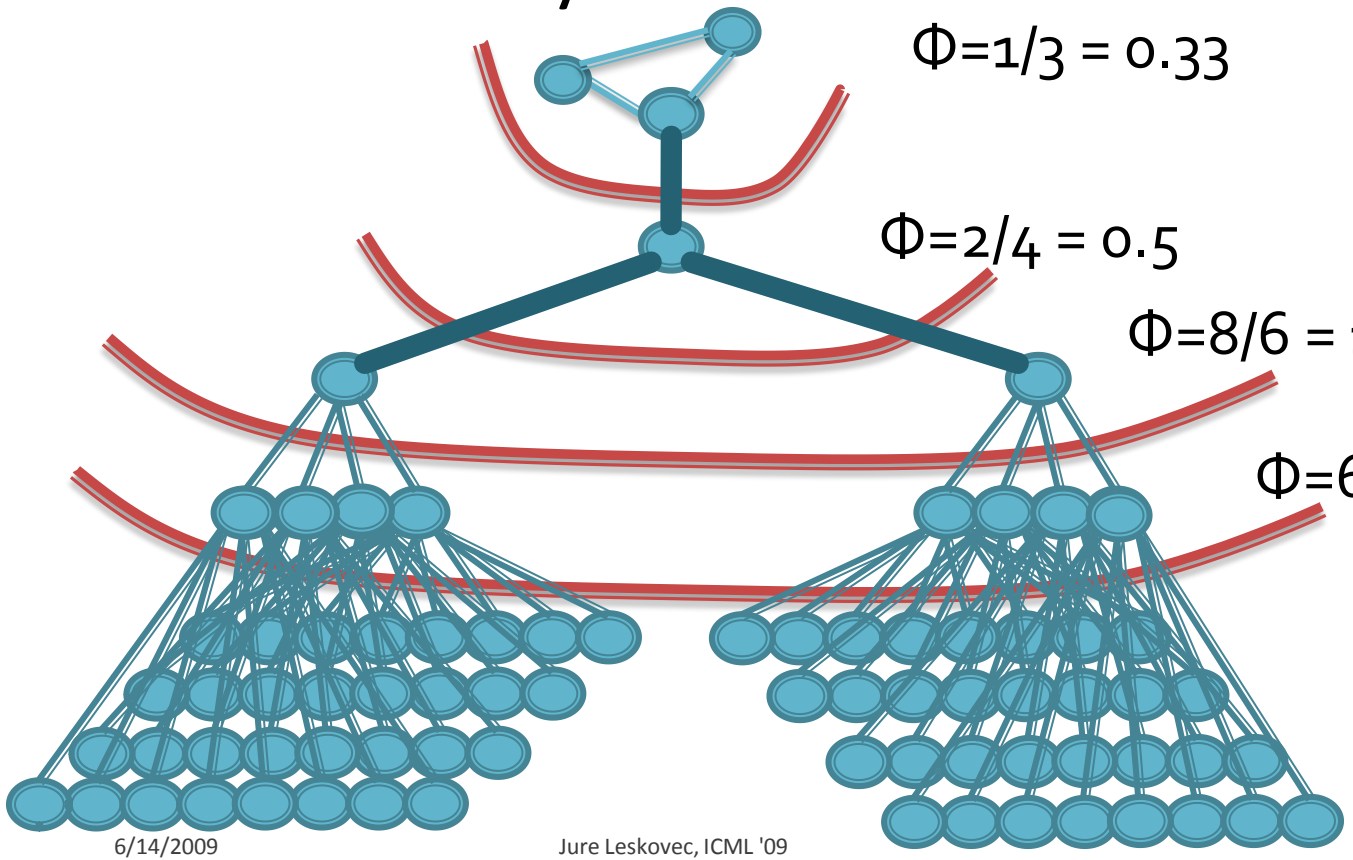


Explanation: Upward part

- Each new edge inside the community costs **more**



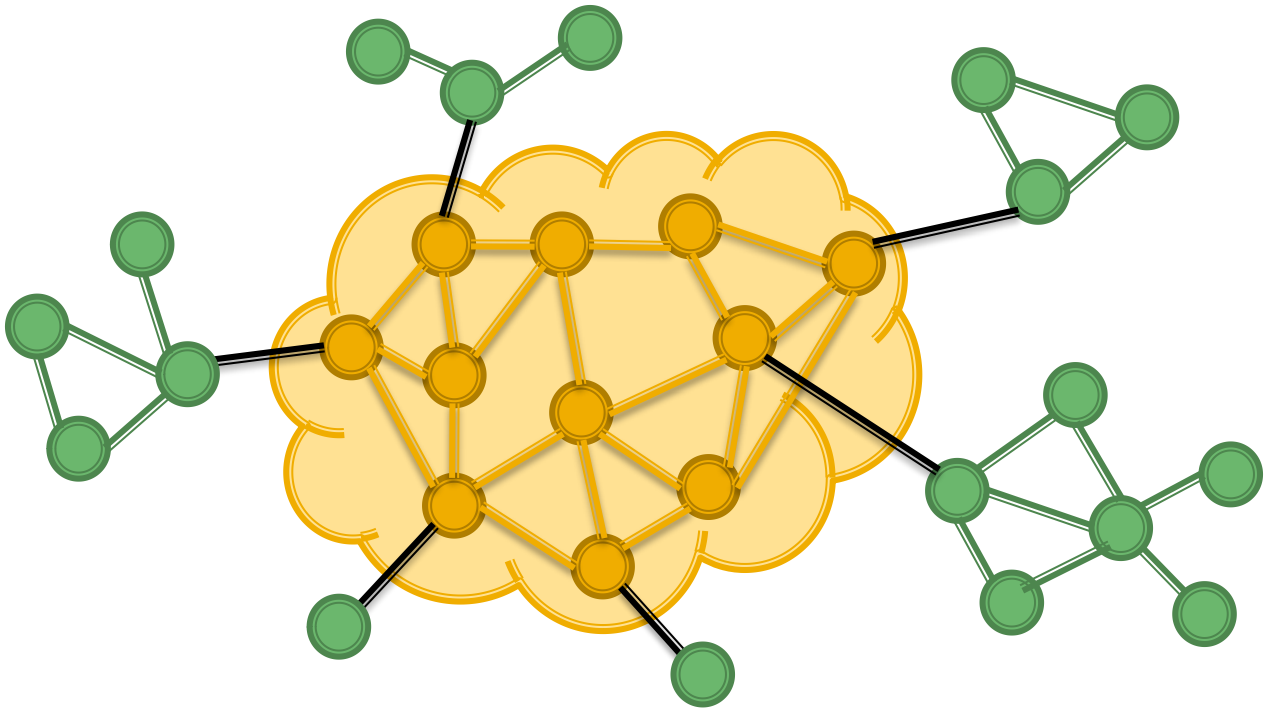
NCP plot

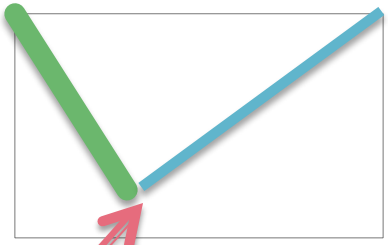


Each node has twice as many children

Explanation: Downward part

- Definition: **Whisker** is a maximal set of nodes connected to the network by a **single** edge



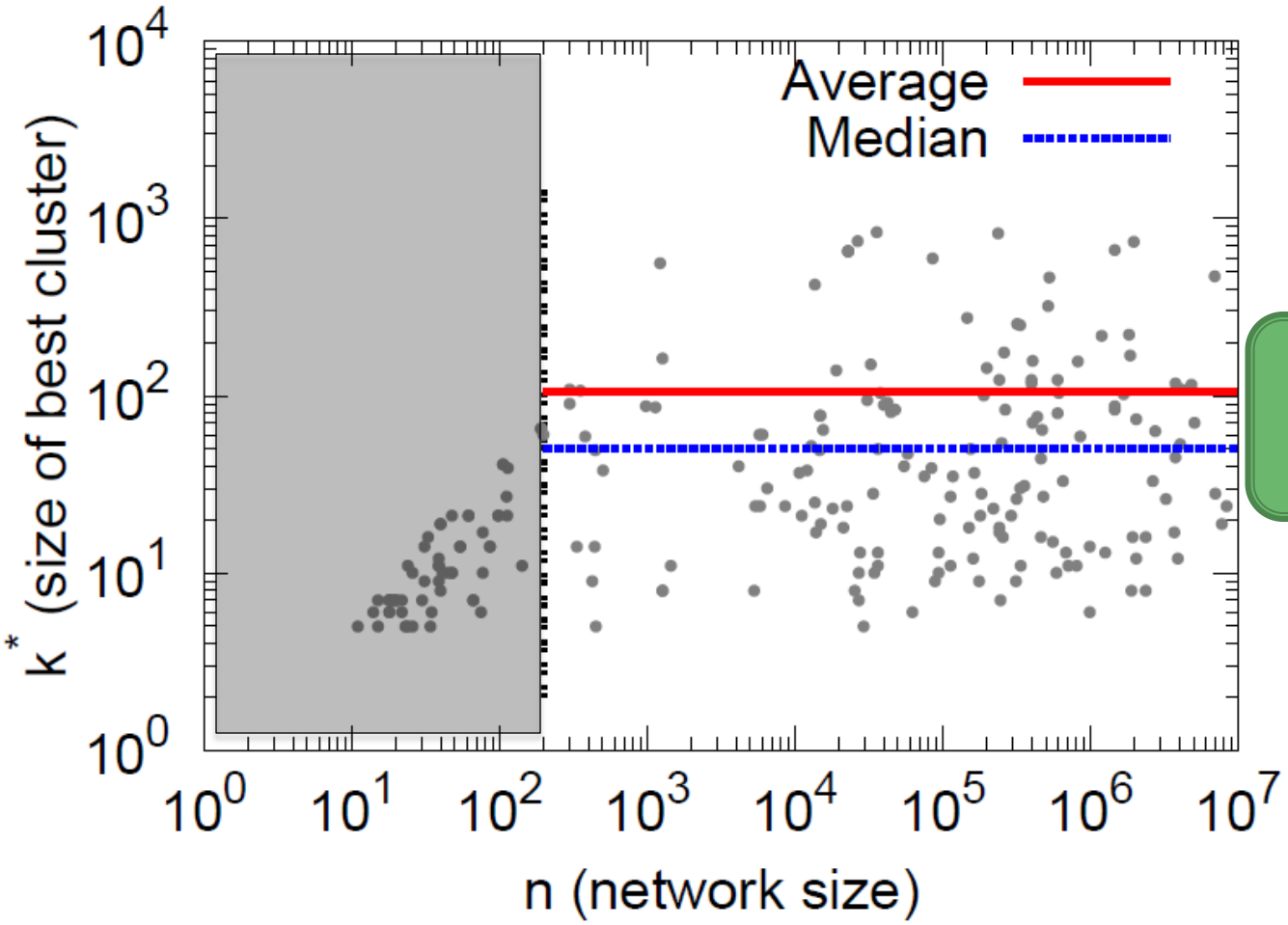


Best community. How does it scale with network size?

Whiskers are responsible for downward slope of NCP plot

[Leskovec et al. Arxiv '09]

Community size is constant!



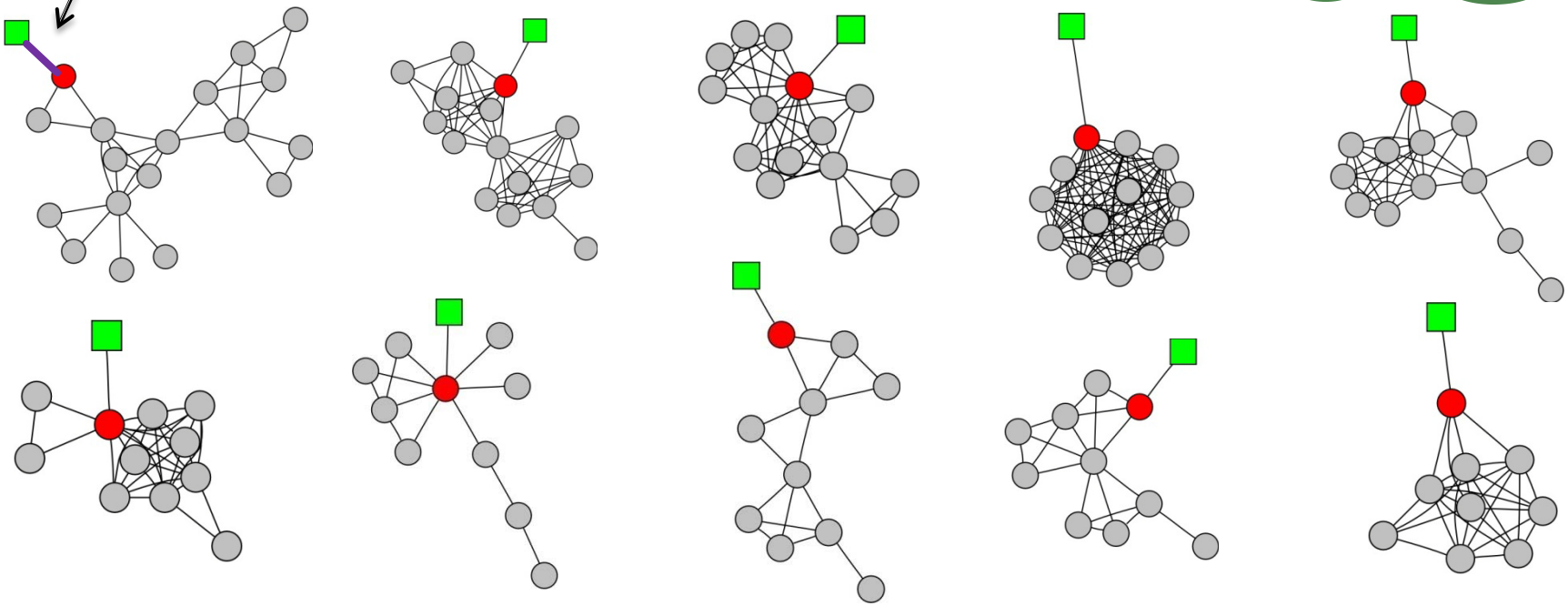
Practically **constant!**

- Each dot is a different network

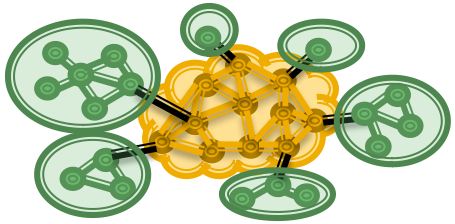
[Leskovec et al. Arxiv '09]

Whisker shapes

Edge to cut



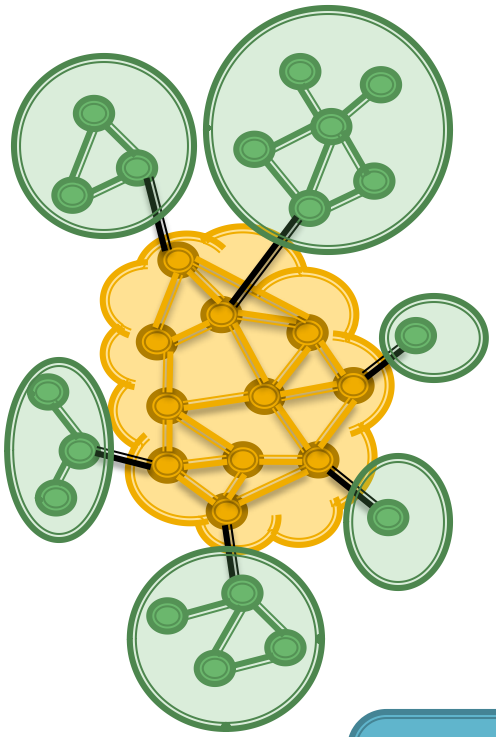
Whiskers:



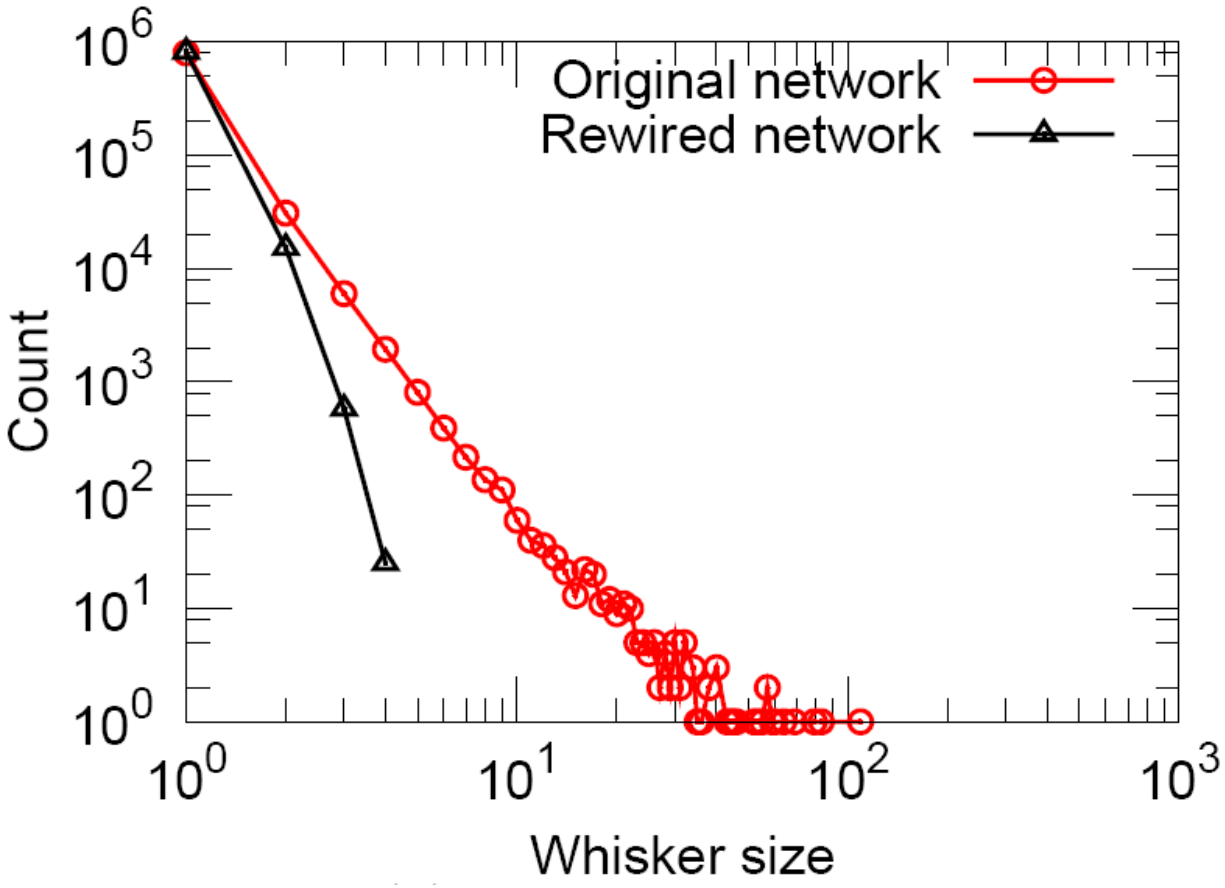
Whiskers in real networks are non-trivial (richer than trees)

[Leskovec et al. Arxiv '09]

Whiskers: Result of edge sparsity?



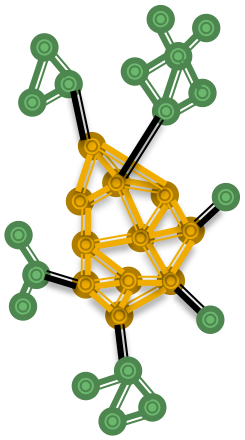
Whiskers

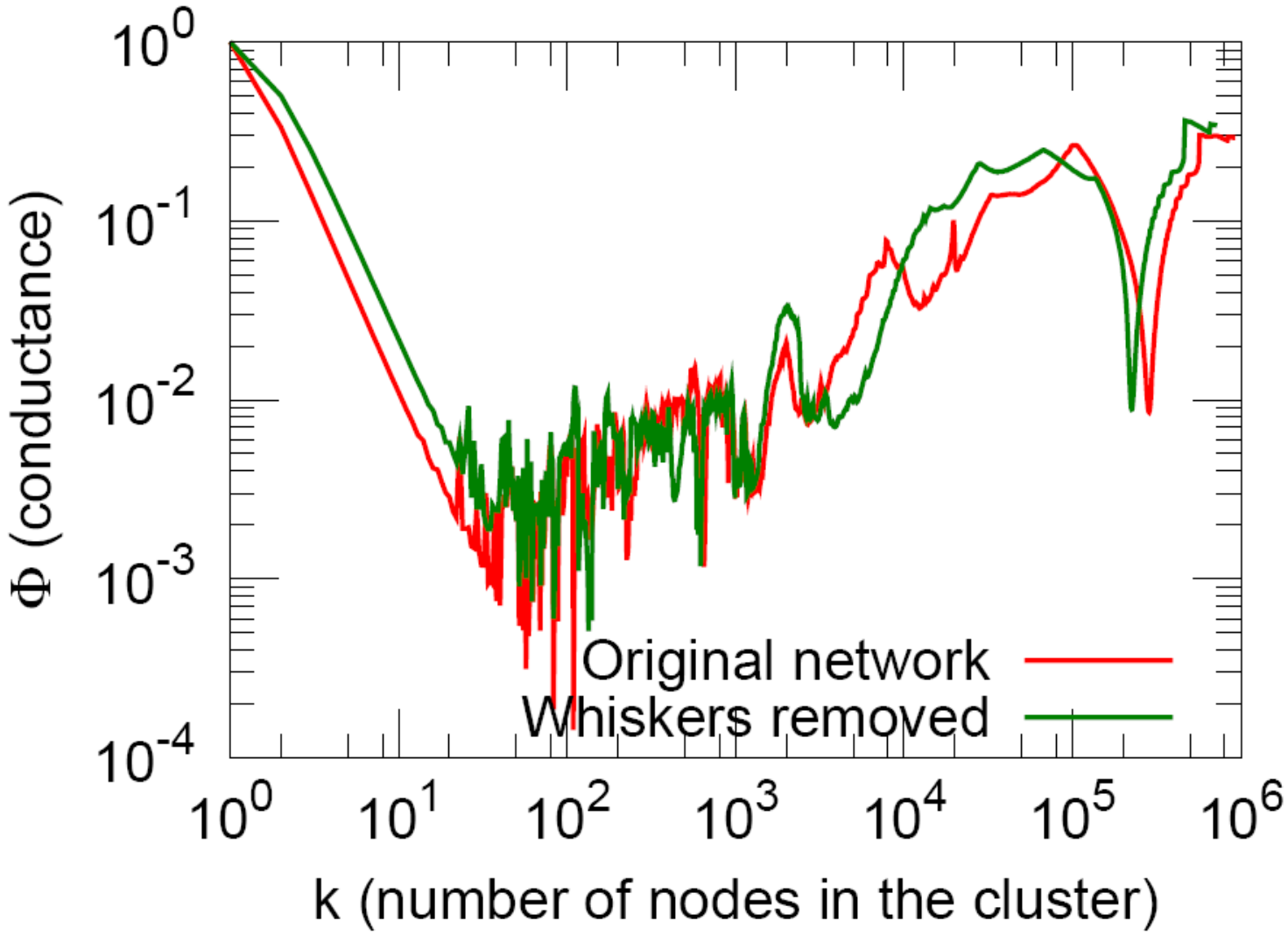


Whiskers in real networks are **larger** than expected based on density and degree sequence

[Leskovec et al. Arxiv '09]

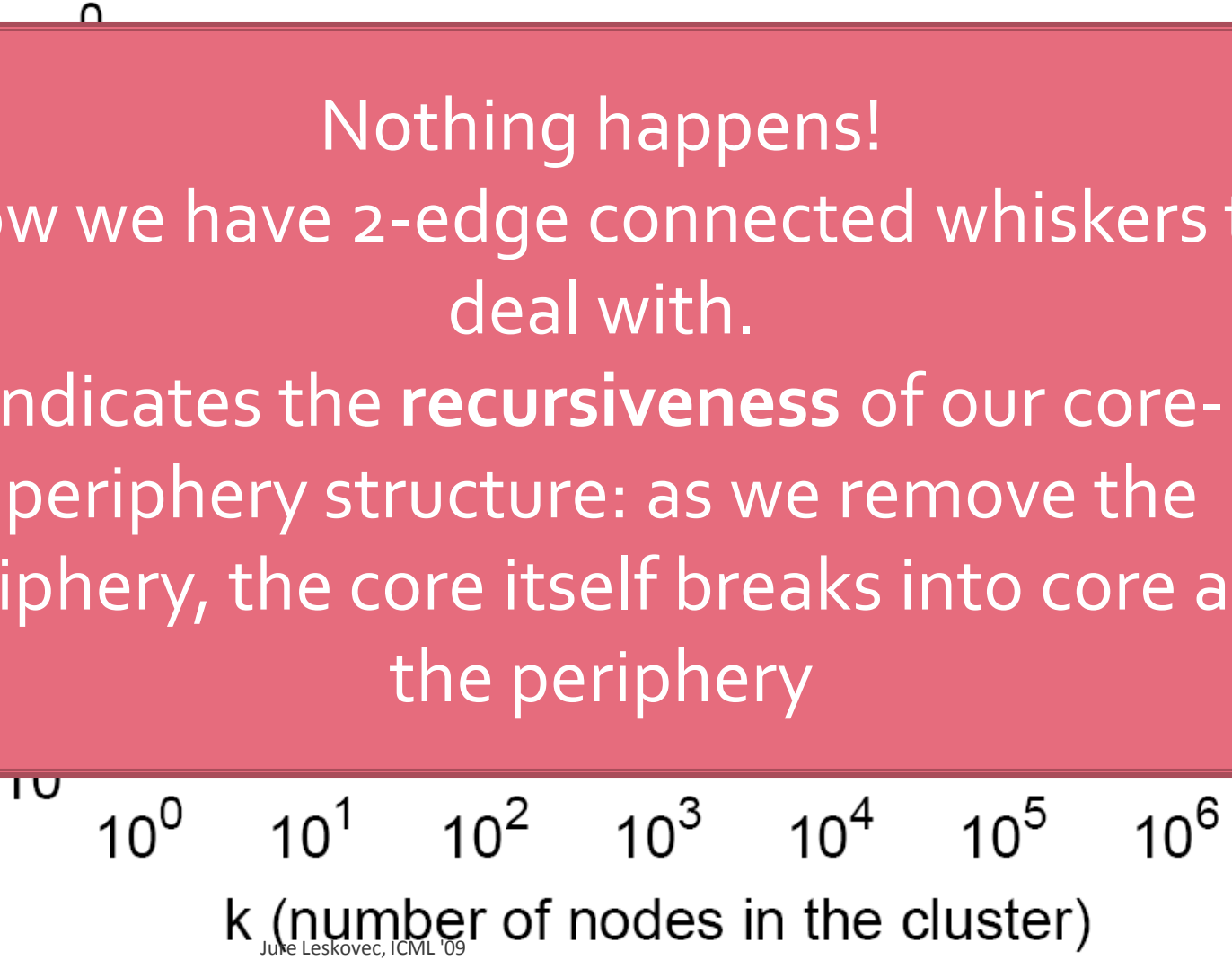
What if we remove whiskers?





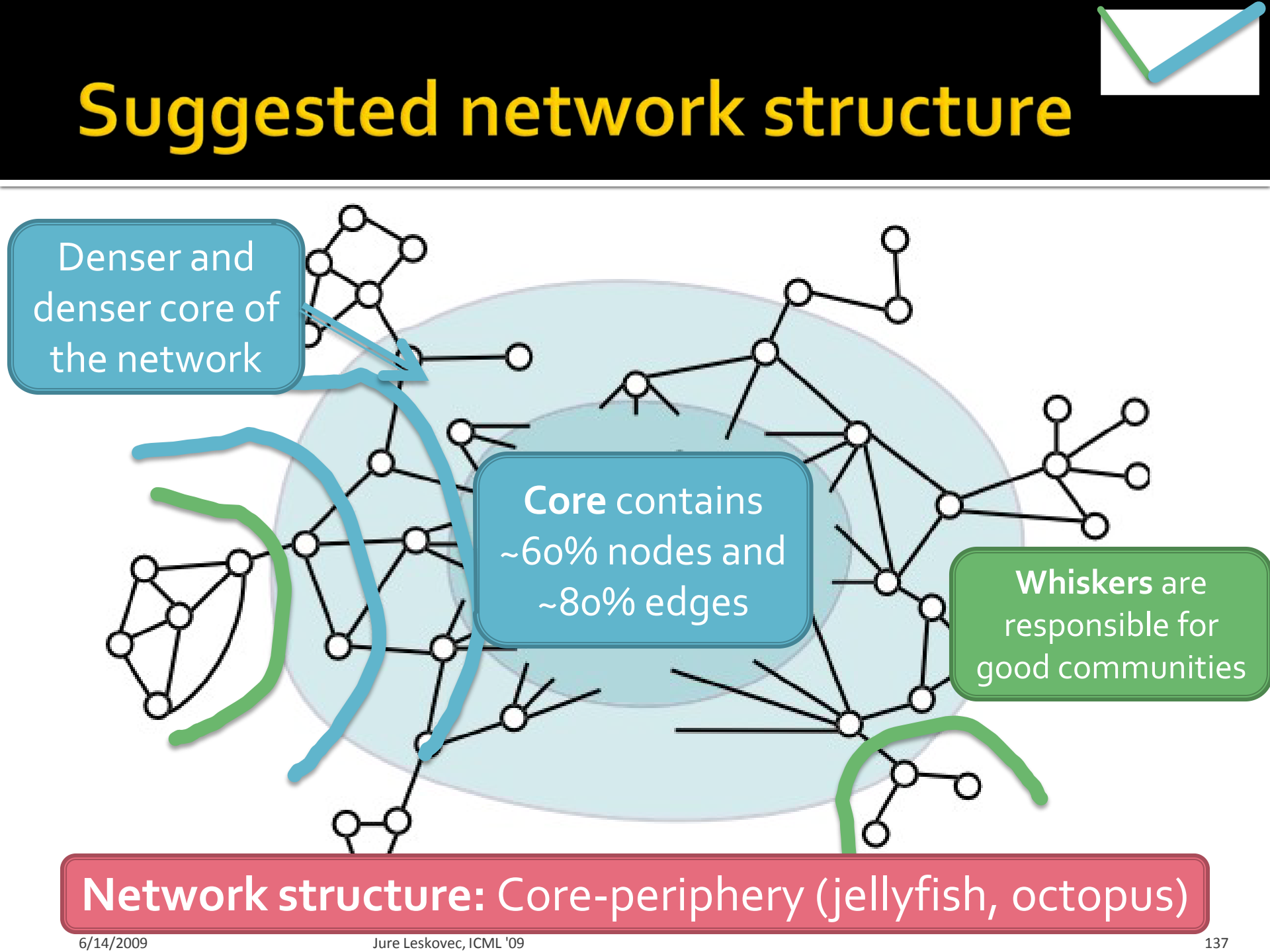
[Leskovec et al. Arxiv '09]

What if we remove whiskers?



Nothing happens!
Now we have 2-edge connected whiskers to deal with.
Indicates the **recursiveness** of our core-periphery structure: as we remove the periphery, the core itself breaks into core and the periphery

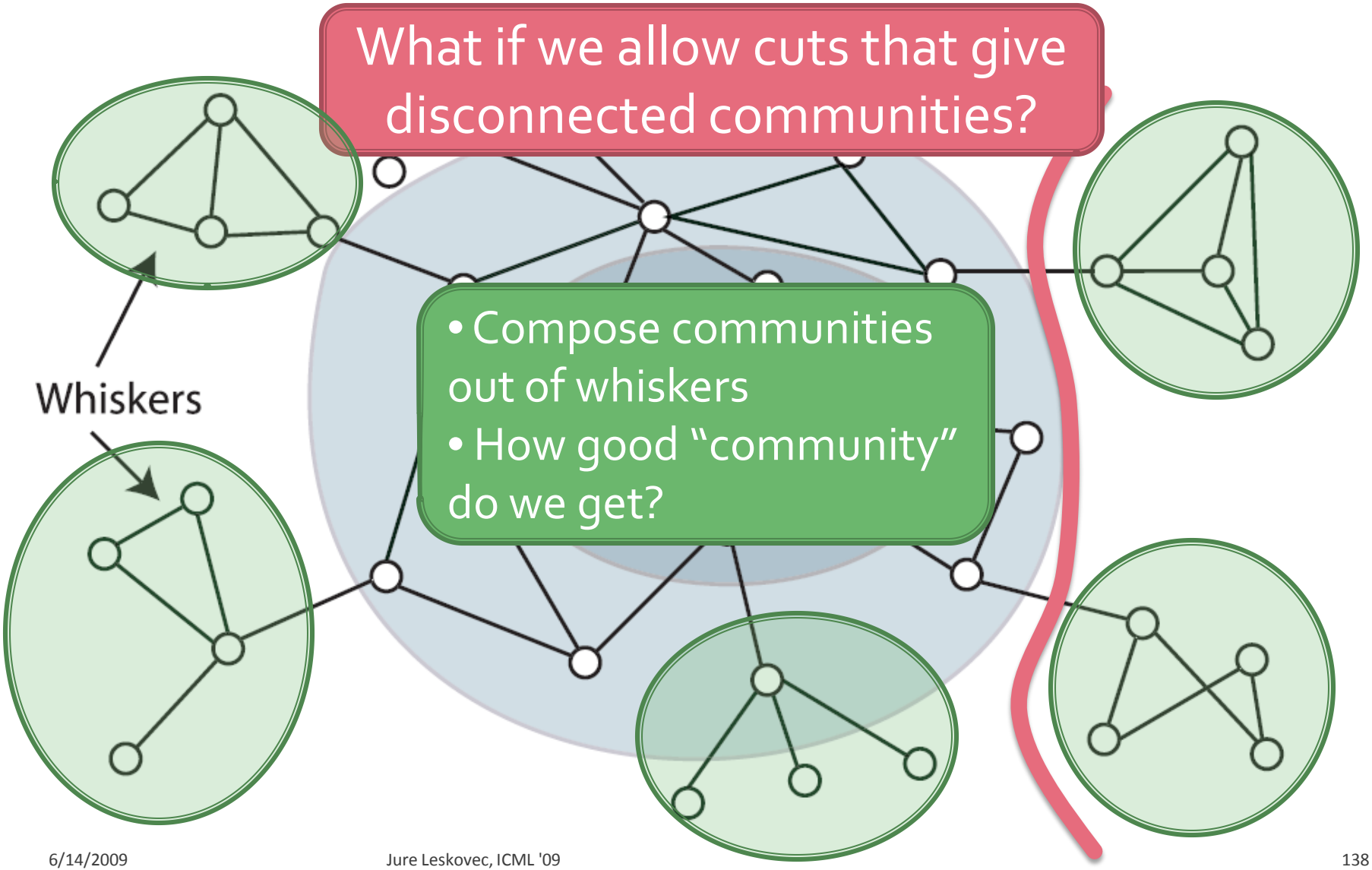
Suggested network structure



Network structure: Core-periphery (jellyfish, octopus)

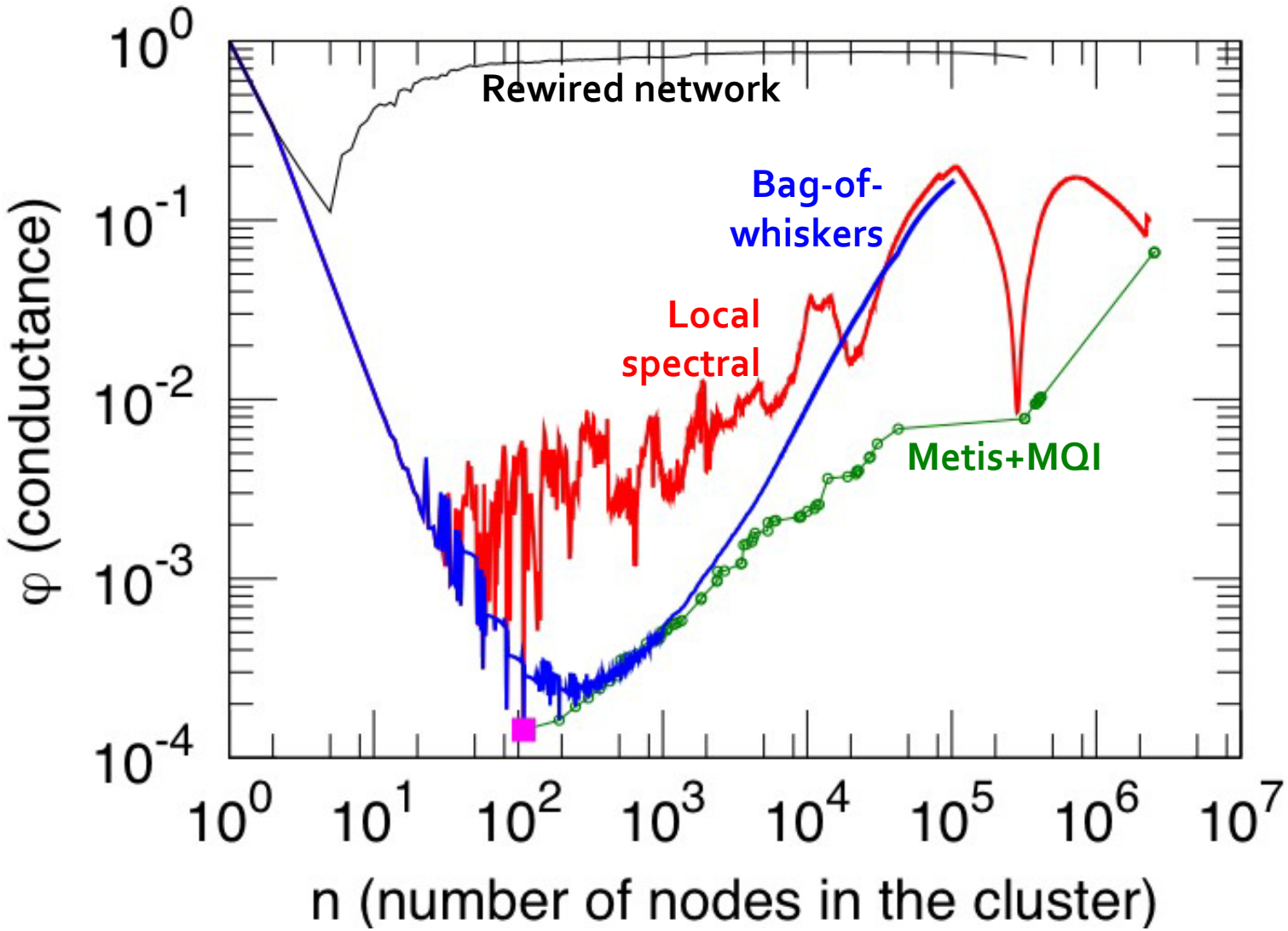
[Leskovec et al. Arxiv '09]

Caveat: How do we cut?



[Leskovec et al. Arxiv '09]

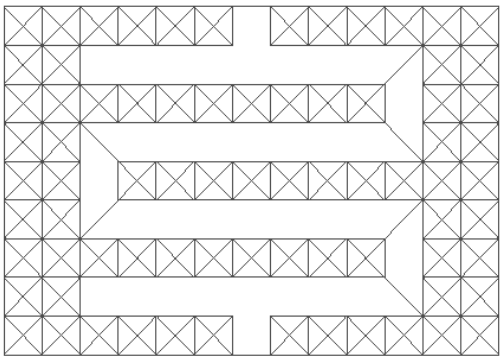
NCP: Complete picture

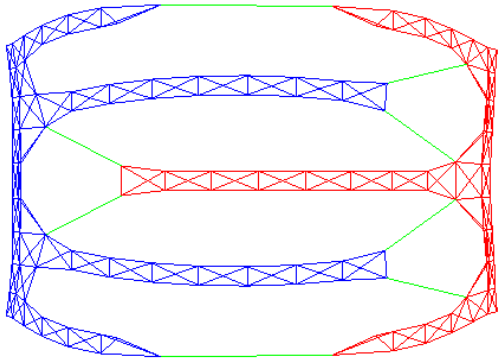


[Leskovec et al. Arxiv '09]

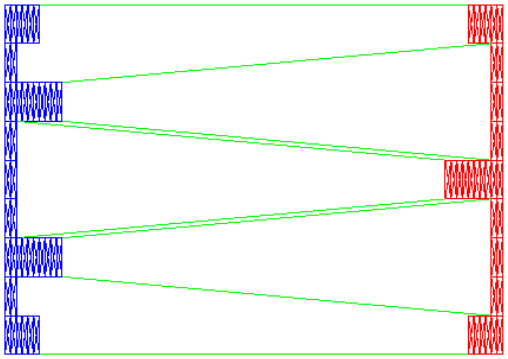
"Regularization" & spectral methods

- Regularization properties: spectral embeddings stretch along directions in which the random-walk mixes slowly
  - Resulting hyperplane cuts have "good" conductance cuts, but may not yield the optimal cuts





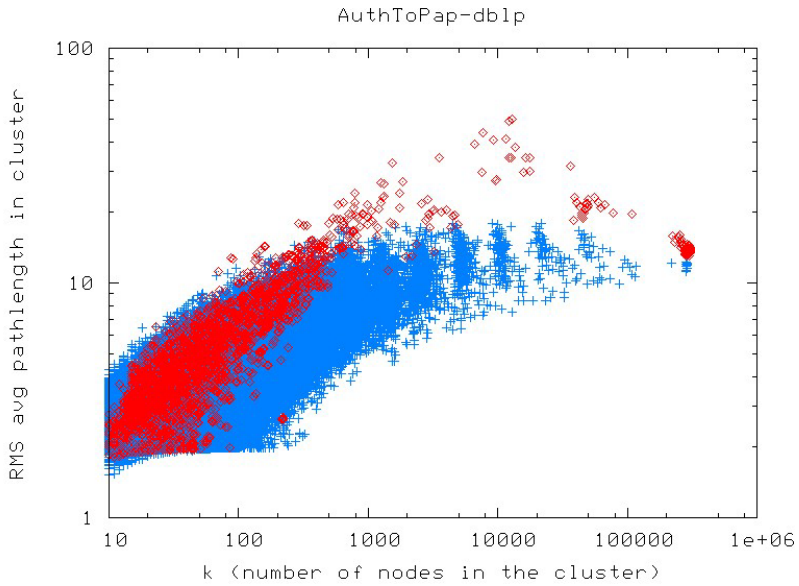
spectral embedding

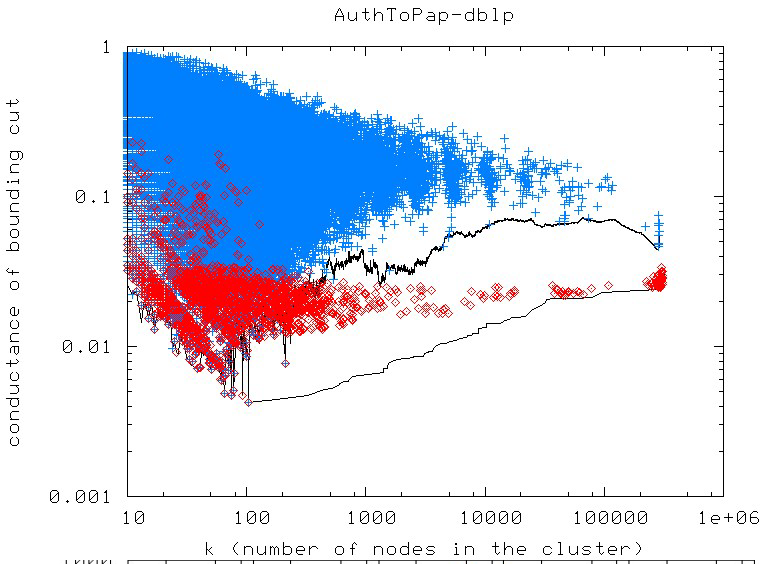


flow based embedding

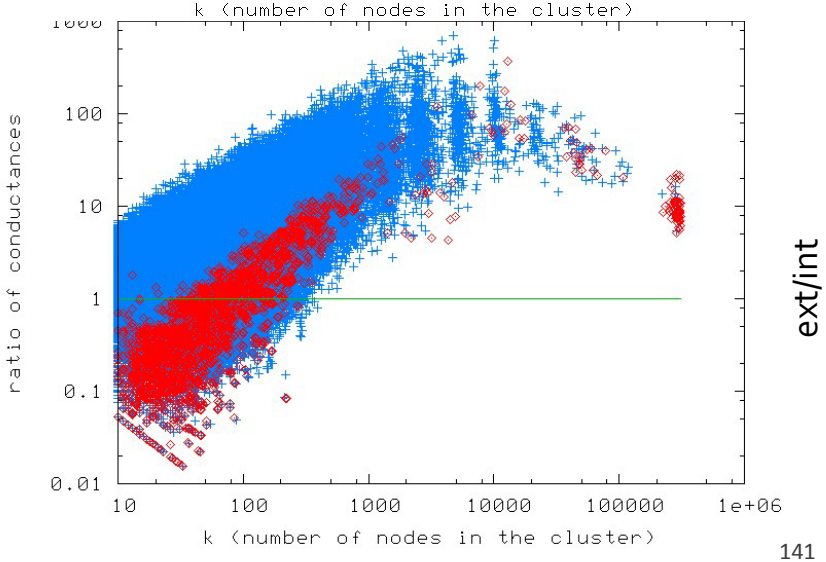
[Leskovec et al. Arxiv '09]

Regularized & non-regularized cuts





Dots are connected clusters



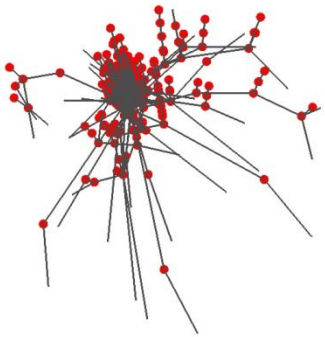
ext/int

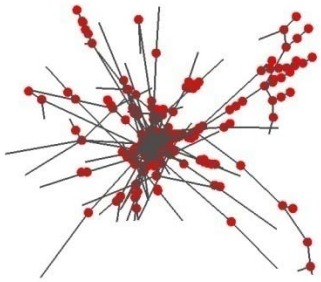
- Metis+MQI (red) gives sets with better conductance.
- Local Spectral (blue) gives tighter and more well-rounded sets.

[Leskovec et al. Arxiv '09]

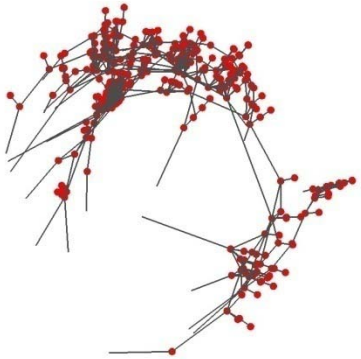
Regularized & non-regularized cuts

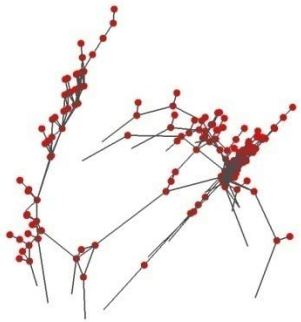
Two ca. 500 node communities from Local Spectral:





Two ca. 500 node communities from Metis+MQI:

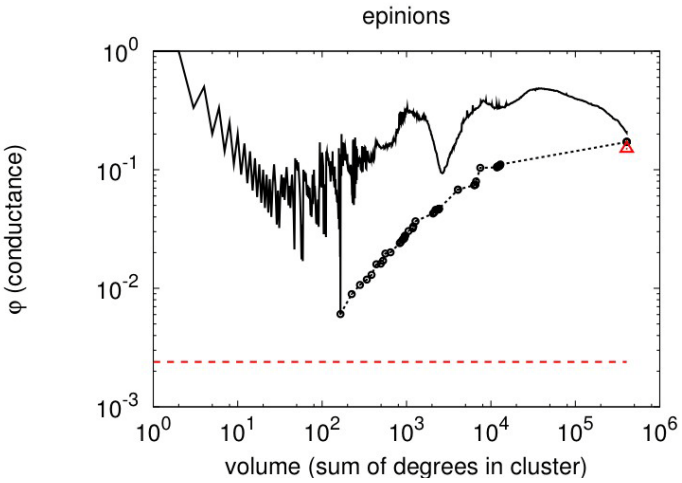


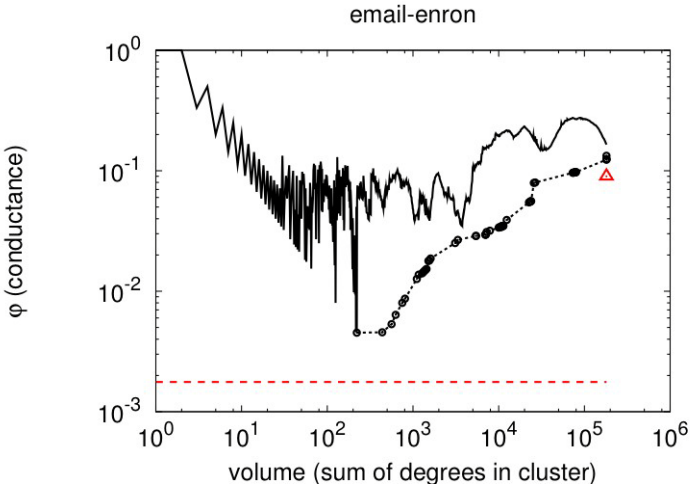


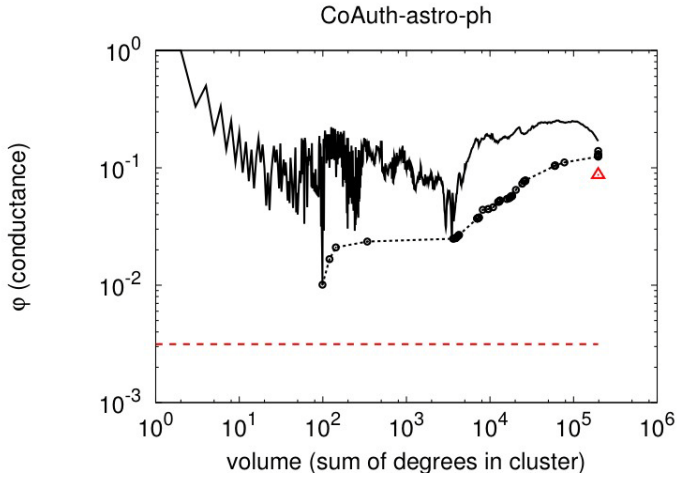
[Leskovec et al. Arxiv '09]

Lower bounds on conductance

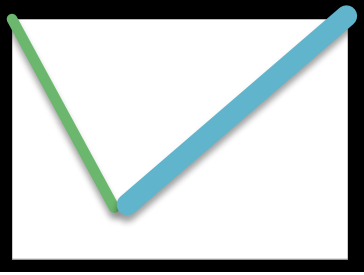
- … can be computed from:
  - Spectral embedding (independent of balance)
  - SDP-based methods (for volume-balanced partitions)

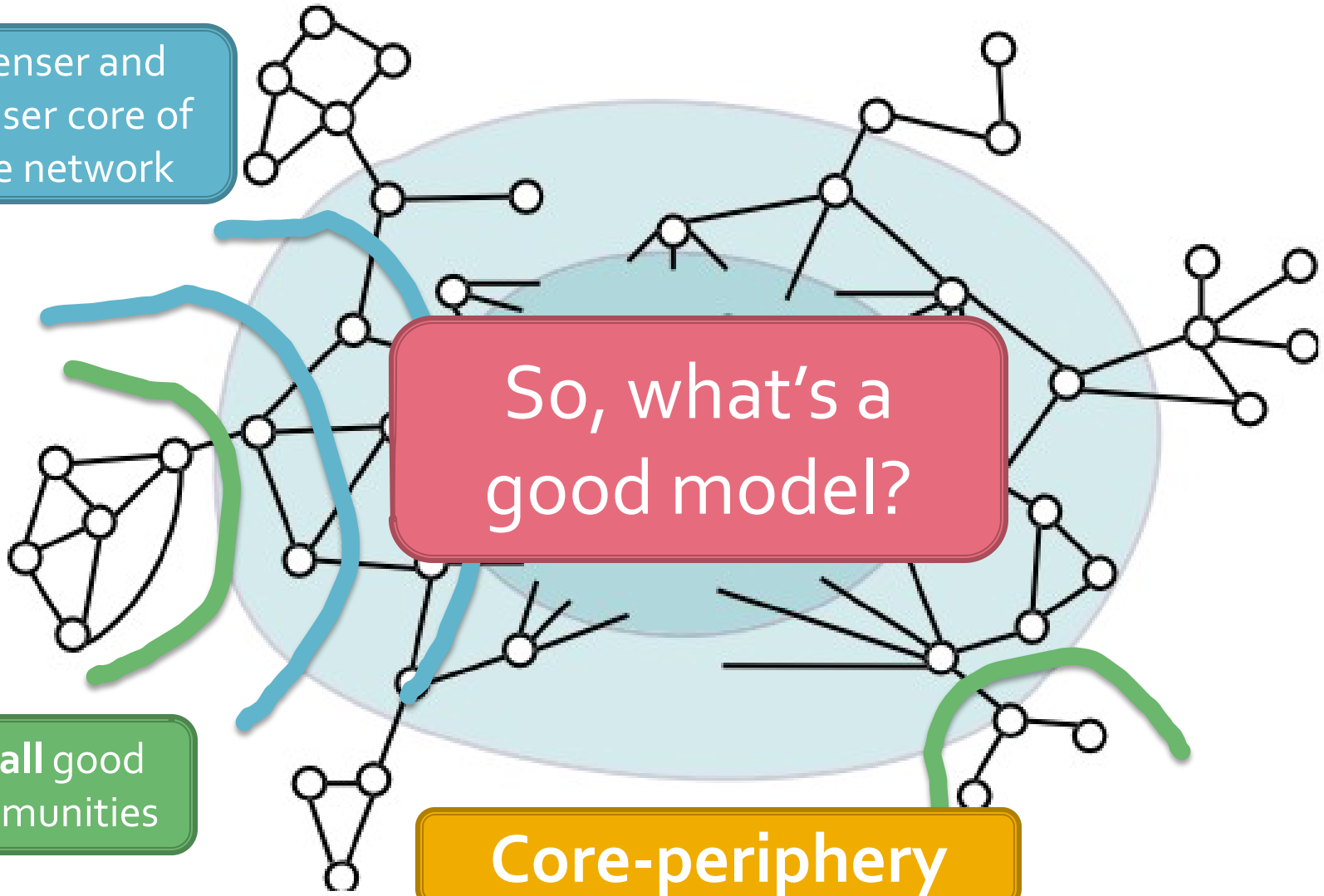






Structure of large networks

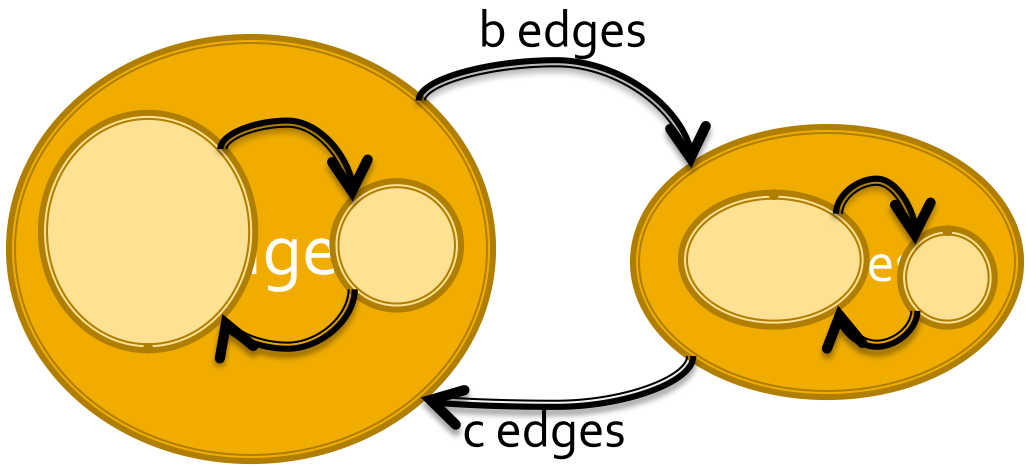




Kronecker & Network structure

- What do estimated parameters tell us about the network structure?

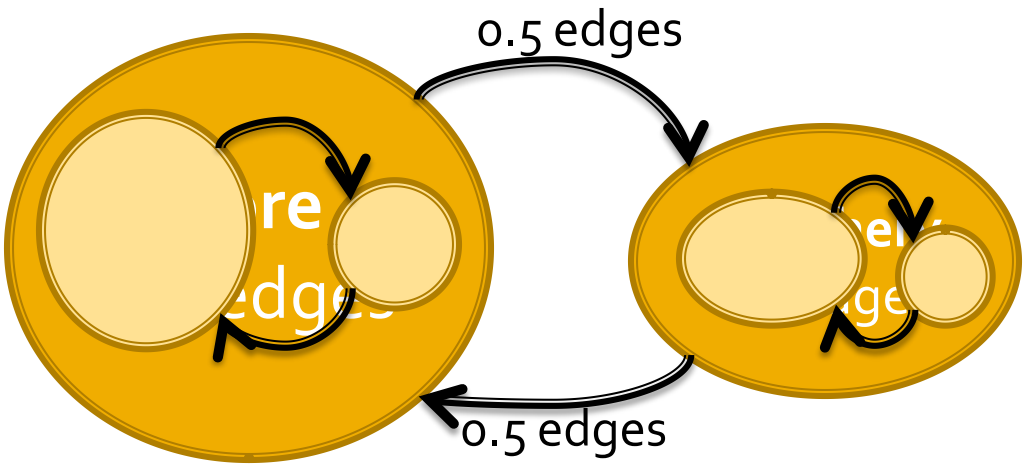
$$K_1 = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

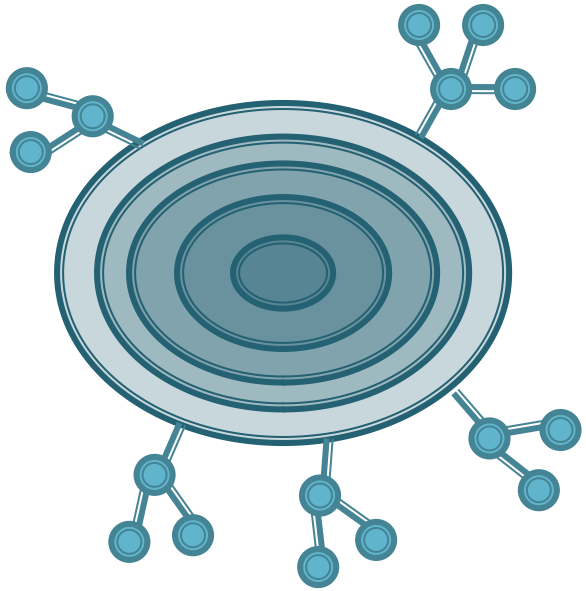


Kronecker & Network structure

- What do estimated parameters tell us about the network structure?

$$K_1 = \begin{bmatrix} 0.9 & 0.5 \\ 0.5 & 0.1 \end{bmatrix}$$

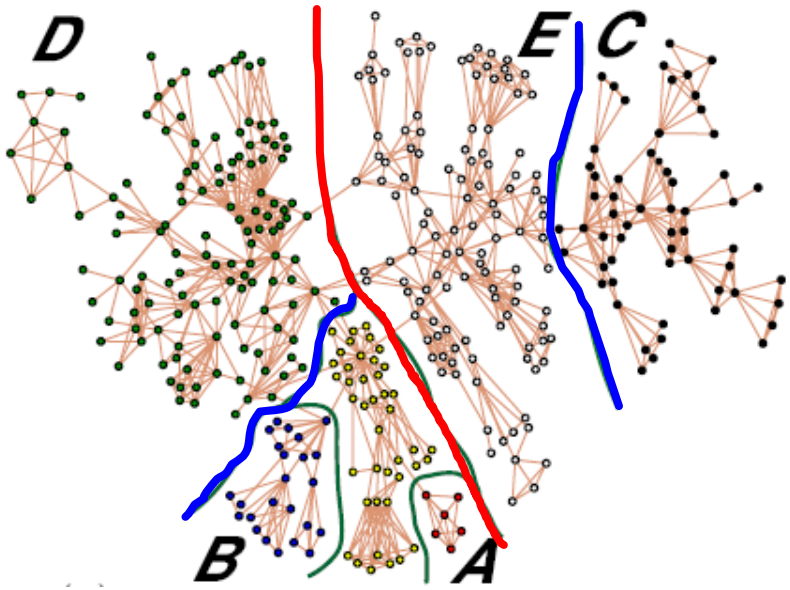




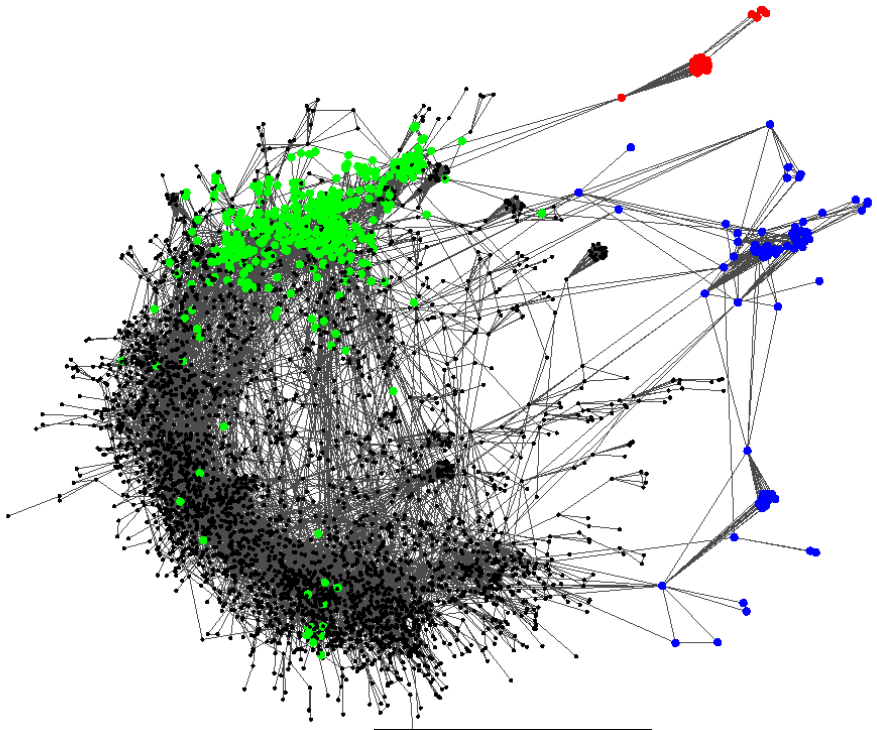
Core-periphery

Small vs. Large networks

- Small and large networks are very different:



$$K_1 = \begin{bmatrix} 0.99 & 0.17 \\ 0.17 & 0.82 \end{bmatrix}$$



$$K_1 = \begin{bmatrix} 0.99 & 0.54 \\ 0.49 & 0.13 \end{bmatrix}$$

Cluster structure of large networks

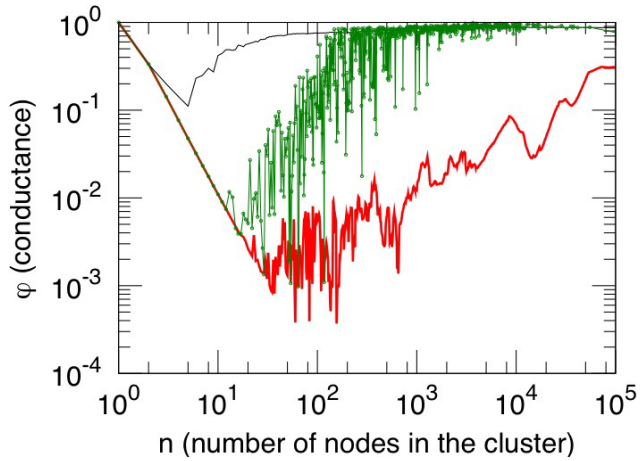
- **Small communities:**
  - Largest have ≈100 nodes
  - Community size is independent of network size
- **Core:**
  - 60% of the nodes, 80% edges
  - Core has little structure (hard to cut)
  - Still more structure than the random network

Comparison with "Ground truth"

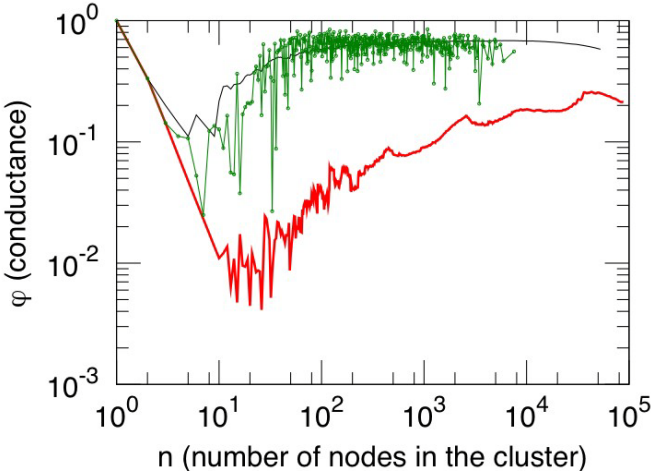
- Compare to networks where nodes explicitly declare group membership:
  - LiveJournal12:
    - users create and explicitly join on-line groups
  - DBLP co-authorships:
    - publication venues can be viewed as communities
  - Amazon product co-purchasing:
    - each item belongs to one or more hierarchically organized categories, as defined by Amazon
  - IMDB collaboration:
    - countries of production and languages may be viewed as communities

[Leskovec et al. Arxiv '09]

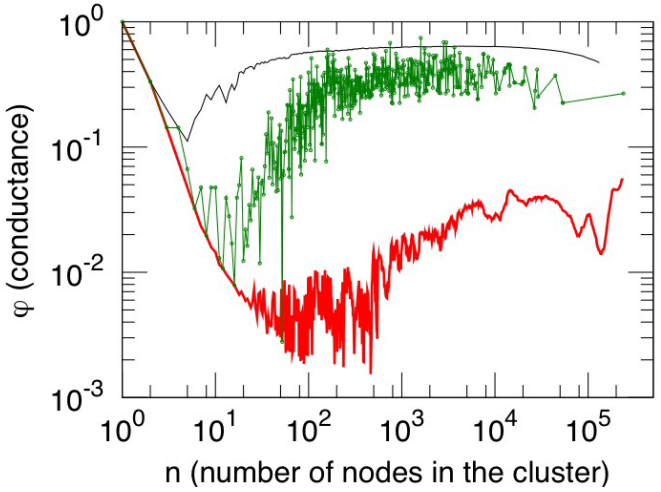
Comparison with "Ground truth"



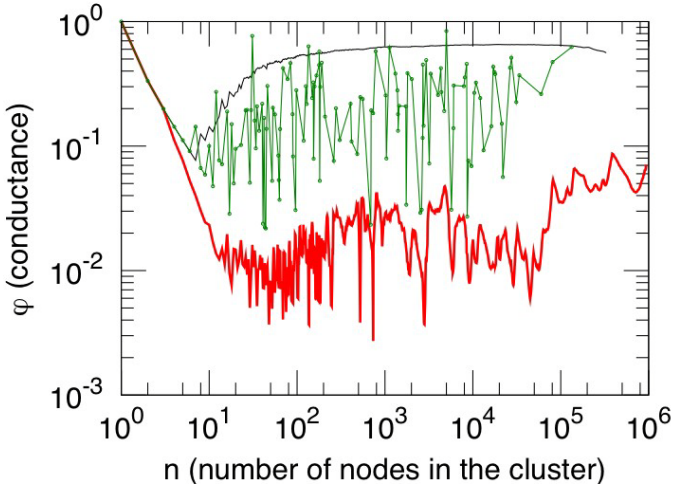
LiveJournal



DBLP



Amazon



IMDB

Rewired

Network

Ground truth

Community structure: Conclusion

- Community structure of large networks:
  - Recursive Core-periphery structure
  - Scale to natural community size: Dunbar number
    - 150 individuals is maximum community size
- Model: Kronecker graphs
  - Analytically tractable: provable properties
  - Can efficiently estimate parameters from data

Community structure: Implications

- **Large social & information networks**
  - No large clusters: no/little hierarchical structure
  - Can't be well embedded – no underlying geometry
- Are fundamentally different from small networks and manifolds
- **So…** in large networks…
  - Manifold learning won't really work
  - Semi-supervised learning ideas won't really work (in the core)

The end: Conclusion

- Statistical properties of networks across various domains
  - Key to understanding the behavior of many "independent" nodes
- Models of network structure and growth
  - Help explain, think and reason about properties
- Prediction, understanding of the structure
  - Fitting the models

The end: Reflections

- How to systematically characterize the network structure?
- How do properties relate to one another?
- Is there something else we should measure?

The end: Reflections

- Why are networks the way they are?
- Steer the network evolution
- Predictive modeling of large communities
  - Online massively multi-player games are closed worlds with detailed traces of activity
- Design systems (networks) that will
  - Be robust to node failures
  - Support local search (navigation): P2P networks

The end: Reflections

- Why are networks the way they are?
- Only recently have basic properties been observed on a large scale
  - Confirms social science intuitions; calls others into question
- What are good tractable network models?
  - Builds intuition and understanding
- Benefits of working with large data
  - Observe structures not visible at smaller scales