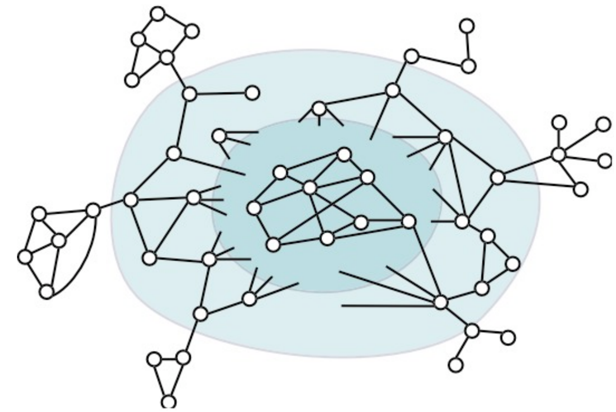


Geometric Network Analysis Tools

Michael W. Mahoney

Stanford University
MMDS, June 2010

(For more info, see:
http:// cs.stanford.edu/people/mmahoney/
or Google on "Michael Mahoney")

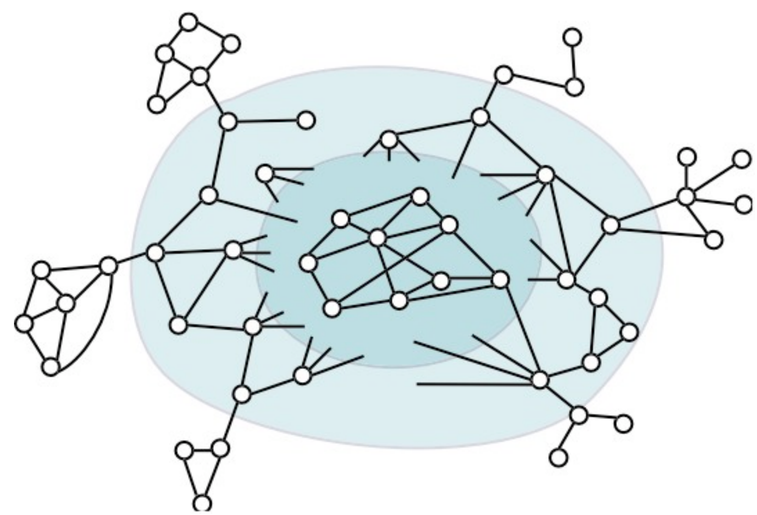
Networks and networked data

Lots of "networked" data!!

- technological networks
  - AS, power-grid, road networks
- biological networks
  - food-web, protein networks
- social networks
  - collaboration networks, friendships
- information networks
  - co-citation, blog cross-postings, advertiser-bidded phrase graphs...
- language networks
  - semantic networks...
- ...

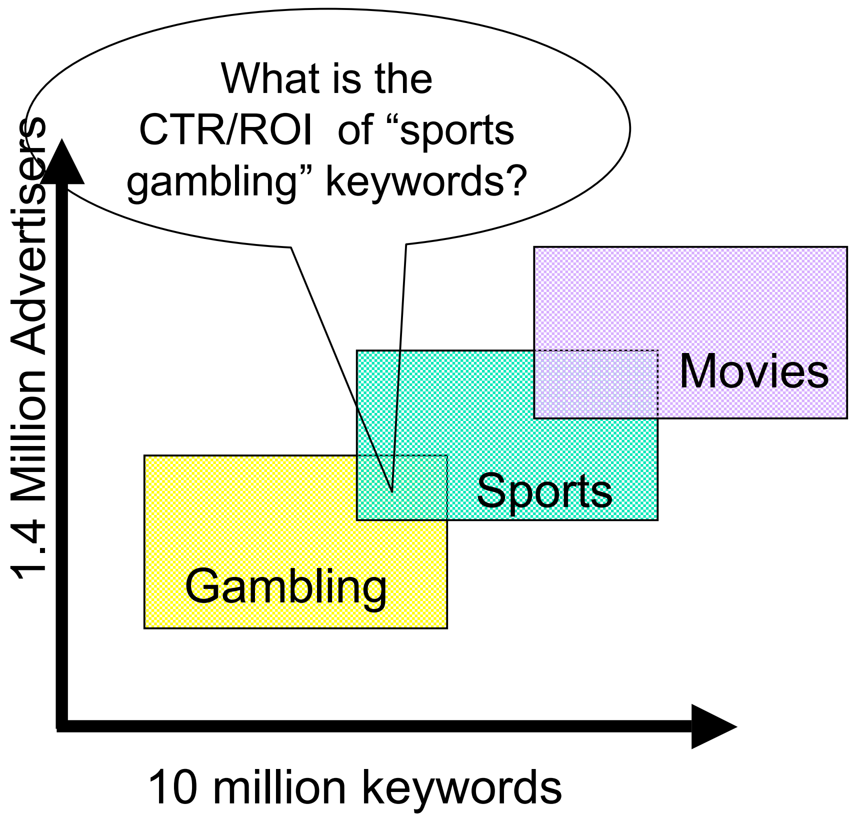
Interaction graph model of networks:

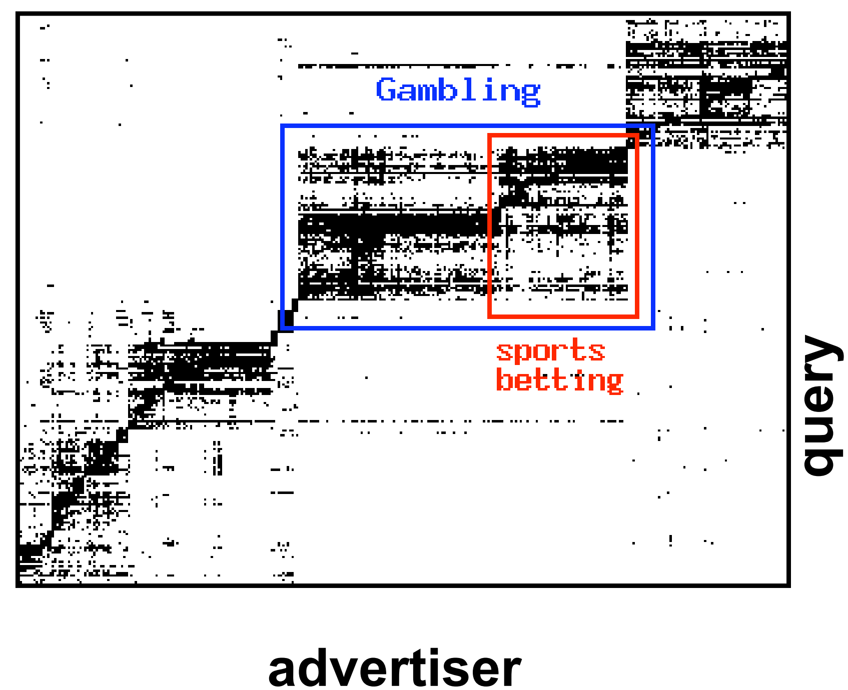
- Nodes represent "entities"
- Edges represent "interaction" between pairs of entities



Micro-markets in sponsored search "keyword-advertiser graph"

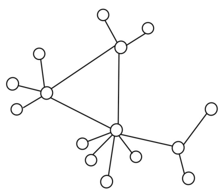
Goal: Find *isolated* markets/clusters with *sufficient money/clicks* with *sufficient coherence.*
Ques: Is this even possible?

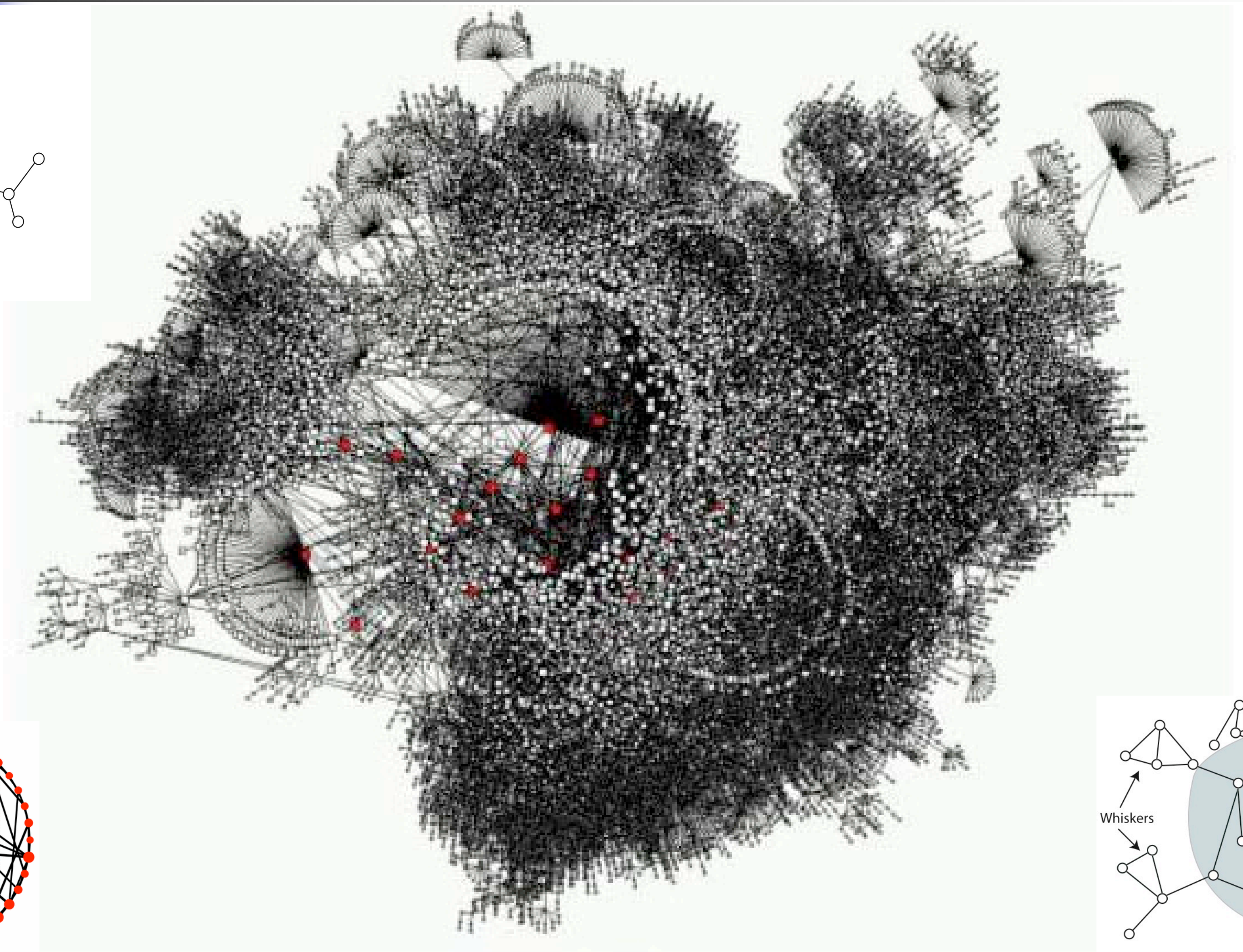


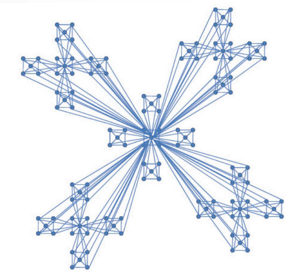


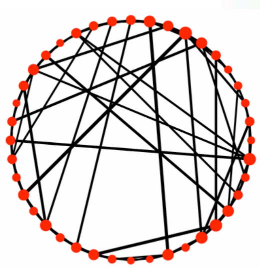
Question: Is this visualization evidence for the schematic on the left?

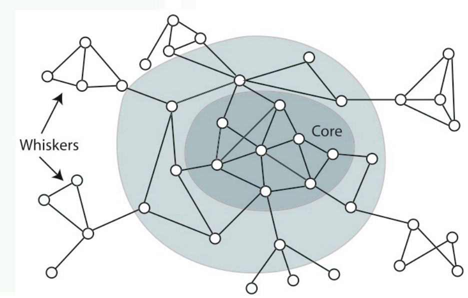
What do these networks "look" like?



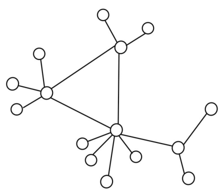








Popular approaches to large network data

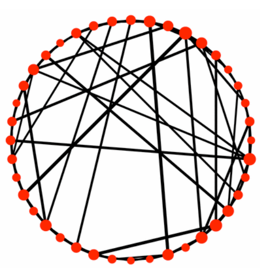


Heavy-tails and power laws (at *large size-scales*):

• extreme heterogeneity in local environments, e.g., as captured by degree distribution, and relatively unstructured otherwise

• basis for preferential attachment models, optimization-based models, power-law random graphs, etc.

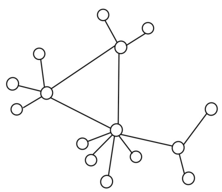
Local clustering/structure (at *small size-scales*):



• local environments of nodes have structure, e.g., captures with clustering coefficient, that is meaningfully "geometric"

• basis for small world models that start with global "geometry" and add random edges to get small diameter and preserve local "geometry"

Popular approaches to data more generally

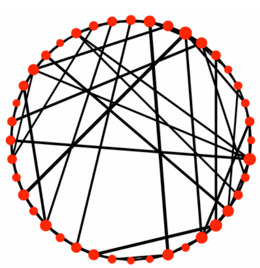


Use **geometric data analysis tools**:

- Low-rank methods - very popular and flexible
- "Kernel" and "manifold" methods - use other distances, e.g., diffusions or nearest neighbors, to find "curved" low-dimensional spaces

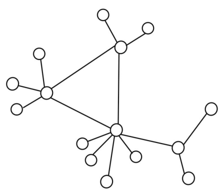
These **geometric data analysis tools**:

- View data as a point cloud in R^n, i.e., *each of the m data points is a vector in R^n*
- Based on SVD*, a basic vector space *structural* result
- Geometry gives a lot -- scalability, robustness, capacity control, basis for inference, etc.



*perhaps in an implicitly-defined infinite-dimensional non-linearly transformed feature space

Can these approaches be combined?

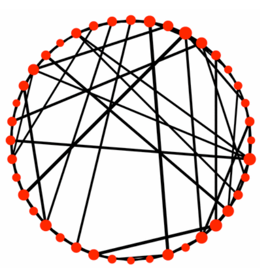


These approaches are *very* different:

- network is a *single data point*---*not a collection of feature vectors* drawn from a distribution, and not really a matrix
- can't easily let m or n (number of data points or features) go to infinity---so nearly every such theorem fails to apply

Can associate matrix with a graph, vice versa, *but*:

- often do more damage than good
- questions asked tend to be *very* different
- graphs are really *combinatorial* things*



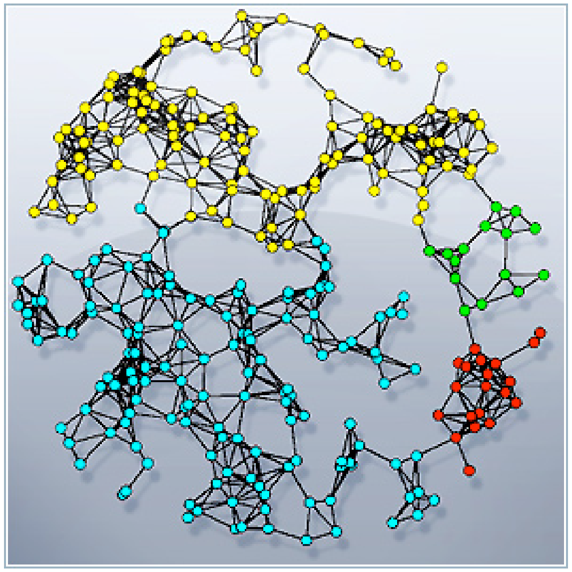
*But, graph geodesic distance is a metric, and metric embeddings give fast approximation algorithms in worst-case CS analysis!

Overview

- Large networks and different perspectives on data
- Approximation algorithms as "experimental probes"
  - Graph partitioning: good test case for different approaches to data
  - Geometric/statistical properties *implicit* in worst-case algorithms
- An example of the theory
  - Local spectral graph partitioning as an optimization problem
  - Exploring data graphs locally: practice follows theory closely
- An example of the practice
  - Local and global clustering structure in very large networks
  - Strong theory allows us to make *very* strong applied claims

Graph partitioning

A family of combinatorial optimization problems - want to partition a graph's nodes into two sets s.t.:



- Not much edge weight across the cut (cut quality)
- Both sides contain a lot of nodes

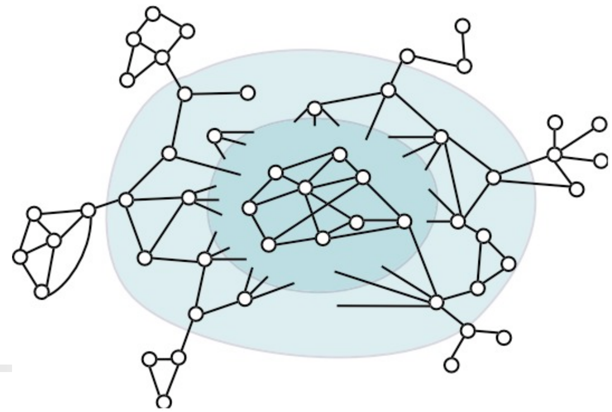
Several standard formulations:

- Graph bisection (minimum cut with 50-50 balance)
- β-balanced bisection (minimum cut with 70-30 balance)
- cutsize/min{|A|,|B|}, or cutsize/(|A||B|) (expansion)
- cutsize/min{Vol(A),Vol(B)}, or cutsize/(Vol(A)Vol(B)) (conductance or N-Cuts)

All of these formalizations are NP-hard!

Later: size-resolved conductance: *algs can have non-obvious size-dependent behavior!*

Why graph partitioning?



Graph partitioning algorithms:

• capture a qualitative notion of connectedness

• well-studied problem, both in theory and practice

• many machine learning and data analysis applications

• good "hydrogen atom" to work through the method (*since spectral and max flow methods embed in very different places*)

We *really* don't care about exact solution to intractable problem:

• output of approximation algs is *not* something we "settle for"

• randomized/approximation algorithms give "better" answers than exact solution

Exptl Tools: Probing Large Networks with Approximation Algorithms

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 for mesh-like graphs) - common in practice

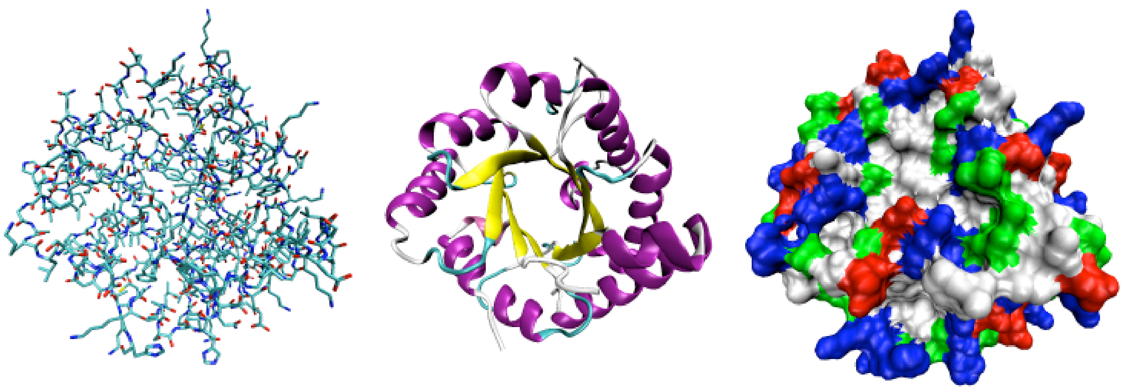
X+MQI - post-processing step on, e.g., Spectral of Metis

Metis+MQI - best conductance (empirically)

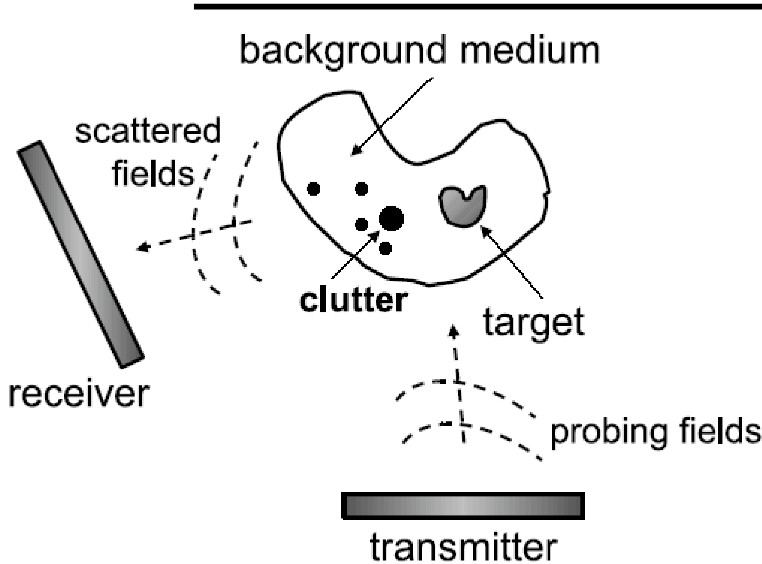
Local Spectral - connected and tighter sets (empirically, regularized communities!)

We are not interested in partitions per se, but in probing network structure.

Analogy: What does a protein look like?



Three possible representations (all-atom; backbone; and solvent-accessible surface) of the three-dimensional structure of the protein triose phosphate isomerase.



Experimental Procedure:

- Generate a bunch of output data by using the unseen object to filter a known input signal.
- Reconstruct the unseen object given the output signal and what we know about the artifactual properties of the input signal.

Overview

- Large networks and different perspectives on data
- Approximation algorithms as "experimental probes"
  - Graph partitioning: good test case for different approaches to data
  - Geometric/statistical properties *implicit* in worst-case algorithms
- An example of the theory
  - Local spectral graph partitioning as an optimization problem
  - Exploring data graphs locally: practice follows theory closely
- An example of the practice
  - Local and global clustering structure in very large networks
  - Strong theory allows us to make *very* strong applied claims

Recall spectral graph partitioning

The basic optimization problem:

$$\begin{aligned} \text{minimize} \quad & x^T L_G x \\ \text{s.t.} \quad & \langle x, x \rangle_D = 1 \\ & \langle x, 1 \rangle_D = 0 \end{aligned}$$

- Relaxation of:

$$\phi(G) = \min_{S \subset V} \frac{E(S, \bar{S})}{Vol(S) Vol(\bar{S})}$$

- Solvable via the eigenvalue problem: $$\mathcal{L}_G y = \lambda_2(G) y$$

- Sweep cut of second eigenvector yields:

$$\lambda_2(G)/2 \leq \phi(G) \leq \sqrt{8 \lambda_2(G)}$$

Local spectral partitioning ansatz

Mahoney, Orecchia, and Vishnoi (2010)

Primal program:

$$\begin{aligned} \text{minimize} \quad & x^T L_G x \\ \text{s.t.} \quad & < x, x >_D = 1 \\ & < x, s >_D^2 \geq \kappa \end{aligned}$$

Interpretation:

- Find a cut well-correlated with the seed vector s - *geometric notion of correlation between cuts*!
- If s is a single node, this relaxes:

$$\min_{S \subset V, s \in S, |S| \leq 1/k} \frac{E(S, \bar{S})}{Vol(S) Vol(\bar{S})}$$

Dual program:

$$\begin{aligned} \max \quad & \alpha - \beta(1-\kappa) \\ \text{s.t.} \quad & L_G \succeq \alpha L_{K_n} - \beta \left(\frac{L_{K_T}}{\mathrm{vol}(\bar{T})} + \frac{L_{K_{\bar{T}}}}{\mathrm{vol}(T)} \right) \\ & \beta \geq 0 \end{aligned}$$

Interpretation:

- Embedding a combination of scaled complete graph K_n and complete graphs T and $\underline{T}$ (K_T and $K_{\underline{T}}$) - where the latter encourage cuts near (T,$\underline{T}$).

Main results (1 of 2)

Mahoney, Orecchia, and Vishnoi (2010)

Theorem: If x^* is an optimal solution to LocalSpectral, it is a GPPR* vector for parameter α, and it can be computed as the solution to a set of linear equations.

Proof:

(1) Relax non-convex problem to convex SDP

(2) Strong duality holds for this SDP

(3) Solution to SDP is rank one (from comp. slack.)

(4) Rank one solution is GPPR vector.

**GPPR vectors generalize Personalized PageRank, e.g., with negative teleportation - think of it as a *more flexible regularization tool* to use to "probe" networks.

Main results (2 of 2)

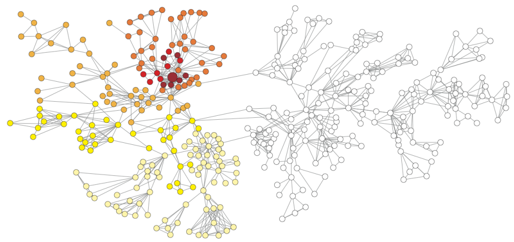
Mahoney, Orecchia, and Vishnoi (2010)

Theorem: If x^* is optimal solution to LocalSpect(G,s,κ), one can find a cut of conductance $\leq 8\lambda(G,s,\kappa)$ in time $O(n \lg n)$ with sweep cut of x^*.

Upper bound, as usual from sweep cut & Cheeger.

Theorem: Let s be seed vector and κ correlation parameter. For all sets of nodes T s.t. $\kappa' := \langle s, s_T \rangle_D^2$, we have: $\phi(T) \geq \lambda(G,s,\kappa)$ if $\kappa \leq \kappa'$, and $\phi(T) \geq (\kappa'/\kappa)\lambda(G,s,\kappa)$ if $\kappa' \leq \kappa$.

Lower bound: Spectral version of flow-improvement algs.



Other “Local” Spectral and Flow and “Improvement” Methods

Local spectral methods - provably-good local version of global spectral

ST04: truncated“local” random walks to compute locally-biased cut

ACL06/Chung08 : locally-biased PageRank vector/heat-kernel vector

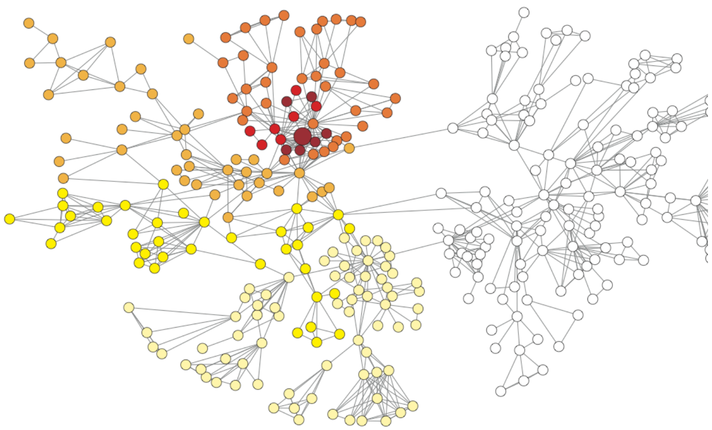
Flow improvement methods - *Given a graph G* and a partition, find a “nearby” cut that is of similar quality:

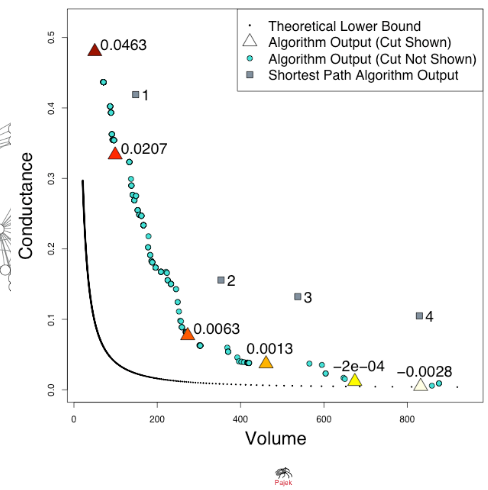
GGT89: find min conductance subset of a “small” partition

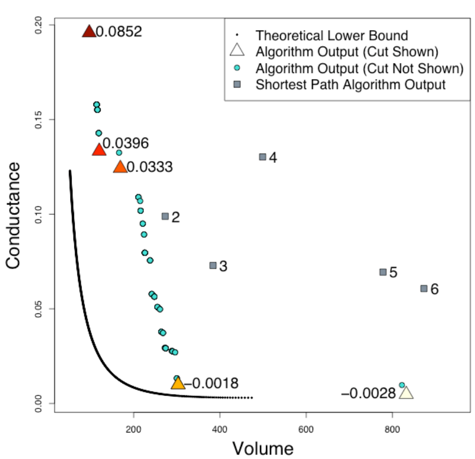
LR04,AL08: find “good” “nearby” cuts using *flow-based methods*

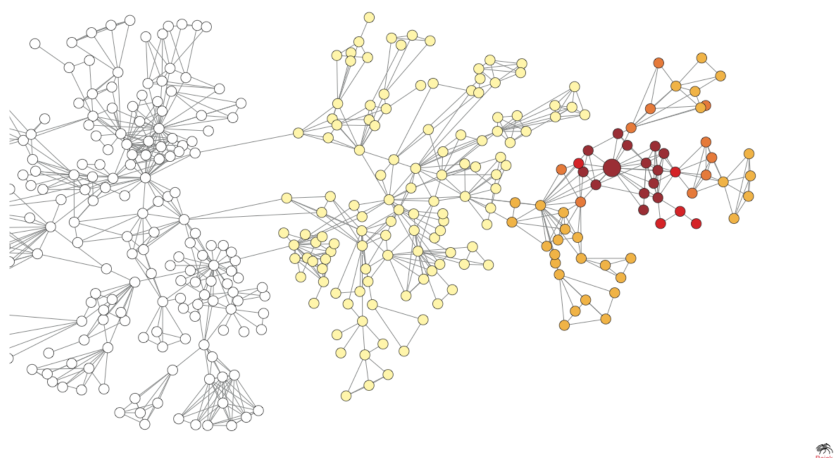
Optimization ansatz ties these two together (but is *not* strongly local in the sense that computations depend on the size of the output).

Illustration on small graphs





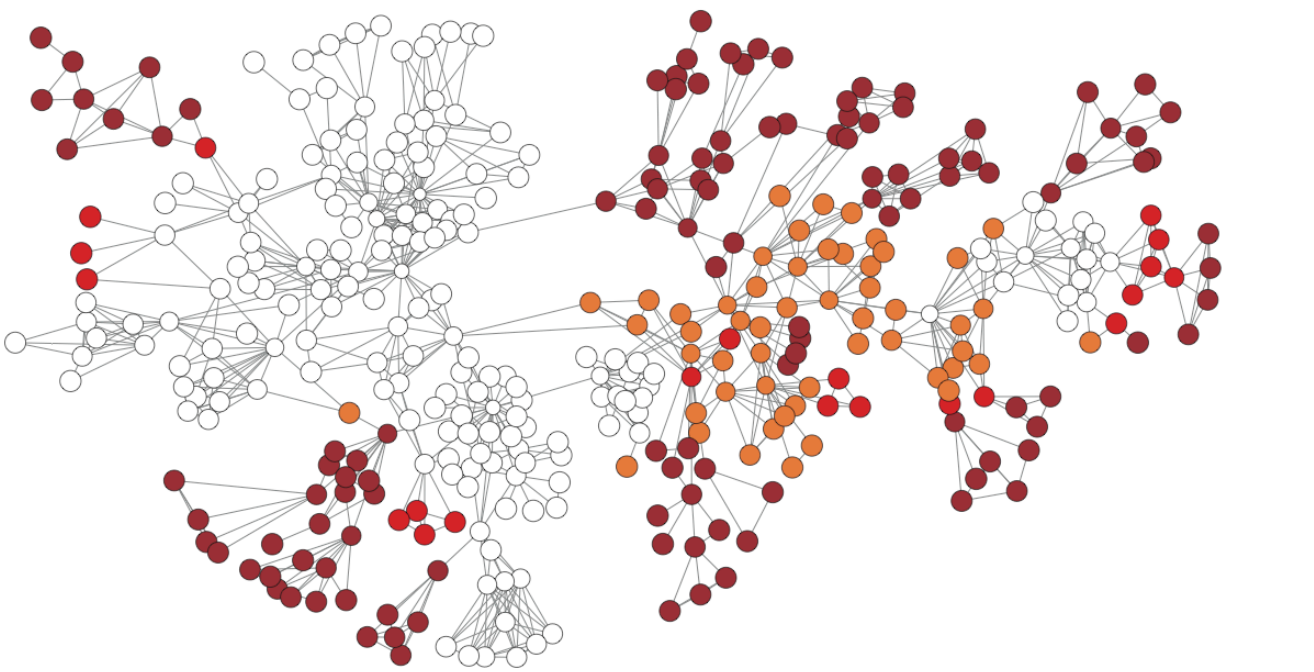




- Similar results if we do local random walks, truncated PageRank, and heat kernel diffusions.
- Often, it finds "worse" quality but "nicer" partitions than flow-improve methods. (Tradeoff we'll see later.)

Illustration with general seeds

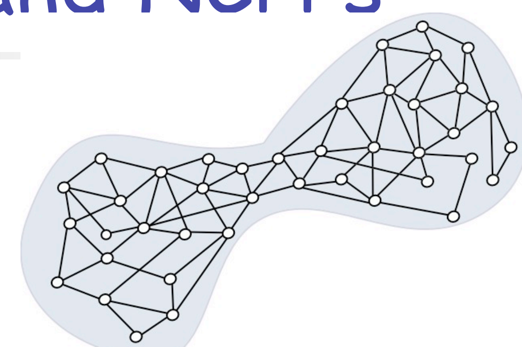
- Seed vector doesn't need to correspond to cuts.
- It could be any vector on the nodes, e.g., can find a cut "near" low-degree vertices with $s_i = -(d_i - d_{av})$, $i \varepsilon [n]$.



Overview

- Large networks and different perspectives on data
- Approximation algorithms as “experimental probes”
    - Graph partitioning: good test case for different approaches to data
    - Geometric/statistical properties *implicit* in worst-case algorithms
- An example of the theory
    - Local spectral graph partitioning as an optimization problem
    - Exploring data graphs locally: practice follows theory closely
- An example of the practice
    - Local and global clustering structure in very large networks
    - Strong theory allows us to make *very* strong applied claims

Conductance, Communities, and NCPPs



Let A be the adjacency matrix of G=(V,E).

The conductance ϕ of a set S of nodes is:

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

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

The Network Community Profile (NCP) Plot of the graph is:

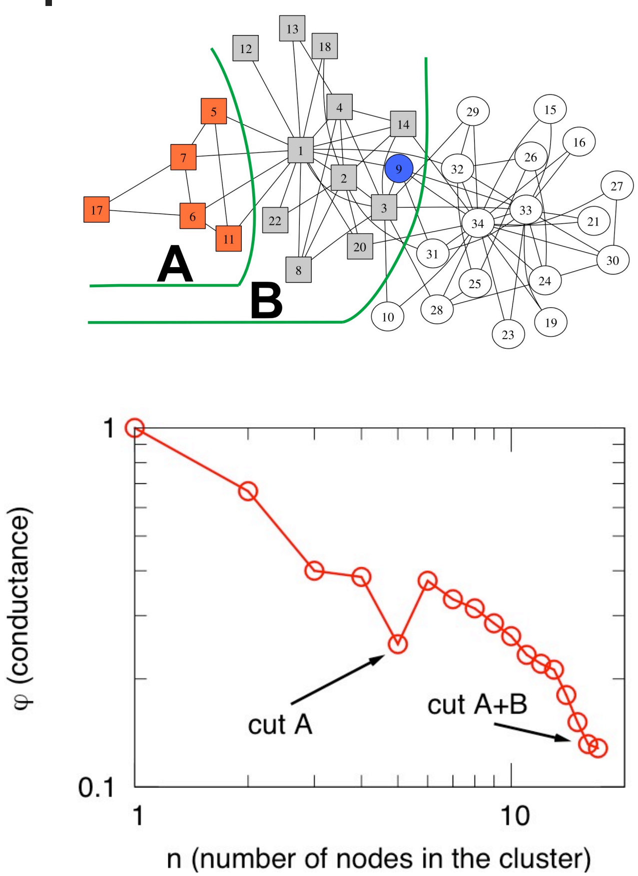
$$\Phi(k) = \min_{S \subset V, |S|=k} \phi(S)$$

Since algorithms often have non-obvious size-dependent behavior.

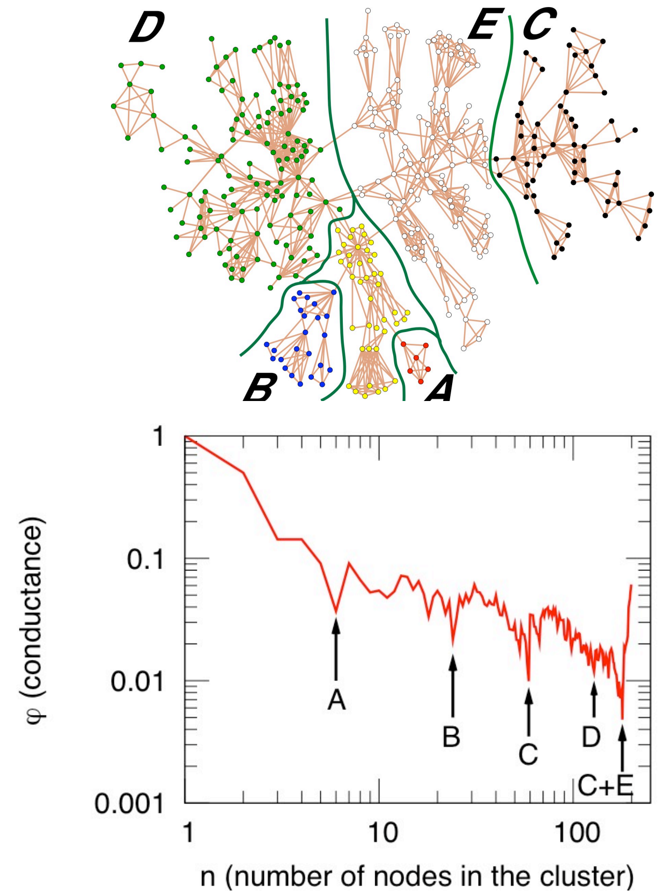
*Just as conductance captures the "gestalt" notion of cluster/community quality, the NCP plot measures cluster/community quality **as a function of size**.*

NCP is intractable to compute --> use approximation algorithms!

Widely-studied small social networks

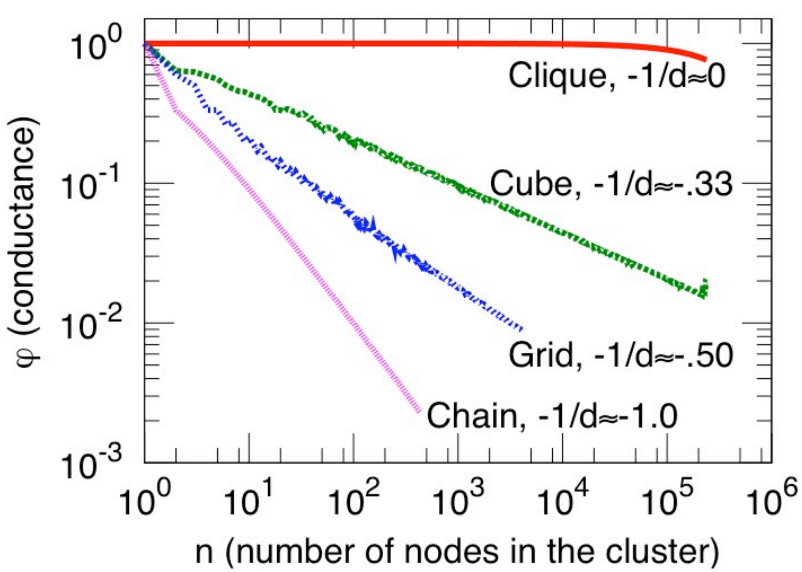


Zachary's karate club

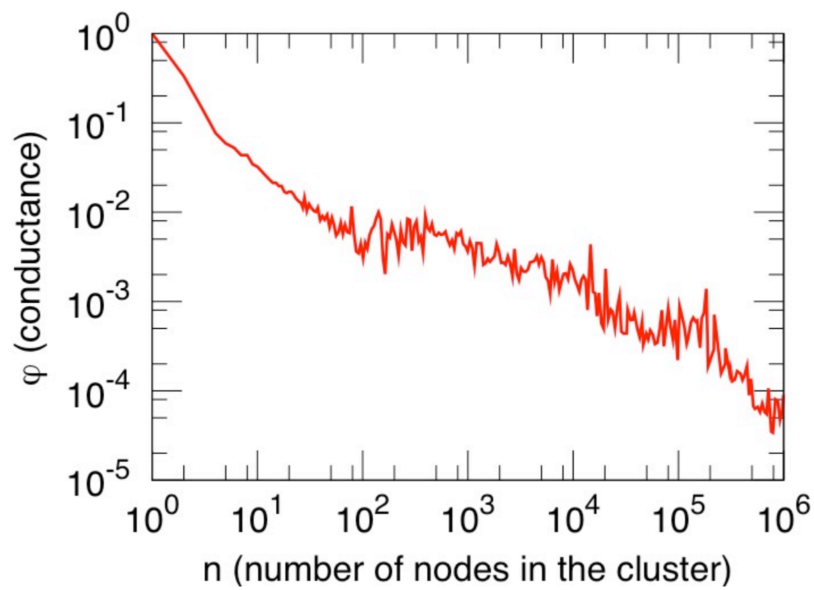


Newman's Network Science

"Low-dimensional" graphs (and expanders)

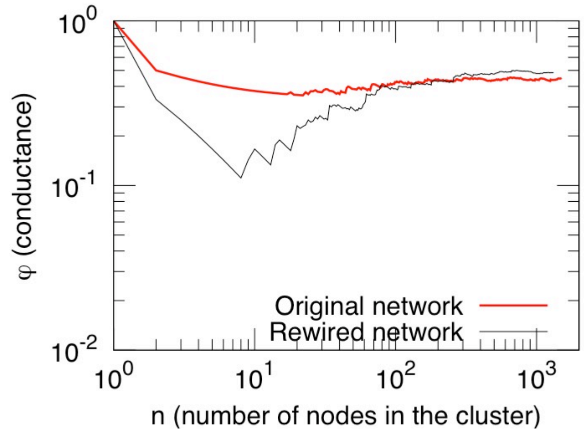


d-dimensional meshes

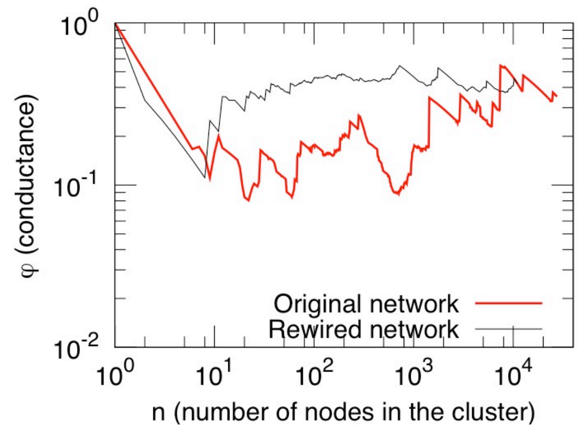


RoadNet-CA

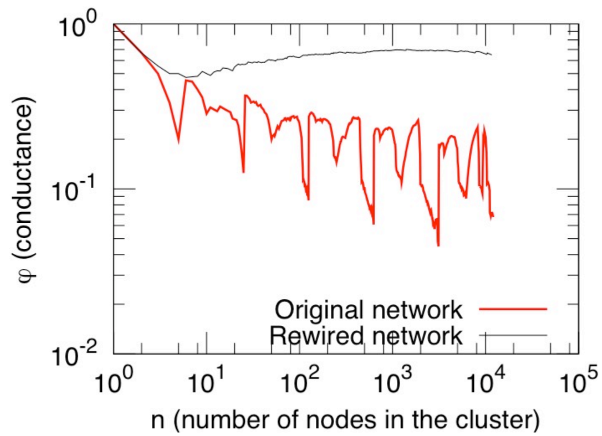
NCPP for common generative models



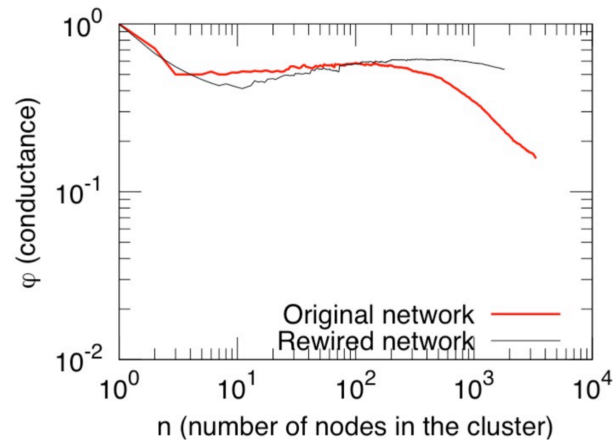
Preferential Attachment



Copying Model



RB Hierarchical



Geometric PA

Large Social and Information Networks

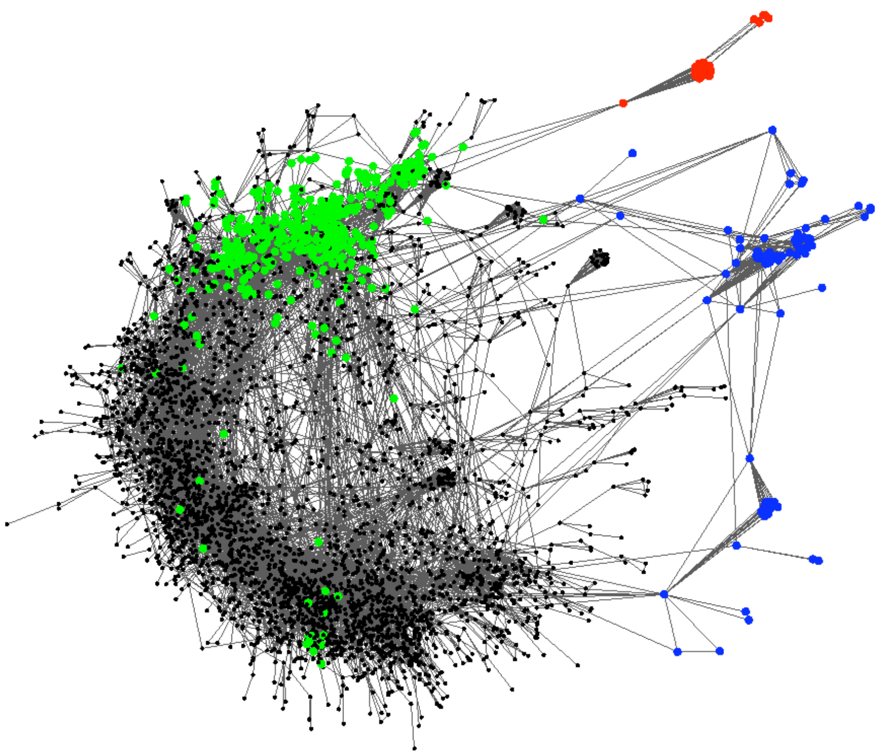
• Social nets	Nodes	Edges	Description
LiveJournal	4,843,953	42,845,684	Blog friendships [4]
Epinions	75,877	405,739	Who-trusts-whom [35]
Flickr	404,733	2,110,078	Photo sharing [21]
Delicious	147,567	301,921	Collaborative tagging
CA-DBLP	317,080	1,049,866	Co-authorship (CA) [4]
CA-cond-mat	21,363	91,286	CA cond-mat [25]
• Information networks			
Cit-hep-th	27,400	352,021	hep-th citations [13]
Blog-Posts	437,305	565,072	Blog post links [28]
• Web graphs			
Web-google	855,802	4,291,352	Web graph Google
Web-wt10g	1,458,316	6,225,033	TREC WT10G web
• Bipartite affiliation (authors-to-papers) networks			
AtP-DBLP	615,678	944,456	DBLP [25]
AtP-astro-ph	54,498	131,123	Arxiv `astro-ph` [25]
• Internet networks			
AS	6,474	12,572	Autonomous systems
Gnutella	62,561	147,878	P2P network [36]

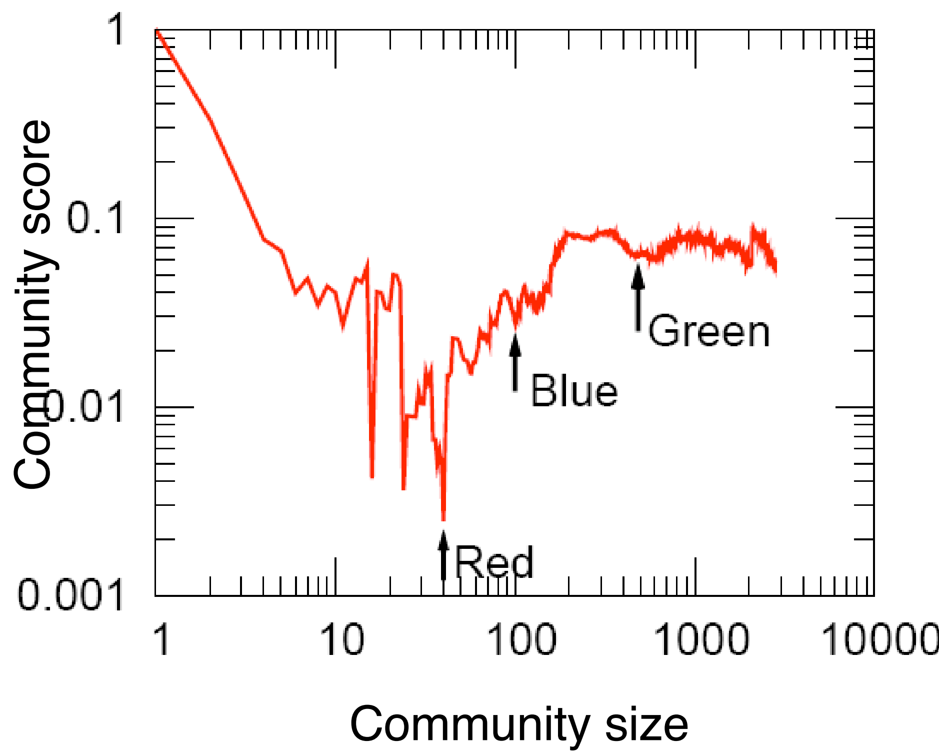
Table 1: Some of the network datasets we studied.

Typical example of our findings

Leskovec, Lang, Dasgupta, and Mahoney (WWW 2008 & arXiv 2008)

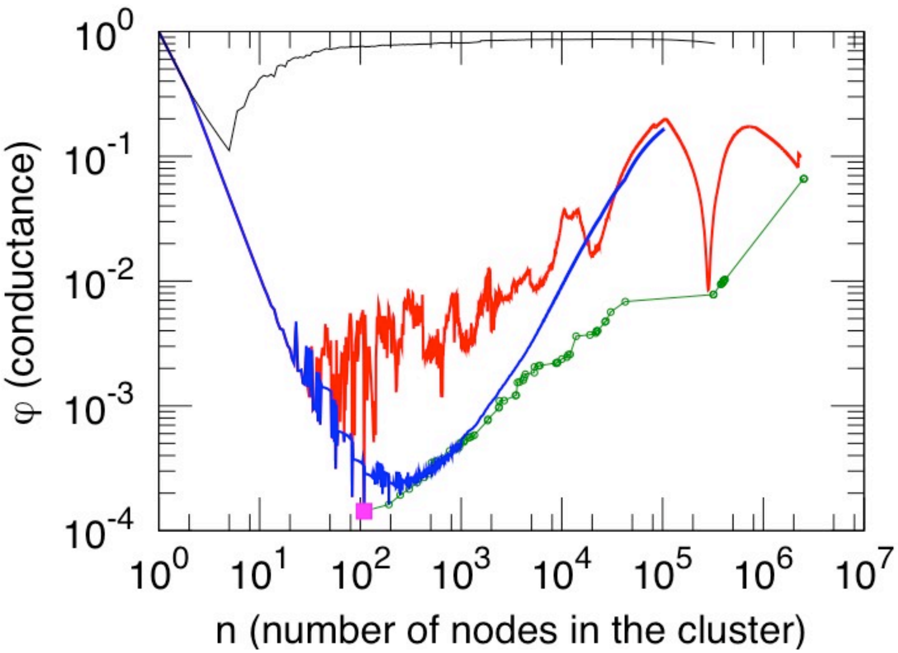
General relativity collaboration network
(4,158 nodes, 13,422 edges)



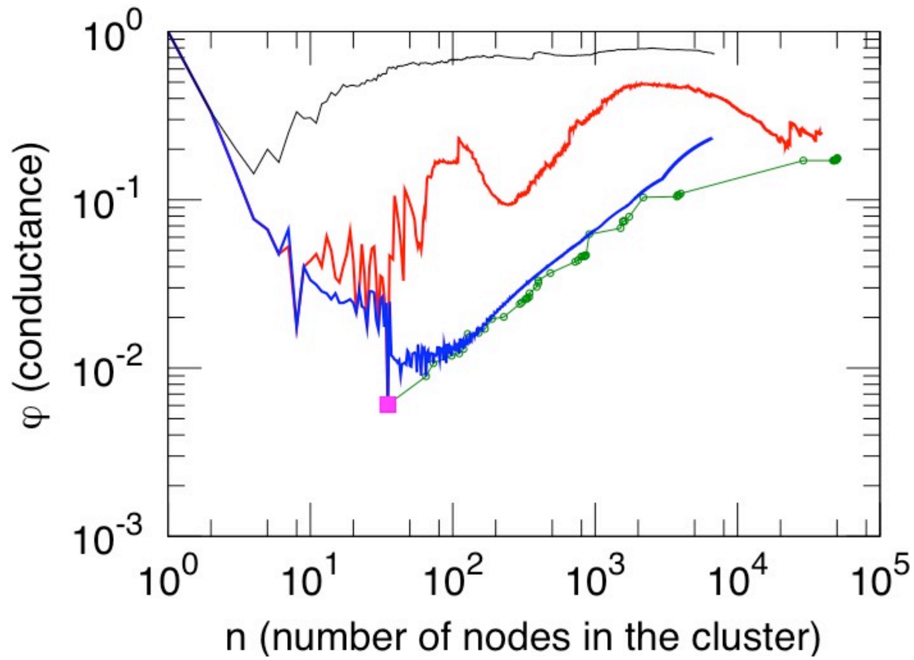


Large Social and Information Networks

Leskovec, Lang, Dasgupta, and Mahoney (WWW 2008 & arXiv 2008 & WWW 2010)



LiveJournal



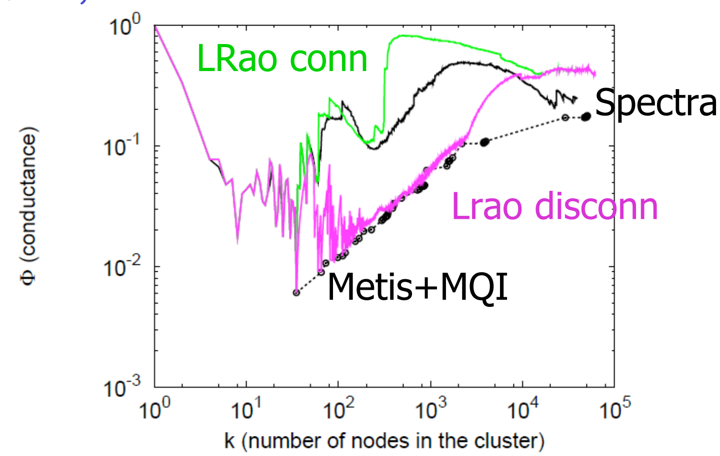
Epinions

Focus on the red curves (local spectral algorithm) - blue (Metis+Flow), green (Bag of whiskers), and black (randomly rewired network) for consistency and cross-validation.

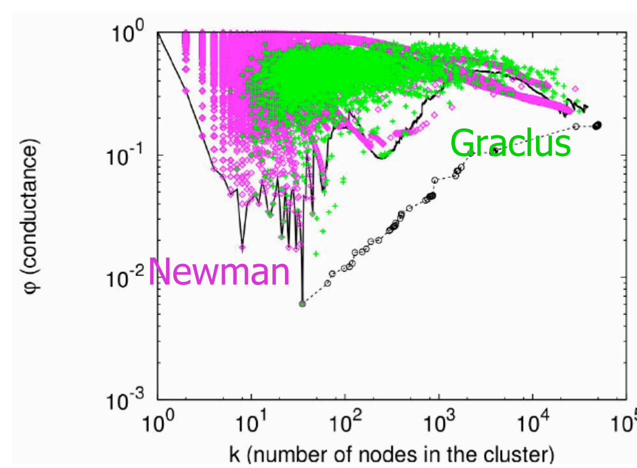
Other clustering methods

Leskovec, Lang, Dasgupta, and Mahoney (WWW 2008 & arXiv 2008 & WWW 2010)

- LeightonRao: based on multi-commodity flow
  - Disconnected clusters vs. Connected clusters

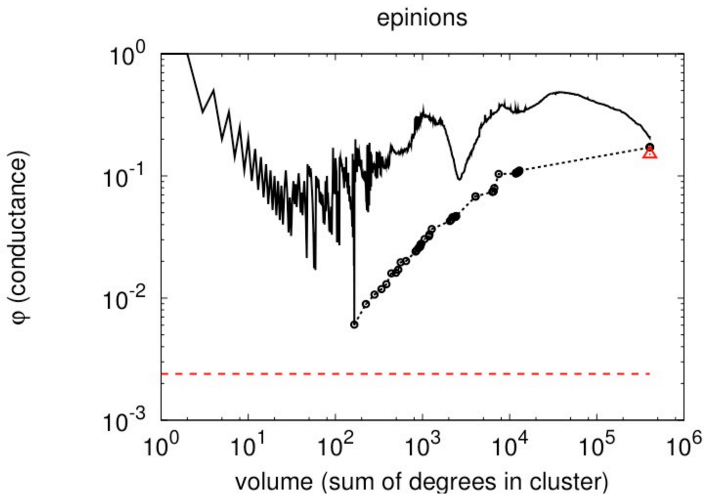


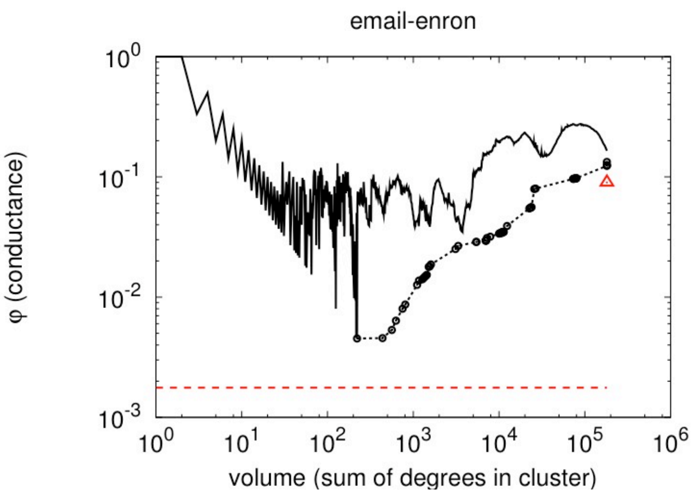
- Graclus prefers larger clusters
- Newman's modularity optimization similar to Local Spectral

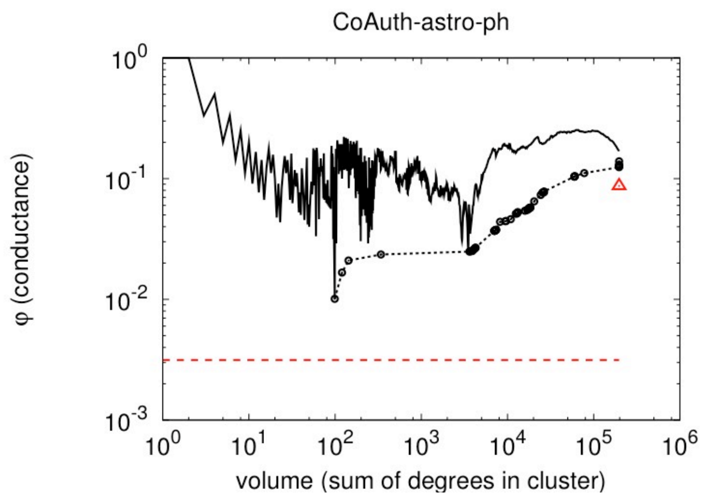


Lower and upper bounds

- Lower bounds on conductance can be computed from:
  - Spectral embedding (independent of balance)
  - SDP-based methods (for volume-balanced partitions)
- Algorithms find clusters close to theoretical lower bounds



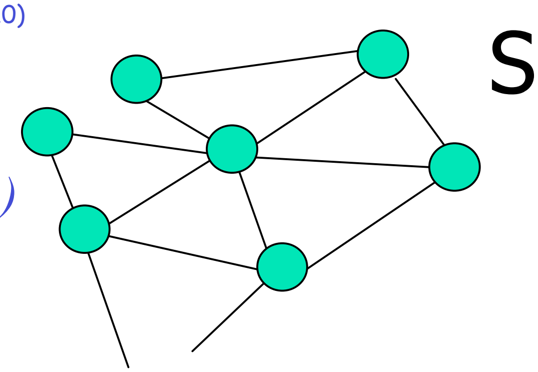




12 clustering objective functions*

Leskovec, Lang, Dasgupta, and Mahoney (WWW 2008 & arXiv 2008 & WWW 2010)

- Clustering objectives:
  - Single-criterion:
    - Modularity: *m-E(m)* *(Volume minus correction)*
    - Modularity Ratio: *m-E(m)*
    - Volume: $\sum_u d(u) = 2m+c$
    - Edges cut: *c*
  - Multi-criterion:
    - Conductance: $c/(2m+c)$ *(SA to Volume)*
    - Expansion: c/n
    - Density: $1-m/n^2$
    - CutRatio: $c/n(N-n)$
    - Normalized Cut: $c/(2m+c) + c/2(M-m)+c$
    - Max ODF: *max frac. of edges of a node pointing outside S*
    - Average-ODF: *avg. frac. of edges of a node pointing outside*
    - Flake-ODF: *frac. of nodes with mode than _ edges inside*

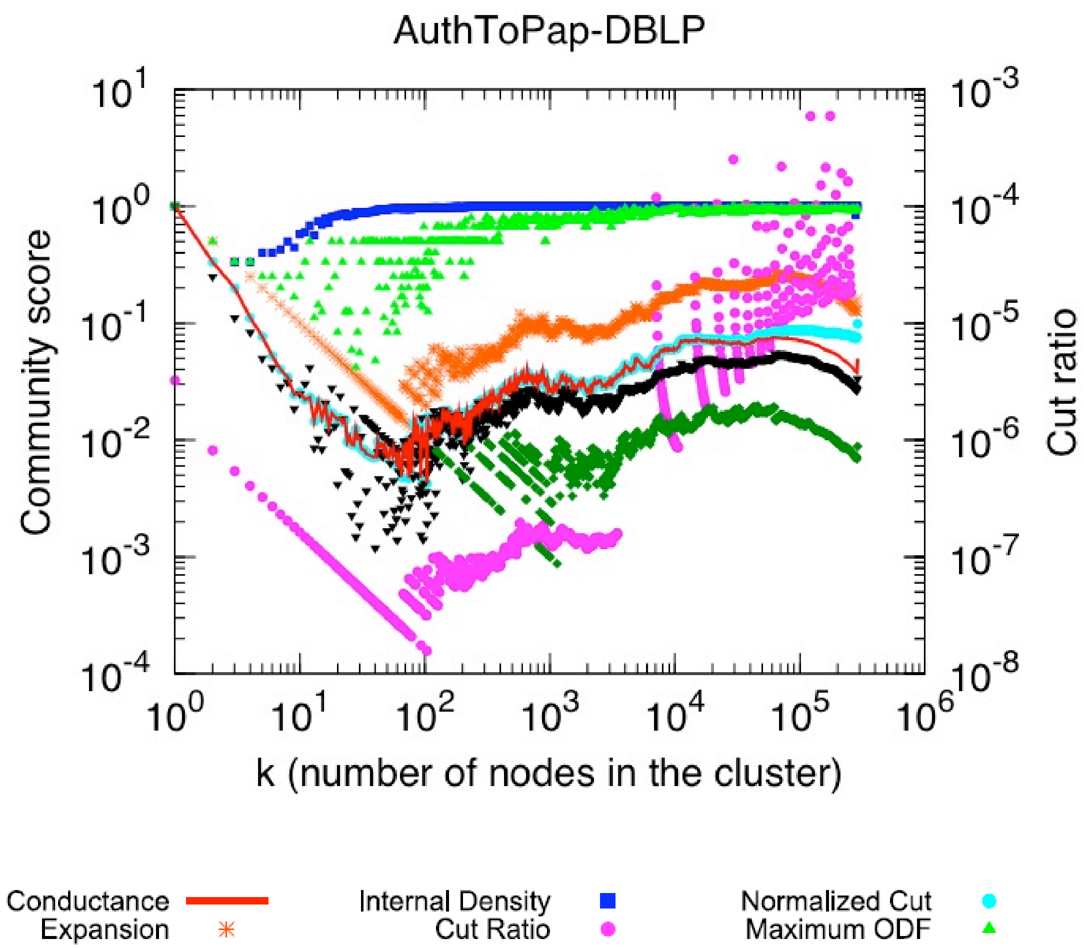


n: nodes in S
m: edges in S
c: edges pointing outside S

*Many of hese typically come with a weaker theoretical understanding than conductance, but are similar/different in known ways for practitioners.

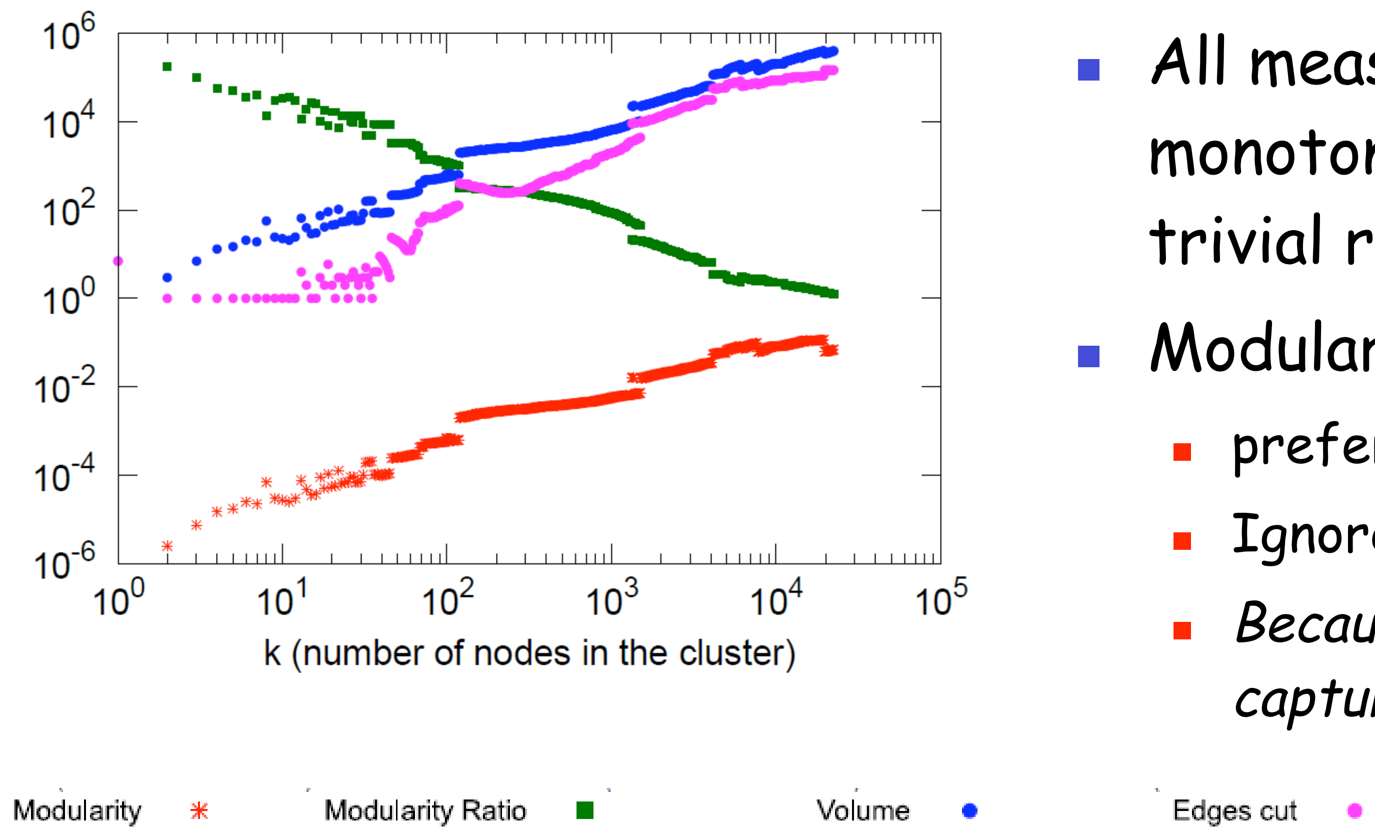
Multi-criterion objectives

Leskovec, Lang, Dasgupta, and Mahoney (WWW 2008 & arXiv 2008 & WWW 2010)



- Qualitatively similar to conductance
- Observations:
  - Conductance, Expansion, NCut, Cut-ratio and Avg-ODF are similar
  - Max-ODF prefers smaller clusters
  - Flake-ODF prefers larger clusters
  - Internal density is bad
  - Cut-ratio has high variance

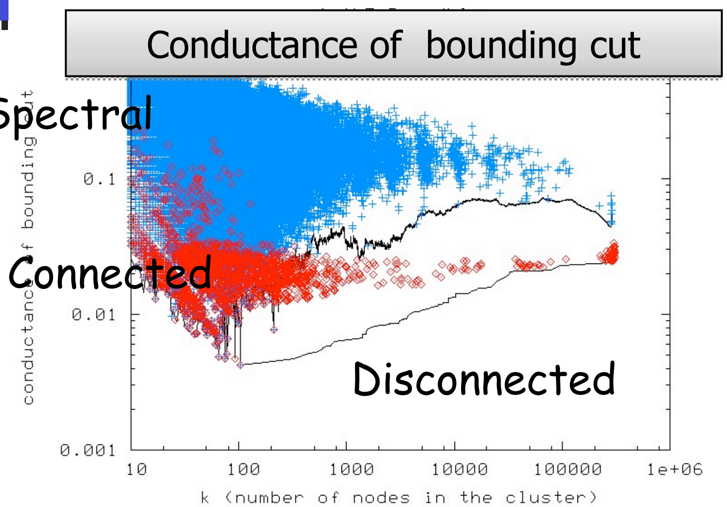
Single-criterion objectives

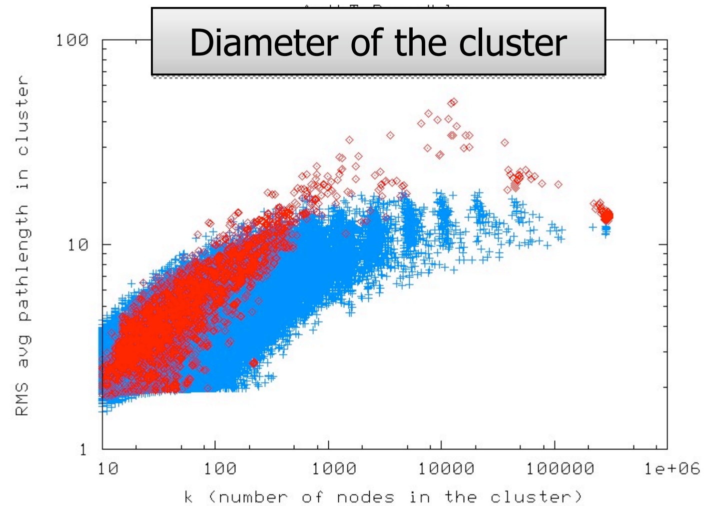


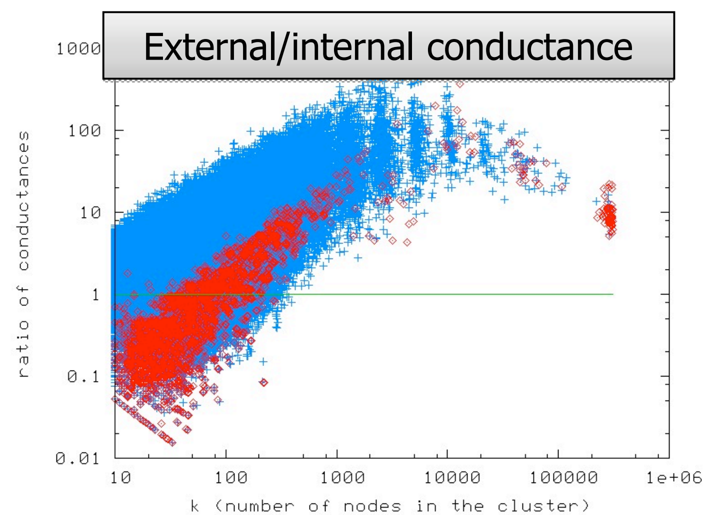
Observations:

- All measures are monotonic (for rather trivial reasons)
- Modularity
  - prefers large clusters
  - Ignores small clusters
  - *Because it basically captures Volume!*

Regularized and non-regularized communities (1 of 2)





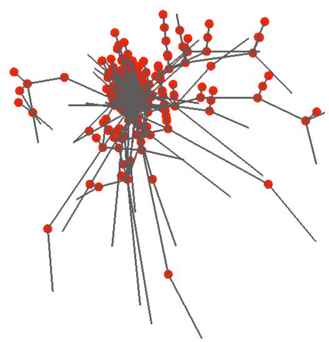


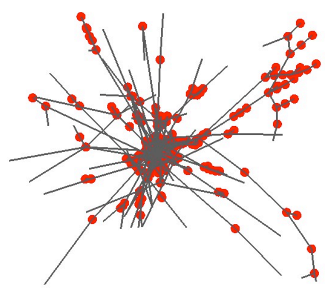
Lower is good

- Metis+MQI (red) gives sets with better conductance.
- Local Spectral (blue) gives tighter and more well-rounded sets.
- **Regularization is *implicit* in the steps of approximation algorithm.**

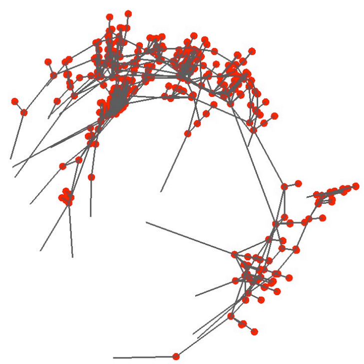
Regularized and non-regularized communities (2 of 2)

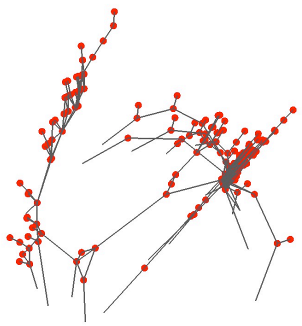
Two ca. 500 node communities from Local Spectral Algorithm:





Two ca. 500 node communities from Metis+MQI:





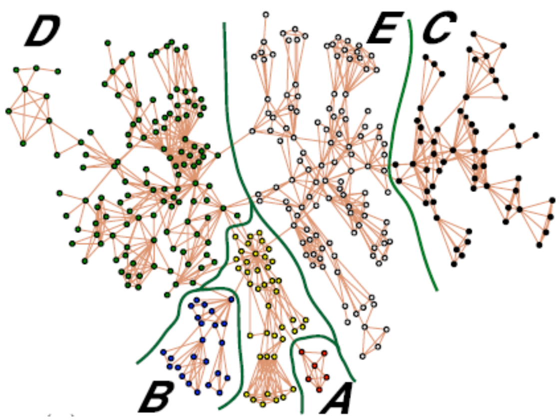
α	β
β	γ

Small versus Large Networks

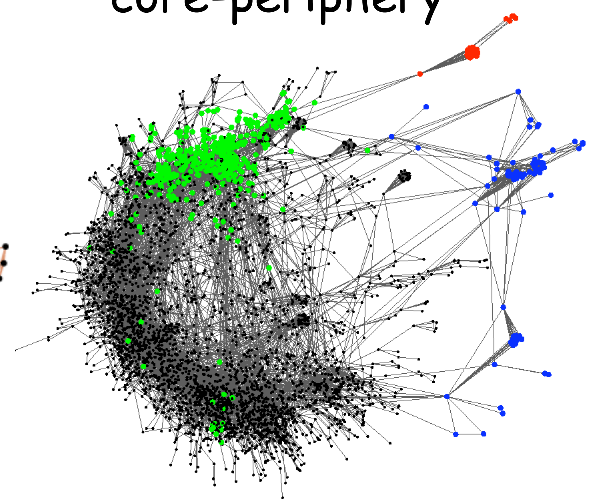
Leskovec, et al. (arXiv 2009); Mahdian-Xu 2007

- Small and large networks are very different:

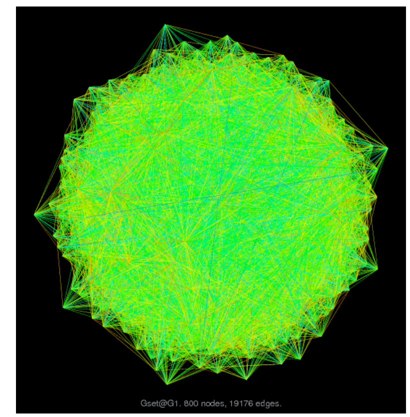
"low-dimensional"



core-periphery



(also, an expander)



E.g., fit these networks to Stochastic Kronecker Graph with "base" K=[a b; b c]:

$K_1 =$ 

Implications

Relationship b/w small-scale structure and large-scale structure in social/information networks is not reproduced (even qualitatively) by popular models

- This relationship governs many things: diffusion of information; routing and decentralized search; dynamic properties; etc., etc., etc.
- This relationship also governs (implicitly) the applicability of nearly every common data analysis tool in these applications
- Local structures are locally "linear" or meaningfully-Euclidean -- do not propagate to more expander-like or hyperbolic global size-scales
- Good large "communities" (as usually conceptualized i.t.o. inter- versus intra- connectivity) don't really exist

Conclusions

Approximation algorithms as “experimental probes”:

• Geometric and statistical properties *implicit* in worst-case approximation algorithms - based on very strong theory

• Graph partitioning is good “hydrogen atom” - for understanding algorithmic versus statistical perspectives more generally

Applications to network data:

• Local-to-global properties not even qualitatively correct in existing models, graphs used for validation, intuition, etc.

• Informatics graphs are good “hydrogen atom” for development of *geometric network analysis tools* more generally