Cycles, Communities, Ergodicity and Communications in Network Topologies

Erik Bollt, Clarkson University
bolltem@clarkson.edu
http://www.clarkson.edu/~bolltem

Dani ben-Avraham, Clarkson U.
Joe Kirk, Clarkson U.
Hernan Rozenfeld, Clarkson U.
Jim Bagrow, Clarkson U.

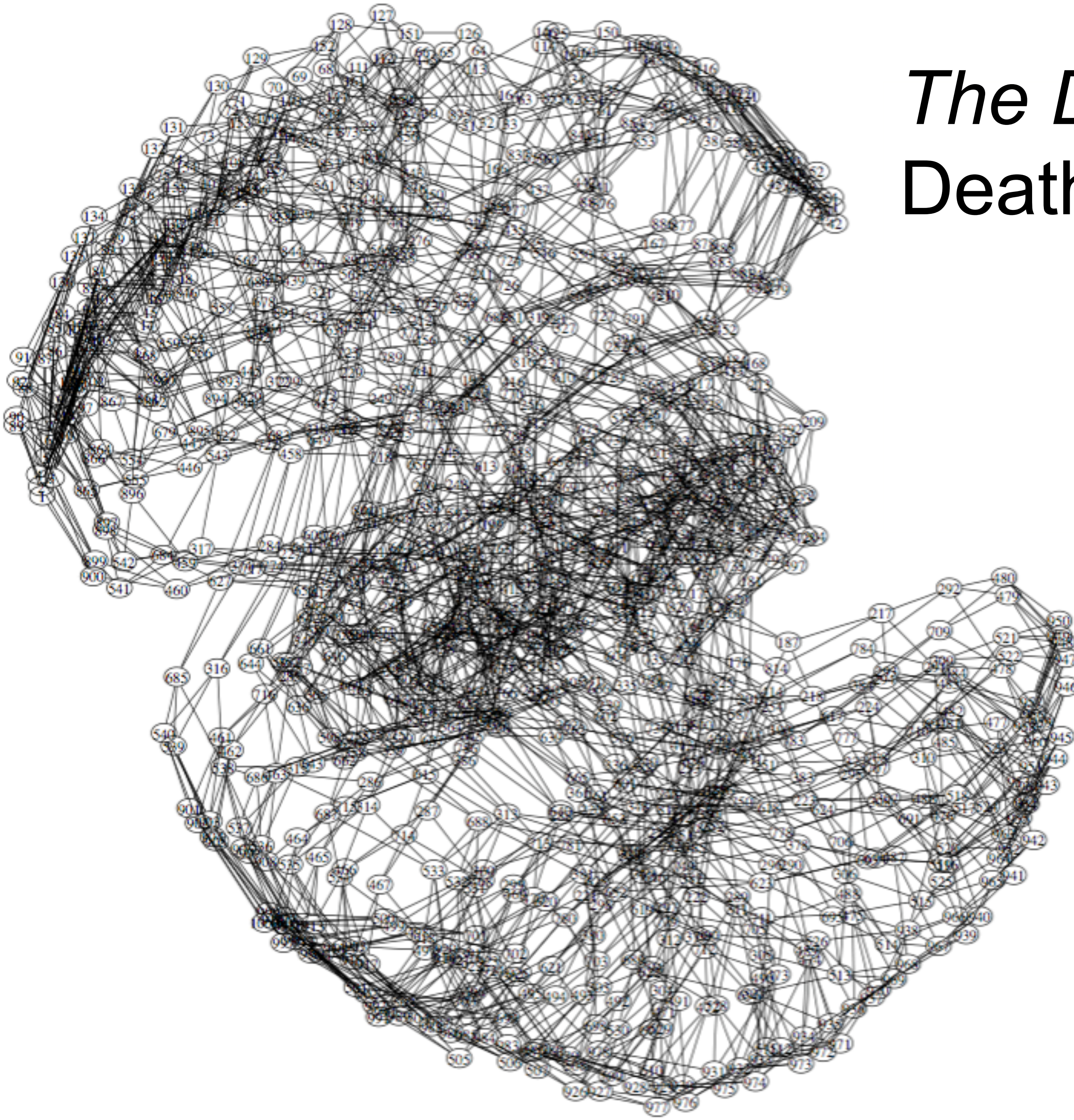
Joe Skufca, USNA, U. MD, Clarkson U.
Dan Stilwell, Virginia Tech.
D. Gray Roberson, Virginia Tech.

Structure of Talk

1. About Community Partitions and Finding them.

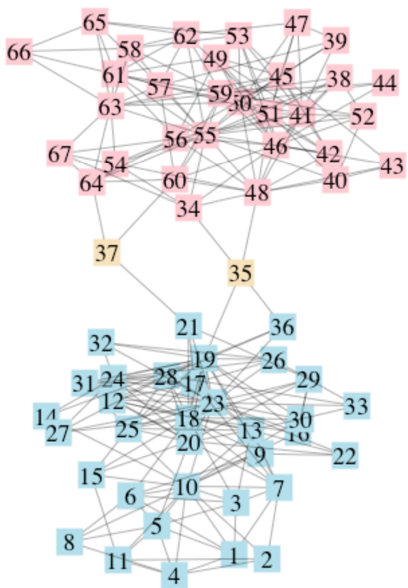
2. About the Loopy Cycles Backbone.

3. About information propagation and sufficiency in dynamically evolving networks.



The Dreaded
Death Star

What are Communities?



- Groups of vertices that are densely connected amongst themselves while being loosely connected to the rest of the graph
- Detection can be difficult (expensive)

Why Find Communities?

- Social networks, epidemiology
- Parallel computing, VLSI
- Drug dependency in protein-protein interaction networks
- etc

..and visualizing interrelationships

Detection Methods

- Most are either agglomerative, divisive, or spectral

1. rewiring the network from scratch while optimizing some parameter
2. selectively removing edges, splitting the network apart
3. analysis of graph Laplacian

- Our local method is unlike these

3. Spectral Bisection

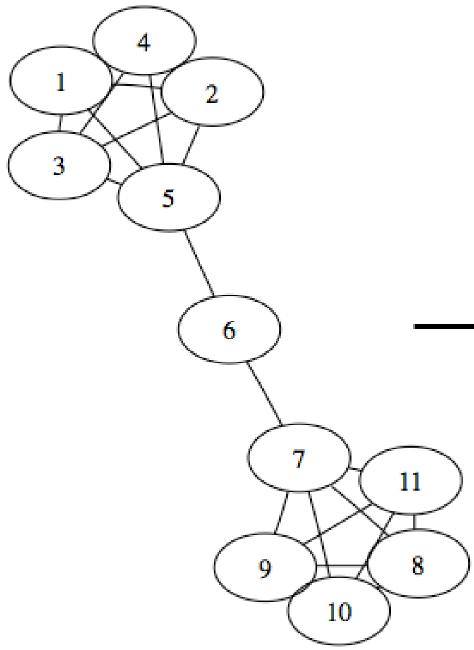
- Fiedler, 1970s. Pothen, Simon, Liou, 1990.
- Use eigenvalues and eigenvectors of graph Laplacian to split graph in half

graph Laplacian, $L(G)$

$$L = diag(d) - A$$

- $L(G)_{i,i}$ = degree of node i
- $L(G)_{i,j} = -1$ if $i \neq j$ and there is an edge (i,j)
- $L(G)_{i,j} = 0$ otherwise

Example $L(G)$



4	-1	-1	-1	-1	0	0	0	0	0	0
-1	4	-1	-1	-1	0	0	0	0	0	0
-1	-1	4	-1	-1	0	0	0	0	0	0
-1	-1	-1	4	-1	0	0	0	0	0	0
-1	-1	-1	-1	5	-1	0	0	0	0	0
0	0	0	0	-1	2	-1	0	0	0	0
0	0	0	0	0	-1	5	-1	-1	-1	-1
0	0	0	0	0	0	-1	4	-1	-1	-1
0	0	0	0	0	0	-1	-1	4	-1	-1
0	0	0	0	0	0	-1	-1	-1	4	-1
0	0	0	0	0	0	-1	-1	-1	-1	4

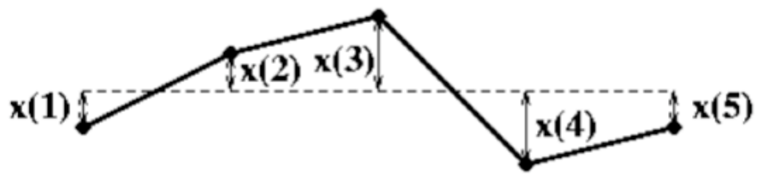
Some Properties of *L(G)*

- Symmetric
- eigenvalues are non-negative.
  - First eigenvalue always zero
- number of connected components in graph equals number of eigenvalues equal to zero
  - $\lambda_2 \neq 0$ i.f.f. G is connected

Partition based on 2nd eigenvector (v_2)

- For each node n in G:
  - if $v_2(n) < 0$, put n in partition N_-
  - else put n in partition N_+

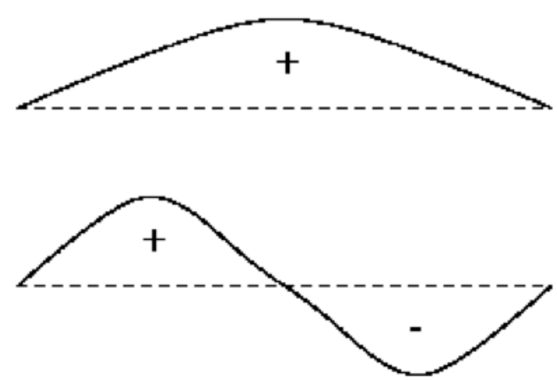
Motivation: vibrating string analogy



$$m\ddot{u}(x,t) = -k\Delta u(x,t)$$

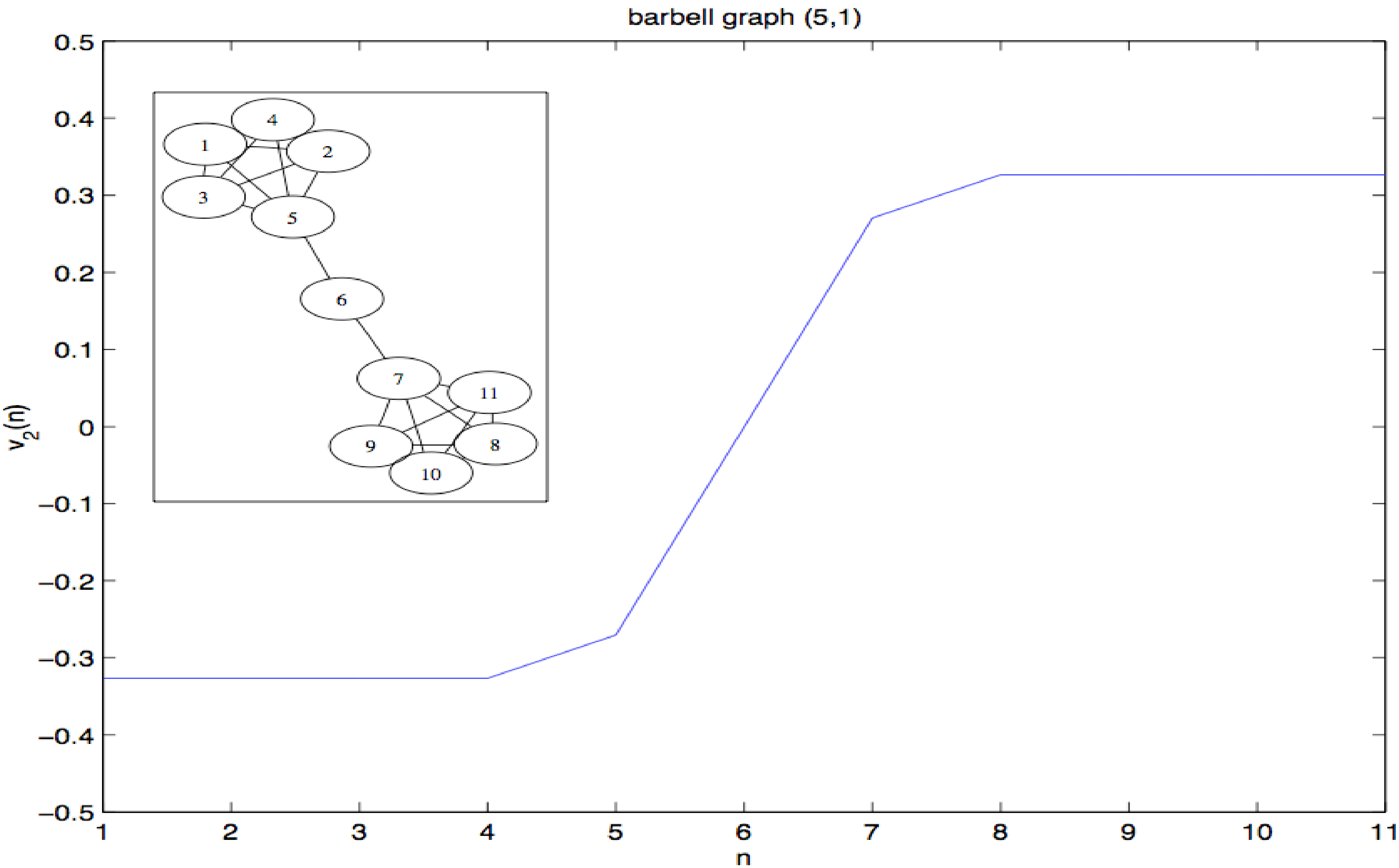
$$m\frac{d^2}{dt^2}\begin{pmatrix} u_1 \\ u_2 \\ \dots \\ u_j \\ \dots \\ u_n \end{pmatrix} = -k\begin{pmatrix} 2u_1 - u_2 \\ -u_1 + 2u_2 - u_3 \\ \dots \\ -u_{j-1} + 2u_j - u_{j+1} \\ \dots \\ 2u_{n-1} + u_n \end{pmatrix} = k\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & & \dots & & \\ & & & -1 & 2 \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \\ \dots \\ u_j \\ \dots \\ u_n \end{pmatrix}$$

- split graph based on second mode



Lowest Frequency

Second Frequency

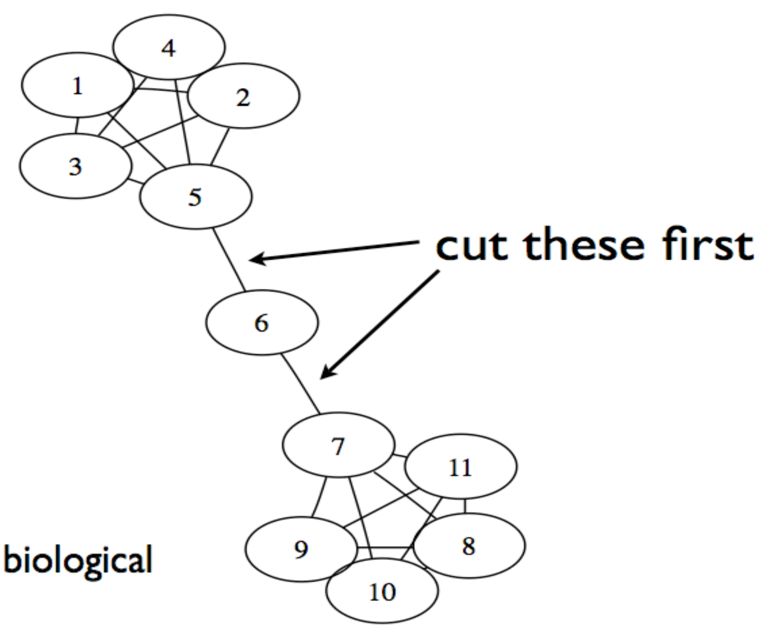


- Need entire Laplacian (could be big)
- Splits graph into two (equal) partitions
  - may not be natural

2. Divisive Methods

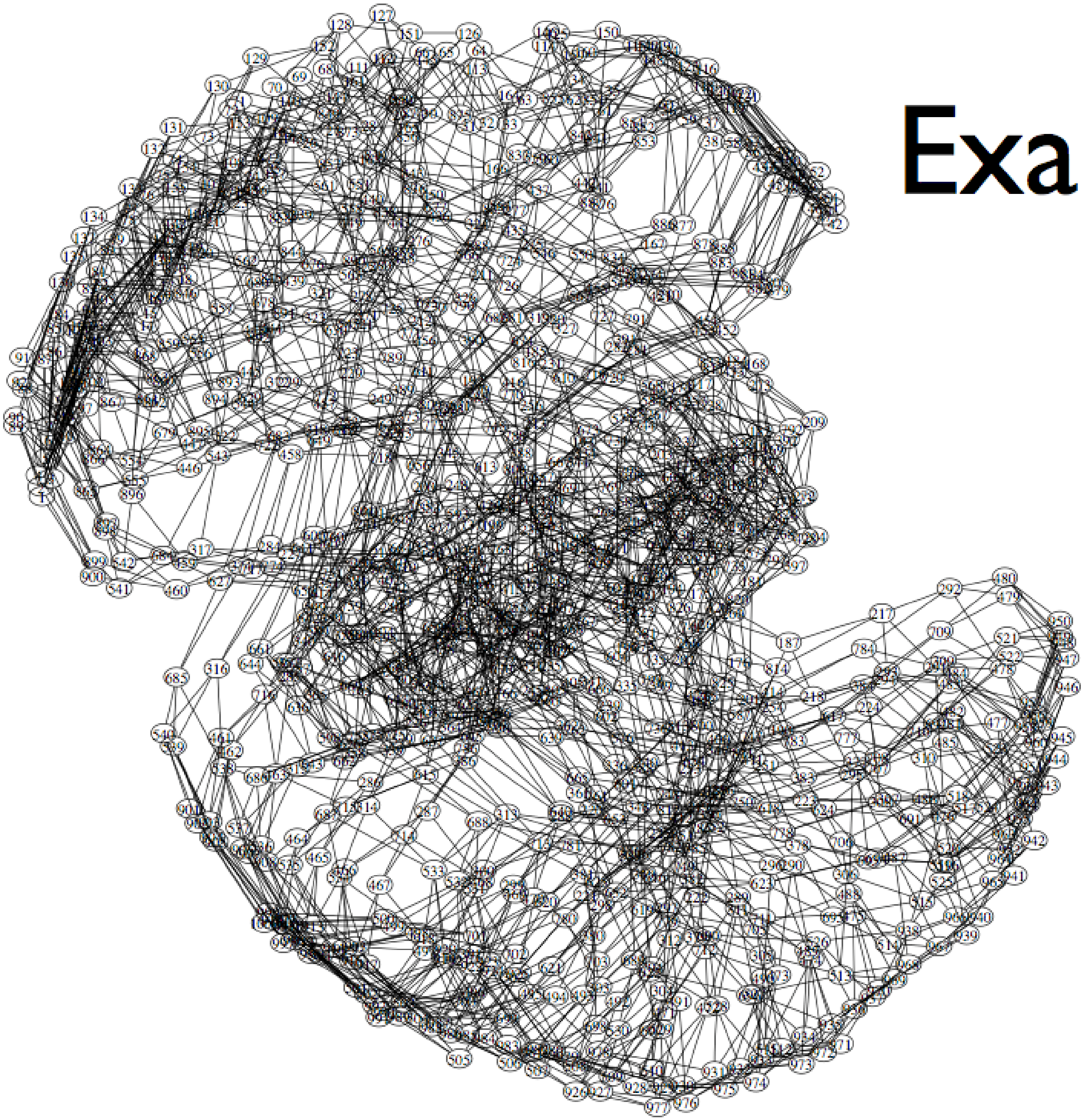
- Cut edges with maximum centrality to split graph. Repeat
- Edge betweenness most popular measure
  - number of times an edge is used in all-pairs shortest paths
- expensive, $O(n^3)$ at best

Maximum Betweenness

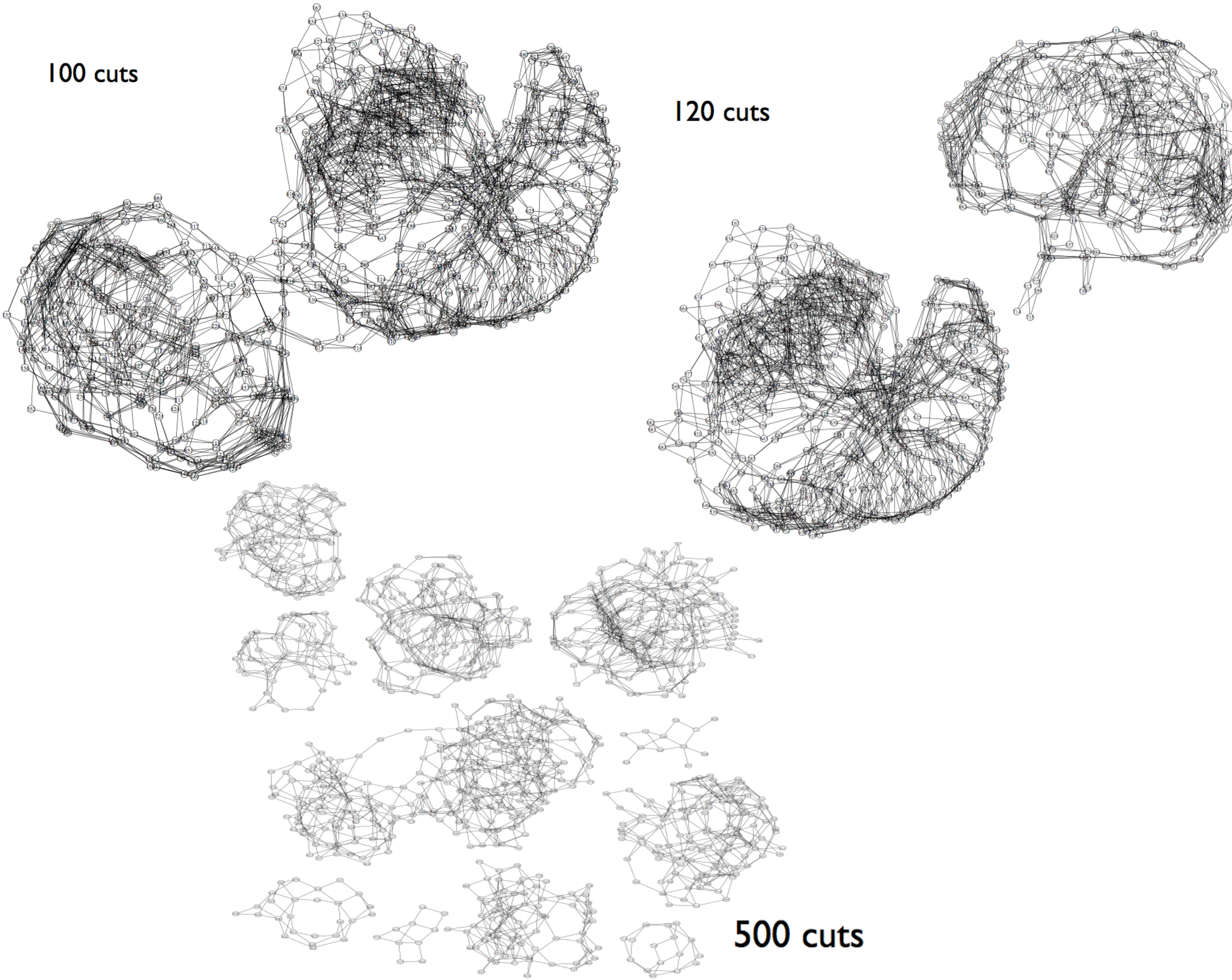


M. Girvan and M.E.J. Newman, Community structure in social and biological networks. Proc. Natl. Acad. Sci. USA 99, 7821-7826 (2002).

Example



0 cuts


100 cuts
120 cuts
500 cuts

1. Agglomerative Methods

- Start with the empty graph, adding edges that maximize some statistic at each step
- Modularity: most common statistic
- Can be very cheap

Modularity, Q

- Property of graph and a given partition
  - evaluates a partitioning, doesn't find it
- Measure number of within-community edges
  - compared to randomly distributed edges

A. Clauset, M. Newman, C. Moore. Finding community structure in very large networks. Phys. Rev. E 70, 066111 (2004).

Modularity, notation

Adjacency Matrix:

$$A_{vw} = \begin{cases} 1 & \text{if vertices v and w are connected,} \\ 0 & \text{otherwise} \end{cases}$$

Partition:

vertices divided such that vertex *v* belongs to community c_v

Total Number of edges: $$m = \frac{1}{2}\sum_{vw} A_{vw}$$

Degree of vertex *v*: $$k_v = \sum_{w} A_{vw}$$

then, the fraction of edges within communities:

$$\frac{\sum_{vw} A_{vw}\delta(c_v, c_w)}{\sum_{vw} A_{vw}} = \frac{1}{2m}\sum_{vw} A_{vw}\delta(c_v, c_w)$$

This value is large for good divisions (many in-community edges) but is not, on its own, a good measure

It takes its largest value of 1 in the trivial case where all vertices belong to a single community

To correct this: subtract the expected value of the same quantity in the case of a randomized network

The probability of an edge existing between vertices *v* and *w* if connections are made at random (respecting vertex degrees)

$$= \frac{k_v k_w}{2m}$$

Then the modularity is defined to be:

$$Q = \frac{1}{2m} \sum_{vw} \left(A_{vw} - \frac{k_v k_w}{2m} \right) \delta(c_v, c_w)$$

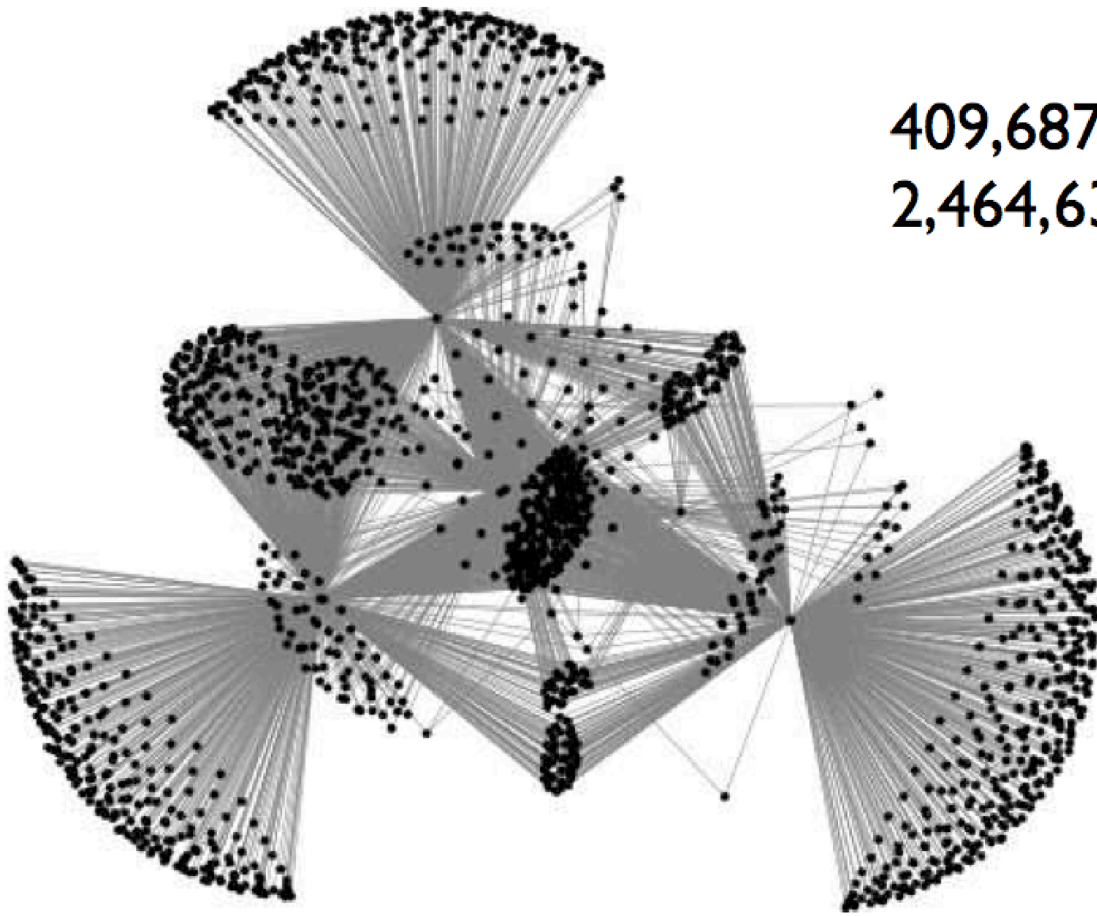
If the fraction of within-community edges is no different from what's expected for a randomized network, then Q will be zero.

Non-zero values represent deviations from randomness. In practice, values above ~ 0.3 show significant community structure

Using Q to get partition

- Start with empty graph
- Add edge (i,j) that leads to maximum ΔQ
- Repeat for next edge
- Can be very efficient: $O(n^2) \rightarrow O(n \cdot \log n)$

Example: Amazon.com



409,687 nodes
2,464,630 edges

Our local method

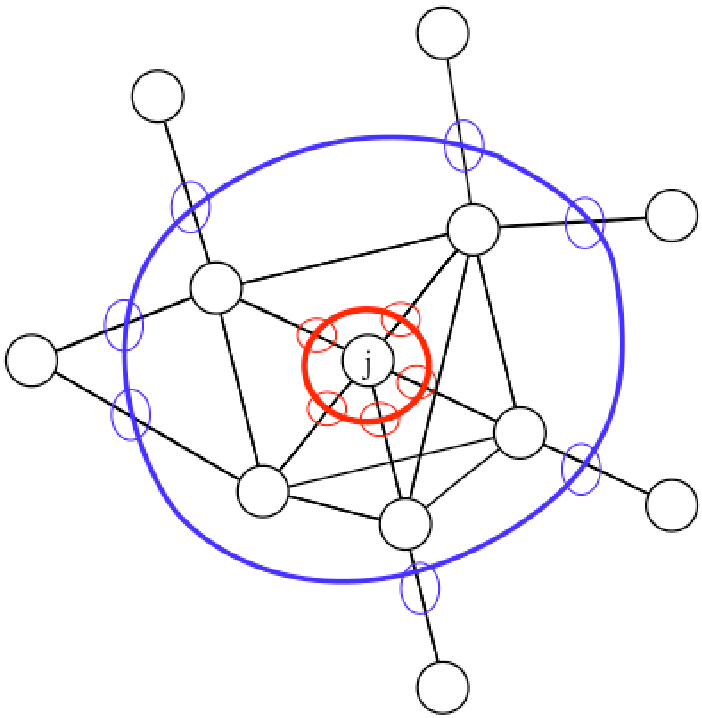
- Choose a starting vertex
- Spread outward until some criteria has been met, then stop
- Does not require knowledge of graph beyond spread

J. P. Bagrow and E. M. Bollt. A local method for detecting communities. arXiv/cond-mat:0412482, 2004.

Notation

$K_j^l \equiv$ Total Emerging Degree of an l-shell starting from vertex j

$$\Delta K_j^l = \frac{K_j^l}{K_j^{l-1}}$$



- $K_j^0 = 5$
- $K_j^1 = 7$
- $K_j^2 = 0$

$$\Delta K_j^1 = \frac{7}{5}$$

$$\Delta K_j^2 = \frac{0}{7}$$

The Algorithm

- Choose a starting vertex j
- Center an l-shell around j (l=0) and spread outward
  - compute the total emerging degree at each depth of the l-shell
- Continue until the current Change in Total Emerging Degree crosses some threshold
- All vertices covered by the last l-shell created are then members of vertex j's community

Stopping Criteria

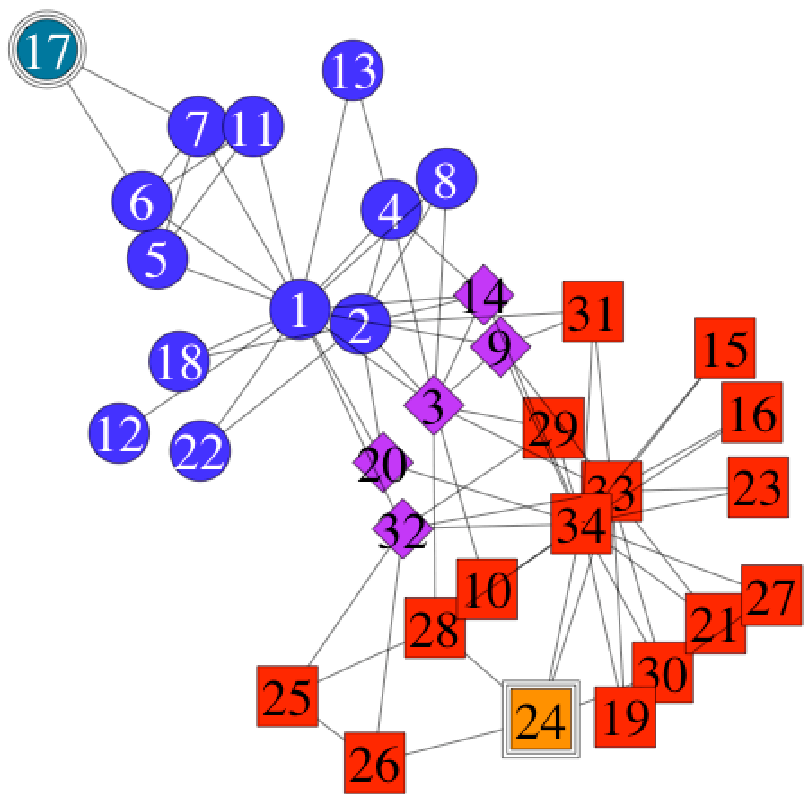
Stop when:

$$\Delta K_j^l < \alpha$$

where α is some arbitrary threshold, > 0

All vertices covered by shells of size l
or less are then members of j's community

Result: Karate Club



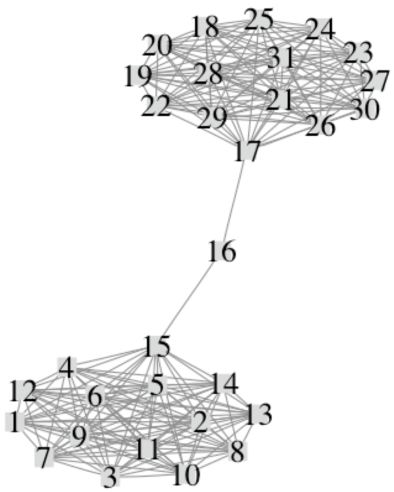
- Two Runs
  - Starting from vertices 17 and 24
- α = 1.9
- Indicates overlap vertices

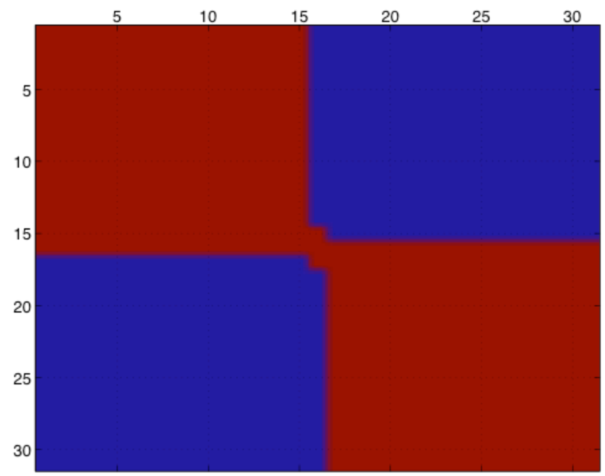
Much ado about **THE** right answer. Can the graph **data** tell us?

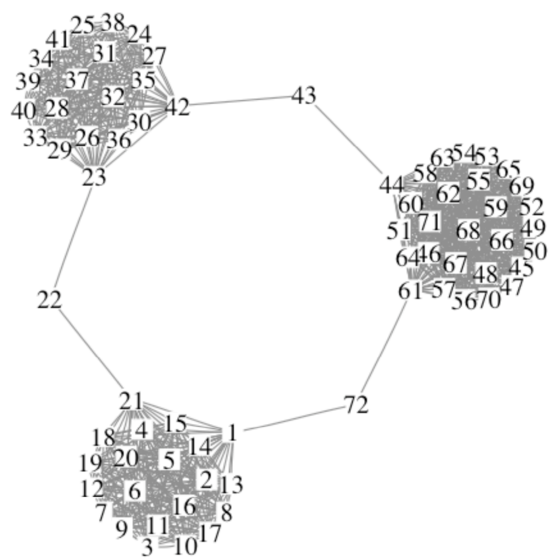
A Global Application

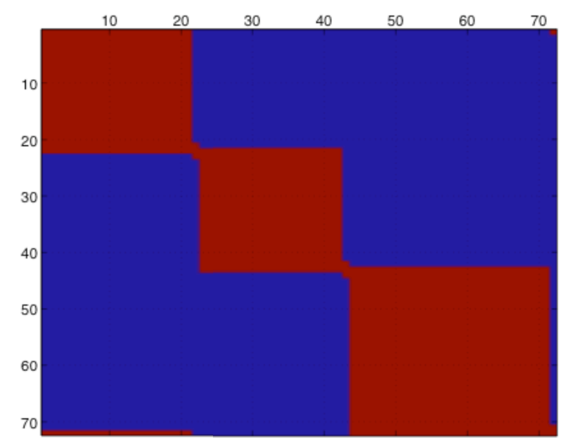
- Can start the algorithm from all n vertices
- Need some way to relate results from disparate runs
- Define Membership Matrix, M
  - M_{ij} = 1 if vertex j is a member of i's community
  - 0 otherwise
  - M is of size n x n for a network of n vertices

Some example M's



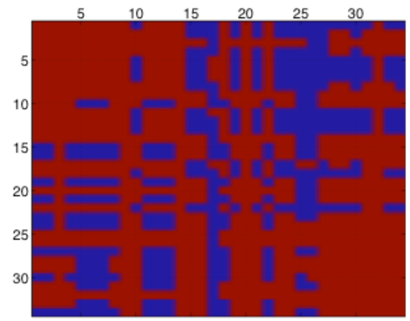






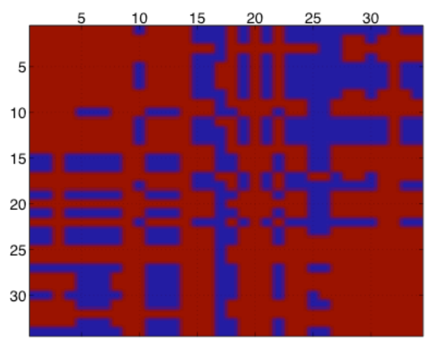
Problem with M

- These last results are trivial
  - the vertices are already numbered consecutively by community
- The Karate Club is not:

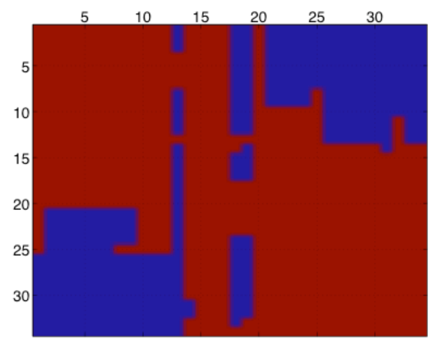


$$\alpha = 1.2$$

Sorting Example



$\longrightarrow$



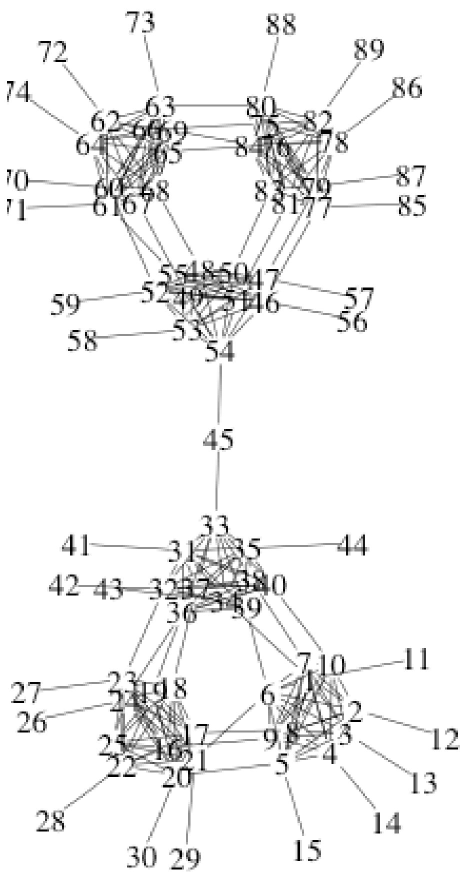
Solution: Sort M

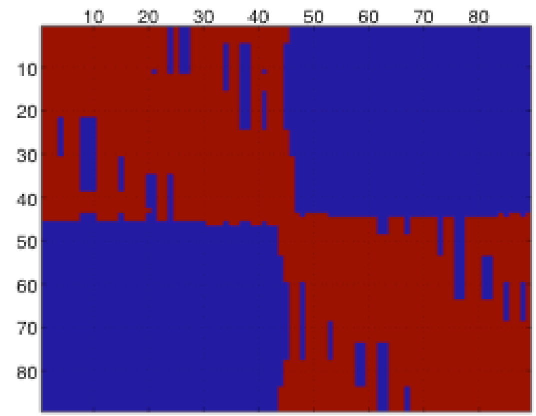
Define the distance between rows:

$$Distance(i, j) = n - \sum_{k=1}^{n} \delta(M_{ik}, M_{jk})$$

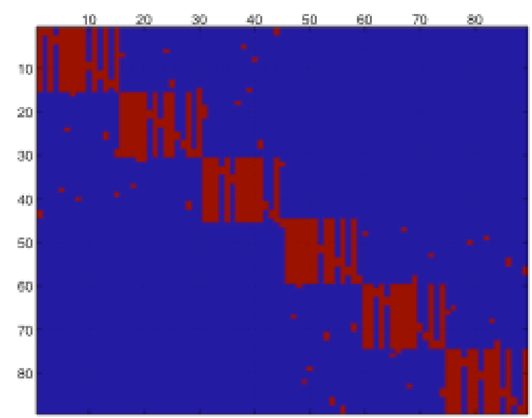
- To sort row i
  - Find Distance(i,j) for all j>i
  - Choose "closest" row, call it k, and interchange rows (and columns) i+1 and k
- Repeat for i+1, etc.

Results

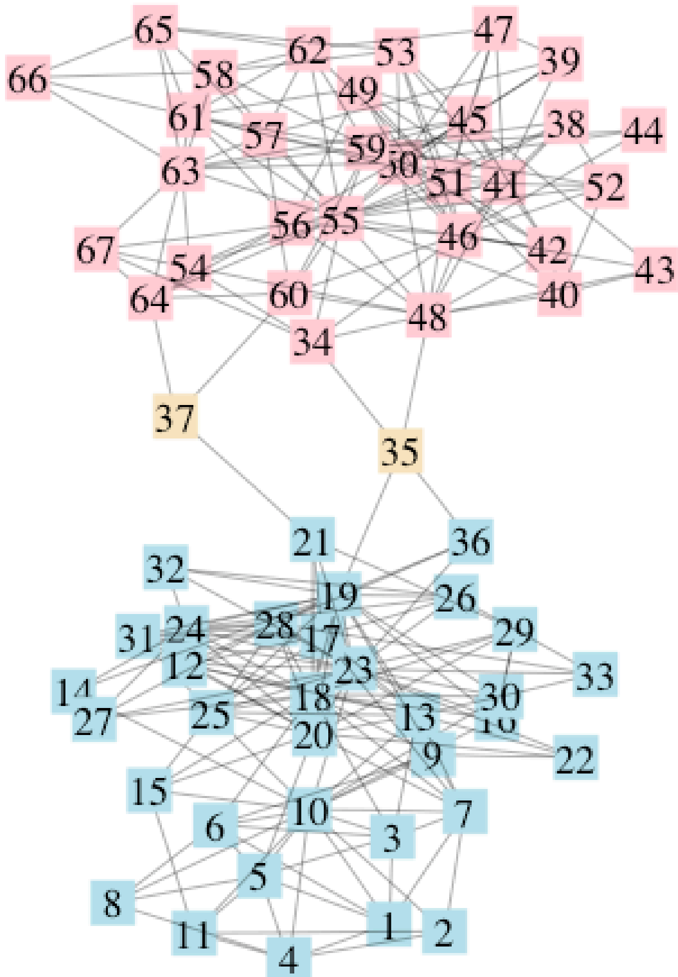


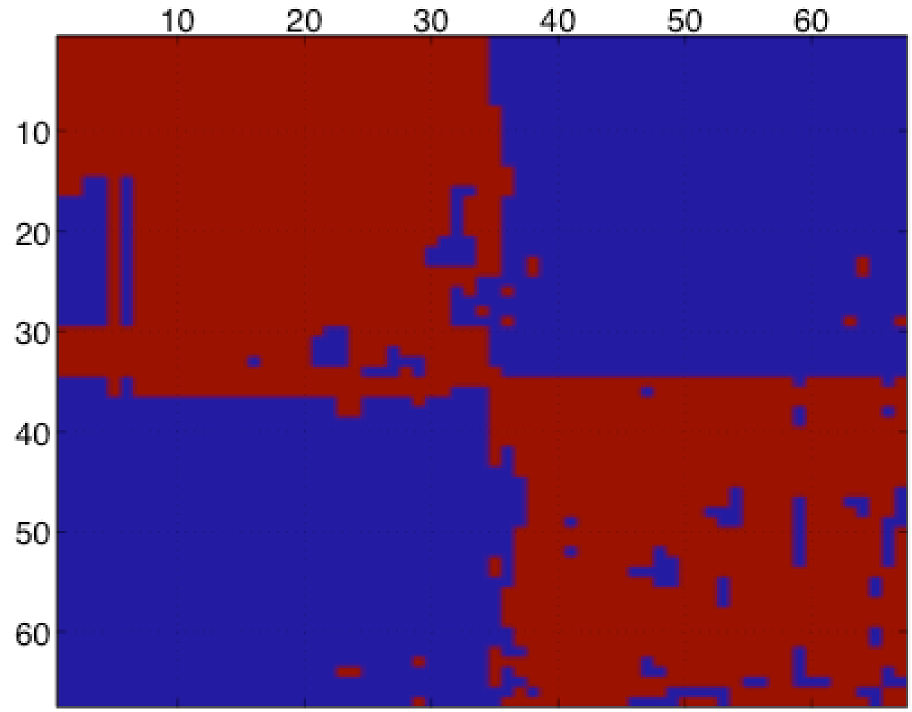


$$\alpha = 0.87$$

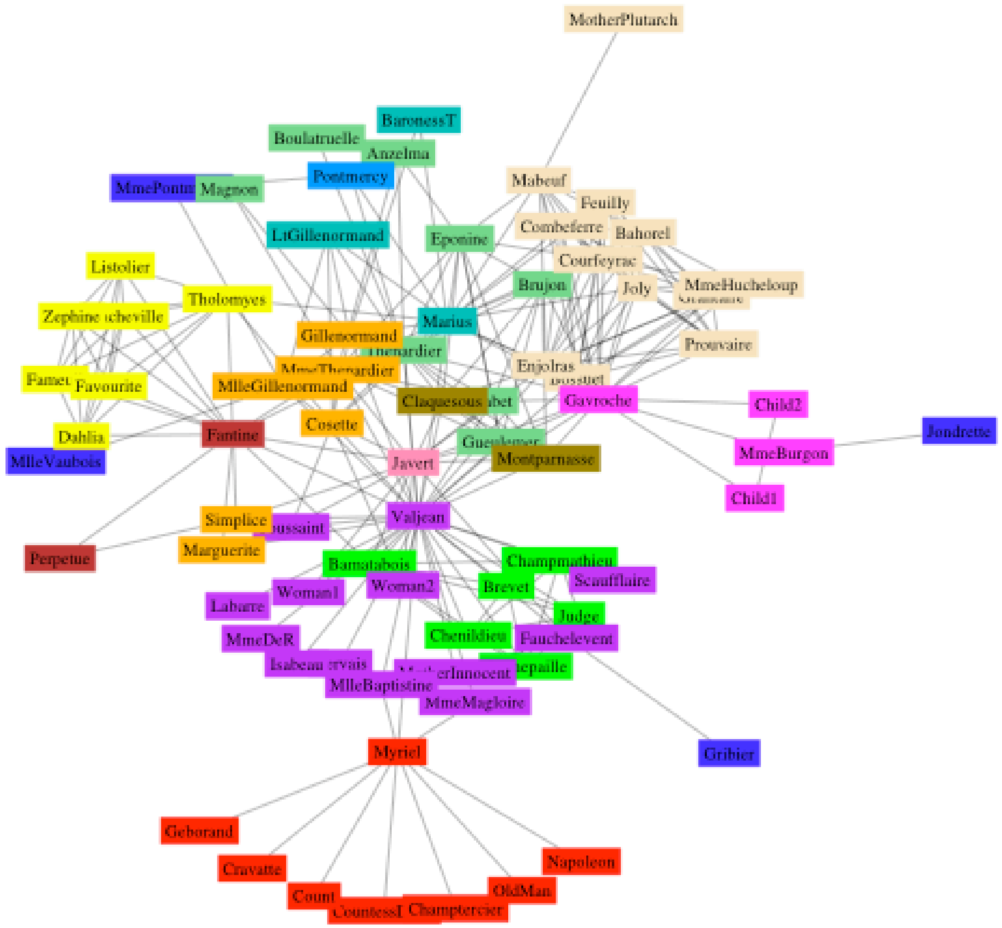


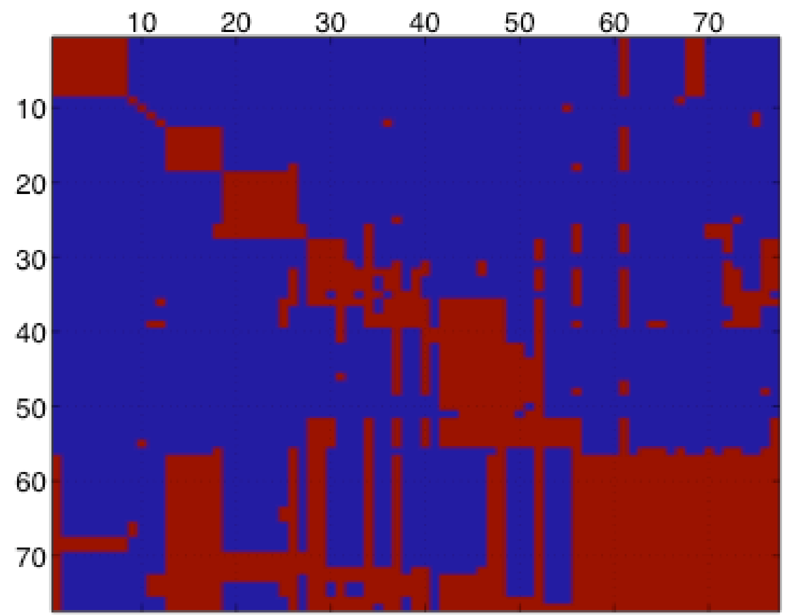
$$\alpha = 3$$





$$\alpha = 1.2$$

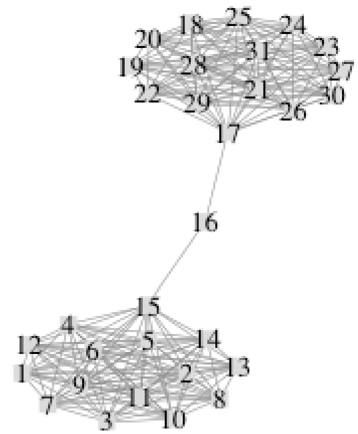

MotherPlutarch
BaronessT
Boulatruelle
Anzelma
MmePontm
Magnon
Pontmercy
Mabeuf
Feuilly
Combeferre
Bahorel
LtGillenormand
Eponine
Courfeyrac
Listolier
Brujon
Joly
MmeHucheloup
Tholomyes
Zephine
Marius
Gillenormand
Prouvaire
Enjolras
Famet
Favourite
MlleGillenormand
Claquesous
Gavroche
Child2
Jondrette
Cosette
Dahlia
Fantine
MlleVaubois
Javert
Montparnasse
MmeBurgon
Child1
Simplice
Valjean
Perpetue
Marguerite
Champmathieu
Bamatabois
Woman1
Woman2
Brevet
Scaufflaire
Labarre
Judge
MmeDeR
Chenildieu
Fauchelevent
MlleBaptistine
MmeMagloire
Myriel
Gribier
Geborand
Napoleon
Cravatte
OldMan
Count
Champtercier

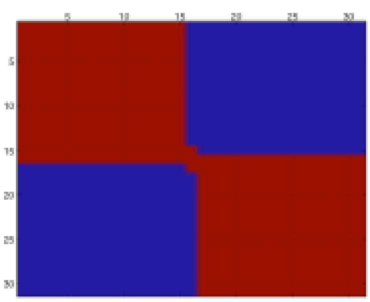

10
20
30
40
50
60
70
10
20
30
40
50
60
70

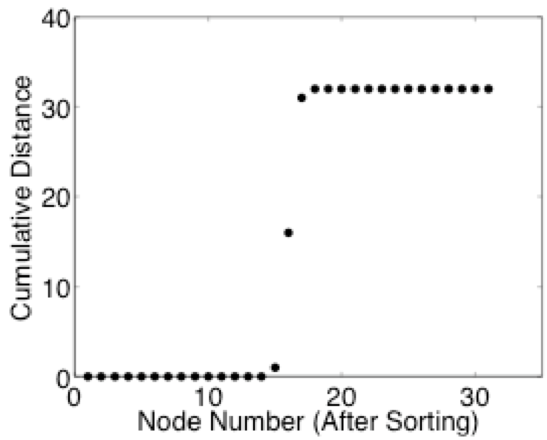
$$\alpha = 6.9$$

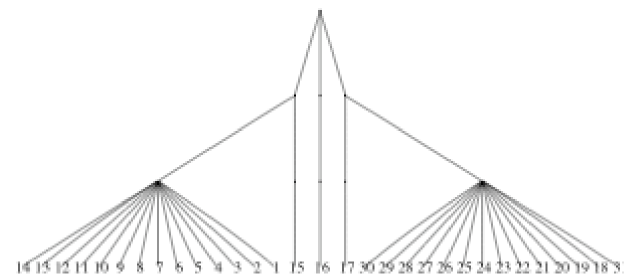
Towards a Dendrogram Visualization of Results

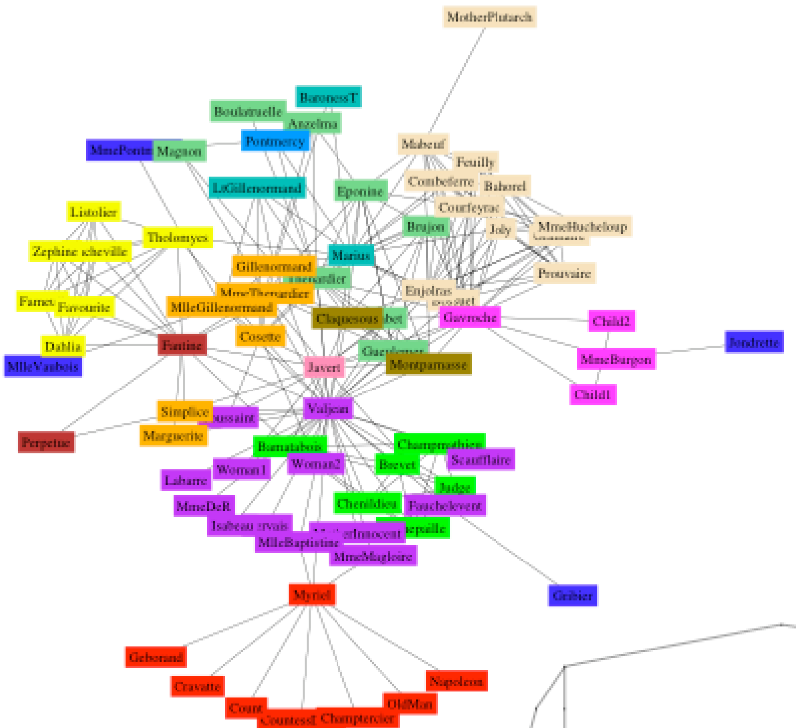
- Choose shortest difference between adjacent points
  - cluster all nodes closer than that distance
  - choose next shortest difference



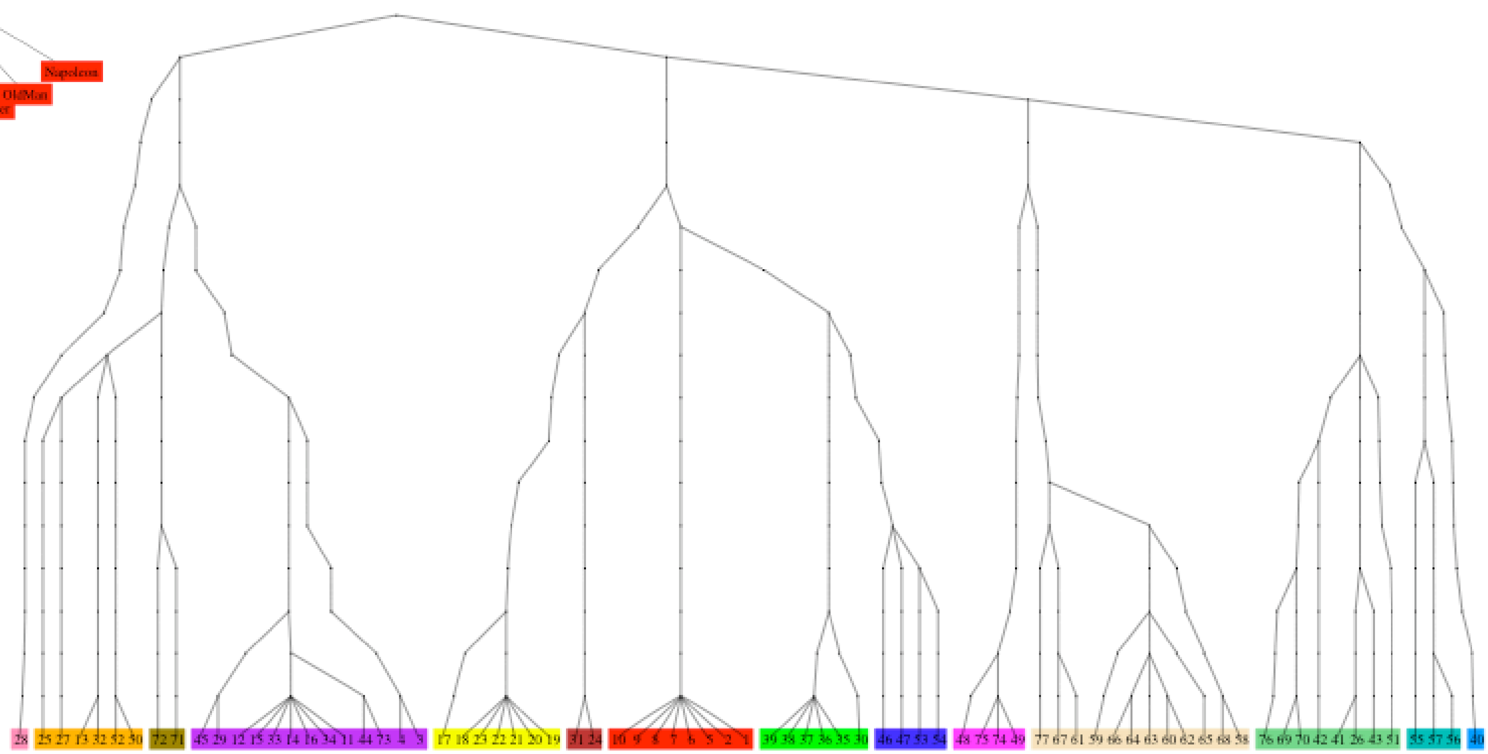








Example Dendrogram



Cycles

- **Cycles are important.**

- **Cycles are the backbone of a network:**

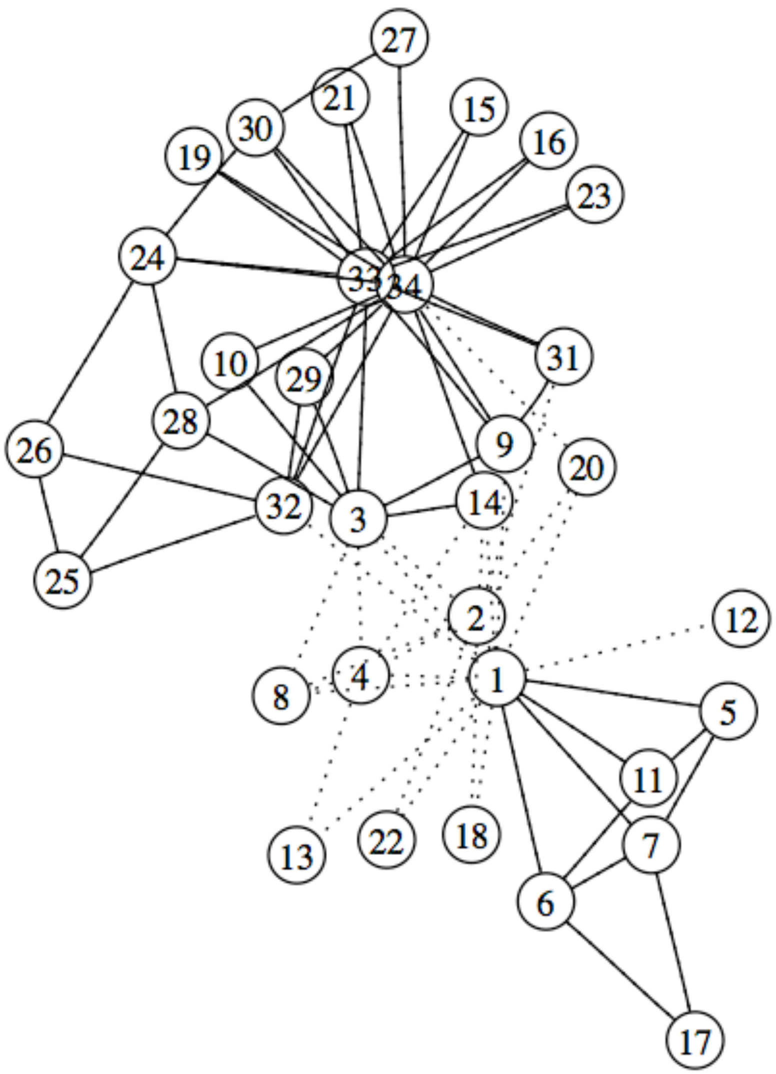
- **Cycles can also be used to find communities.**

Mantras for pics which follow:

- *Vertices more closely connected to each other are more likely to be connected by short cycles.*

- *Density of cycles of different lengths can be used as an indicator of connectivity between subsets of vertices.*

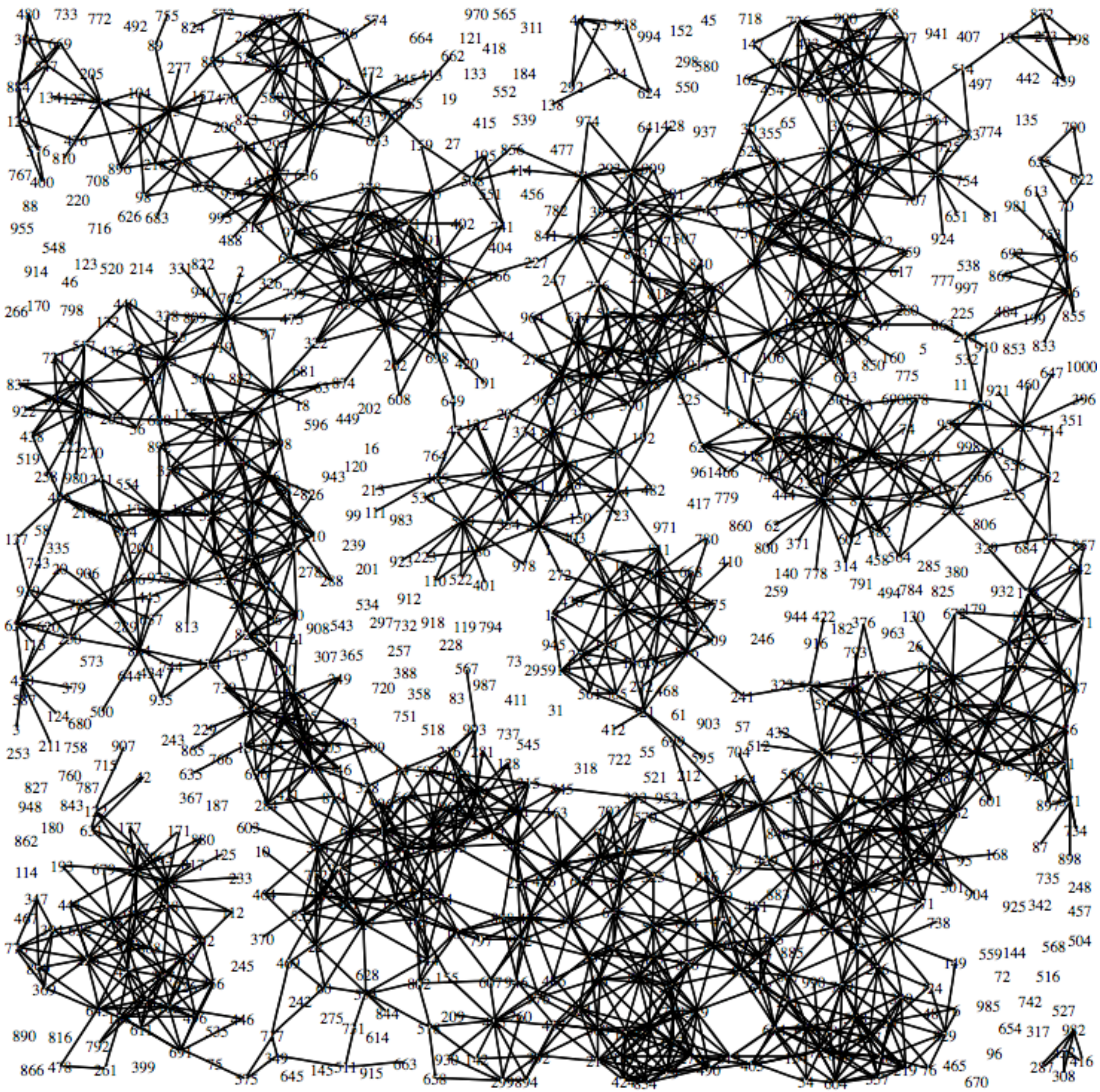
Partition Zachary's Karate Club by Short Cycles



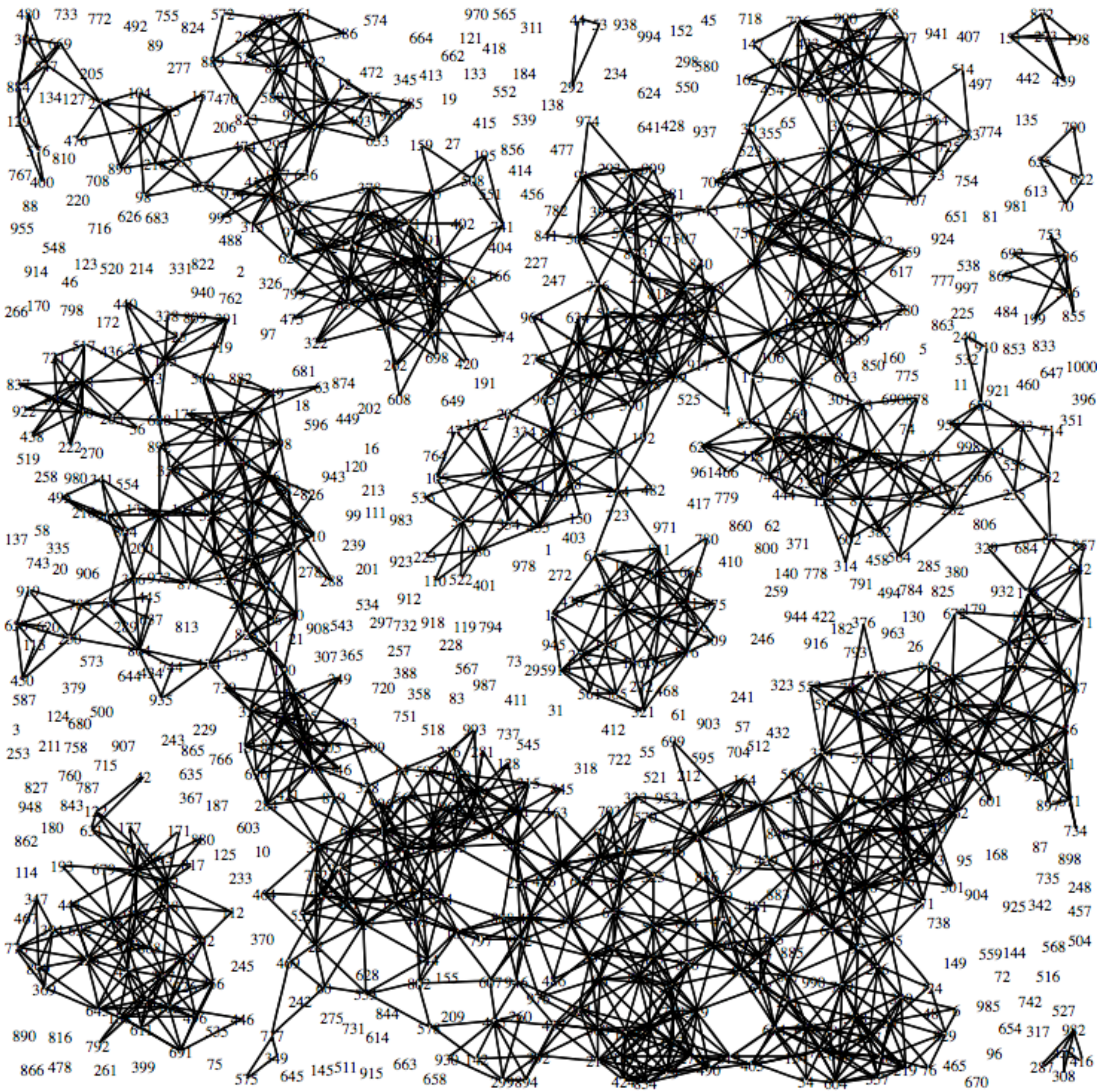
Components with 3,4,or 5 - cycles

Starting at 17, or 24

Preview: *On Loop Communities*.



A Particular Random Graph



Components Minus those w/o 3-cycles.

Statistics of Cycles:
How Loopy is your Network?

In order to characterize Networks we need to study their properties.
Finding global characteristics and properties of Networks.

•Degree Distribution: Probability to find a node with degree k.

Delta Distribution: $P(k)=\delta_{k,k'}$ (lattices or Regular Graph).
Power-law Distribution: $P(k)\sim k^{-\lambda}$ (Barabasi – Albert).

•Average Clustering Coefficient.

•Diameter.

•*Complete Distribution of Cycles:*

- Important for propagation along the network.
- Understanding the structure and topology of the network.
- Characterization of networks.

H. D. Rozenfeld, J. E. Kirk, E. M. Bollt, and D. ben-Avraham "Statistics of Cycles: How Loopy is your Network?," cond-mat/0403536, J. Phys. A: Math. Gen. 38 (2005) 4589-4595.

Networks and Cycles

h-cycle: closed path through h connected links that is self-avoiding.

N_h: total number of distinct h-cycles in the network.

h_*: size of typical cycle in the network.

Finding cycles in Networks

↓

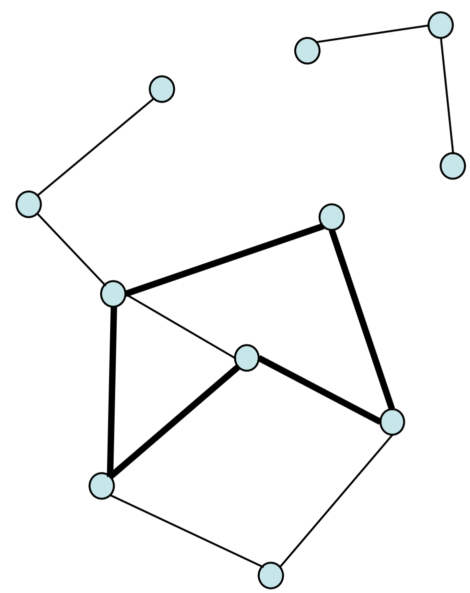
Depth-First-Search Algorithm

↓

Full distribution of
cycles of all sizes

↓

Extremely expensive:
N_h *grows exponentially with the cycle size!*



Networks and Cycles

This is the first time the complete distribution of cycles and the typical cycle size h_* are studied. As will be shown this quantity describes the distribution of cycles in the thermodynamic limit.

This topic is a very active field of research.
Some recent work on cycles:

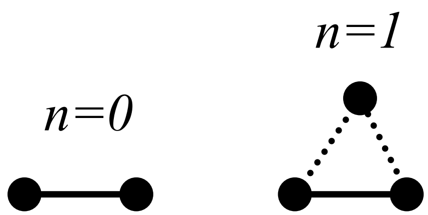
1. G. Bianconi, G. Caldarelli, A. Capocci. arXiv:cond-mat/0408349.
2. S. N. Dorogovtsev, J.F.F Mendes. arXiv:cond-mat/0404593.
3. E. Bollt, D. ben-Avraham. arXiv:cond-mat/0409465.
4. E. Marinari, R. Monasso. arXiv:cond-mat/0407253.
5. D. Sergi. arXiv:cond-mat/0412472.
6. A. Vazquez, J.G. Oliveira, A-L Barabasi. arXiv:cond-mat/0501399.
7. E. Ben-Naim, P.L. krapivsky. arXiv:cond-mat/0408620.
8. G. Bianconi, M. Marsili. arXiv:cond-mat/0502552.
9. S. N. Dorogovtsev, J.F.F. Mendes, J.G. Oliveira. arXiv:cond-mat/0411526.
10. H-J. Kim, J.M. Kim, S. N. Dorogovtsev, Mendes, J.F.F. arXiv:physics/0503168.
11. P.G. Lind, M.C. Gonzalez, H.J. Herrmann. arXiv:cond-mat/0504241.

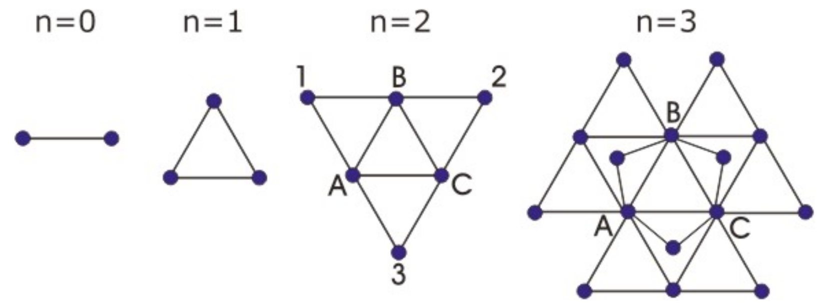
.

.

.

Cycles on a Deterministic Scale-Free Network

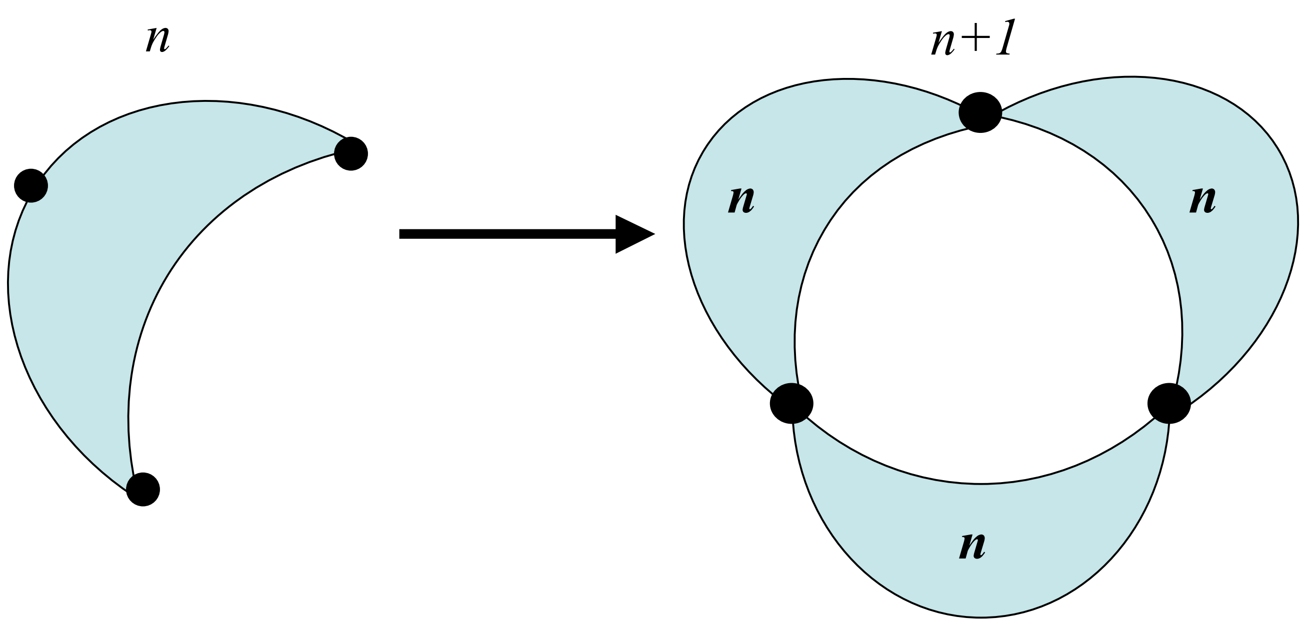




…

S.N. Dorogovtsev, A.V. Goltsev, and J.F.F. Mendes, Phys. Rev. E 65, 066122 (2002).

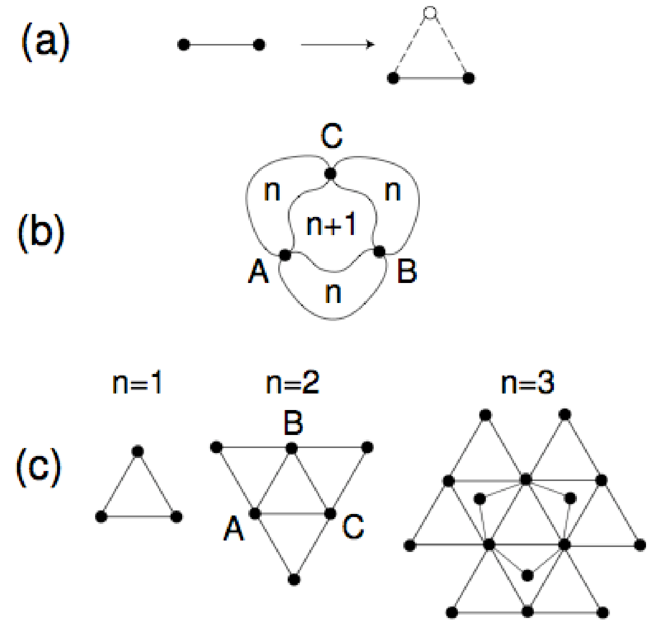
$$P(k) \sim k^{-\lambda}, \qquad \lambda = 1 + \frac{\ln 2}{\ln 3}$$



And through orbits of the Kronecker Algebra

$$A_1 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix},$$

$$A_{n+1} = R_{n+1} \cdot [M \otimes (P_n \cdot A_n \cdot P_n^t)] \cdot R_{n+1}^t.$$



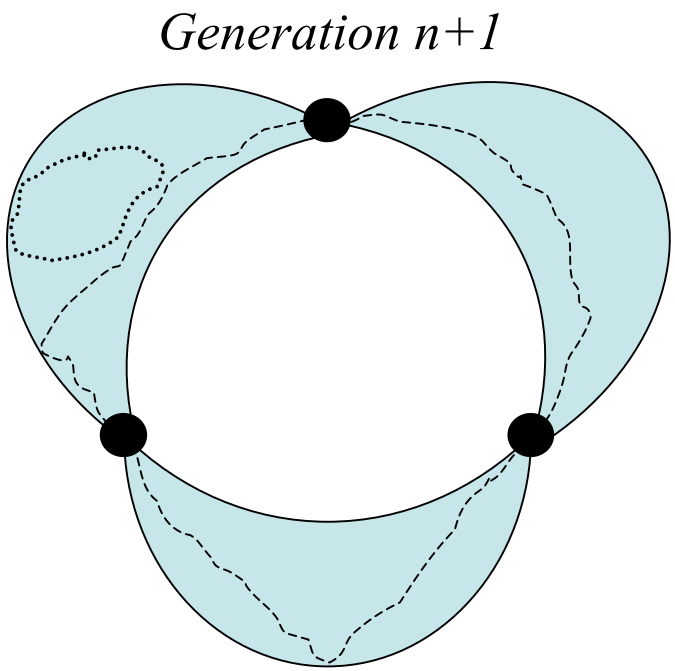
Copy instructions: $M = I_3,$

Assembly instructions

$$R_n = \begin{pmatrix} I_n(1,:) + I_n(n,:) \\ I_n(2:(n/3-1),:) \\ I_n(n/3,:) + I_n((n/3+1),:) \\ I_n((n/3+2):(2n/3-1),:) \\ I_n(2n/3,:) + I_n((2n/3+1),:) \\ I_n((2n/3+2):(n-1),:) \end{pmatrix}.$$

E.M. Bollt, D. ben-Avraham, “What is Special about Diffusion on Scale-Free Nets?” New J. Phys. 7 (2005) 26.

Cycles on a Deterministic Scale-Free Network

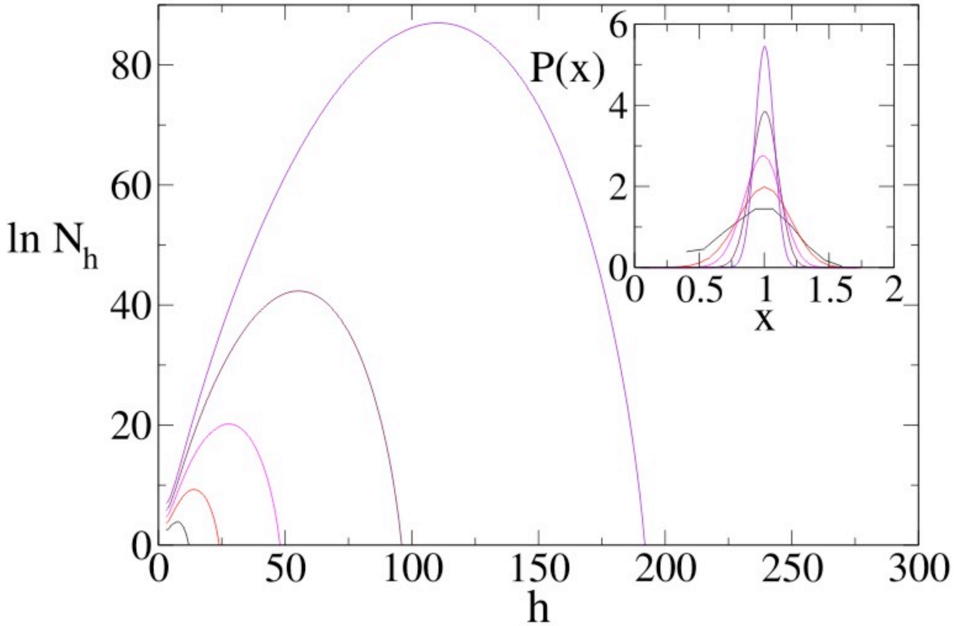


- $N_h(n)$: number of cycles of size h in generation n.
- $L_h(n)$: number of paths of size h between two hubs in generation n.

$$N_h(n+1) = 3N_h(n) + \sum_{\substack{h_1,h_2,h_3 \\ h_1+h_2+h_3=h}} L_{h_1}(n)L_{h_2}(n)L_{h_3}(n)$$

$$L_h(n+1) = L_h(n) + \sum_{\substack{h_1,h_2 \\ h_1+h_2=h}} L_{h_1}(n)L_{h_2}(n)$$

Cycles on a Deterministic Scale-Free Network



- $f(x)$ is well approximated by a parabola about $x=1$ or $h=h_*$.
- The distribution of cycles is nearly a Gaussian of width $1/\sqrt{h_*}$
- In the thermodynamic limit $h_* \to \infty$ or $N \to \infty$ it converges to a delta function.

$$\ln N_h = h_* f\left(\frac{h}{h_*}\right), \qquad h_* \sim 2^n$$

$$\downarrow$$

$$N(n) = \frac{3^n + 3}{2} \longrightarrow \boxed{h_* \sim N^{\alpha} \quad , \alpha = \frac{\ln 2}{\ln 3}}$$

A typical cycle length!

Why the phrase "ergodic" in the title?

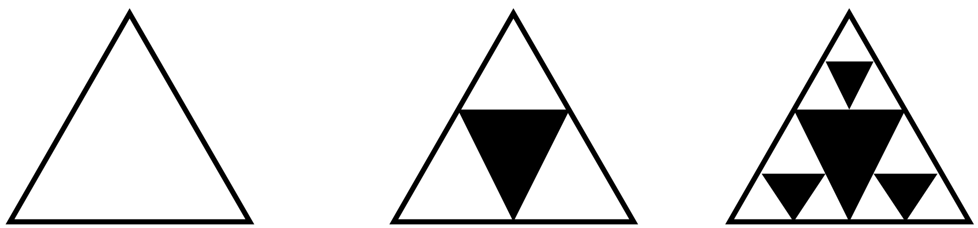
Just suggestive. But in the presence of a typical cycle length, think of a typical averaged quantity across the network, which "mostly" tours the network. Such cycles do not have a sense of location. You should be able to pick up a cycle just about anywhere, and get the same idea of the network as a whole.

Think of averages of some statistic on the network, like clustering coefficient for example. Just a few large, "typical" Cycles will tend to carry the same statistics, and fast converging, as the network as a whole.

Cycles on a Lattice, Fractal, Complete Graph.

•It was found by Jensen and Guttman that the typical cycle length in a lattice (regular graph) with $\sqrt{N} \times \sqrt{N}$ sites follows $h_* \sim N$ or α=1.

•For the Sierpinsky Gasket, a fractal lattice, all nodes have the same degree k=4 (except for three nodes). Again $h_* \sim N$ or α=1.



•For the Complete Graph K_N the distribution of cycles can N_h be obtained analytically

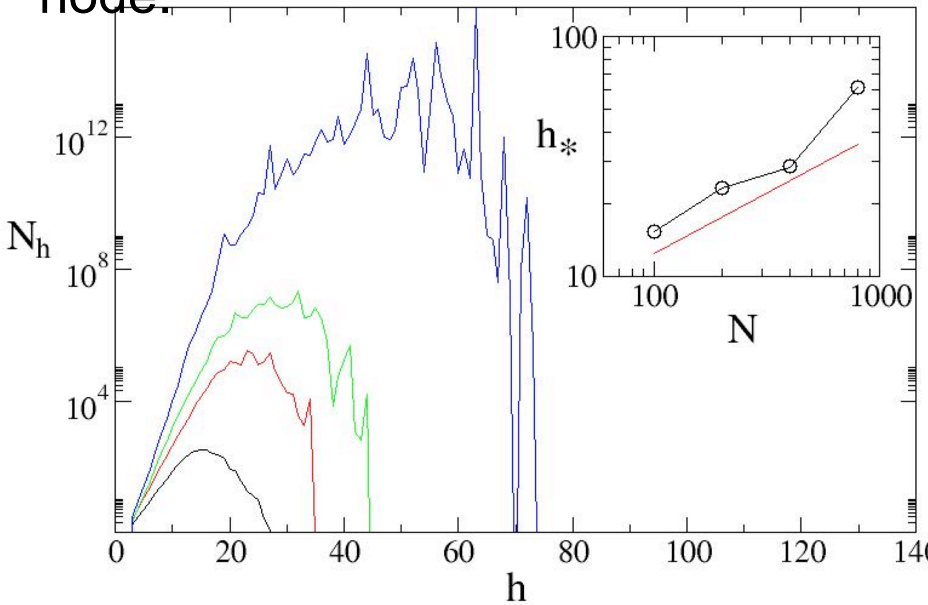
$$N_h = \frac{N!}{2h(N-h)!}$$

so $h_* \sim N$ or α=1.

Cycles on Random Scale-Free Networks

Exact counting is not possible in random Networks.

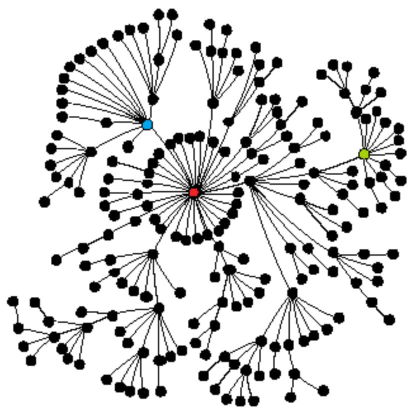
Sampling Algorithm: self-avoiding walk. The walk is terminated when it comes back to the initial node.



$P(k) \sim k^{-\lambda}$

$N = 100,200,400,800.$
$\lambda = 3.$

$$h_* \sim N^{\alpha} \qquad , \alpha = \frac{1}{2}$$

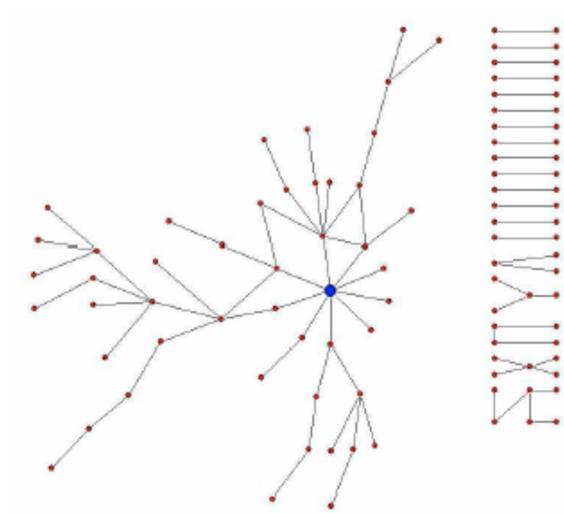


Cycles on Random Scale-Free Networks

We applied the sampling algorithm to Scale-Free Networks with degree exponent λ=3.

For Small Scale Free Networks $h_* \sim N^{\alpha} \quad , \alpha = \frac{1}{\lambda - 1}.$
For large Networks h_* increases.

Why?



For $N \approx 100$ most cycles are formed between the hub and nodes in the first shell. The likely cycle length is then proportional to the degree of the hub.

$$h_* \sim K \sim N^{\frac{1}{\lambda - 1}} \Rightarrow \alpha = \frac{1}{\lambda - 1}$$

As N grows larger, nodes in higher shells form part of interconnected cycles and alpha increases.

Conclusions and open questions

•Cycles of small length have been studied before.

•Our study indicates that the full distribution of cycles, of all possible lengths, displays additional useful properties:

- For large nets, the distribution resembles a delta function that peaks about a typical cycle size, $h \sim N^{\alpha}$.

- The exponent α serves as a single figure of merit that characterizes the "loopiness" of the net in question.
- For regular lattices and fractals, and for complete graphs and ER above criticality $\alpha=1$.

- For small random scale-free nets α = 1/(λ-1), but increases as the nets become larger.
- It remains an open question whether the loopy exponent saturates at α = 1 as $N \to \infty$.

- Ergodicity: the distribution of cycles that pass through a node is similar for most nodes of the net.

References

[1] R. Albert and A.-L. Barabasi, Rev. Mod. Phys. 74, 47 (2002).
[2] S. N. Dorogovtsev and J. F. F. Mendes, Adv. Phys. 51, 1079 (2002).
[3] S. Bornholdt and H. G. Schuster, Handbook of Graphs and Networks, (Wiley-VCH, Berlin, 2003).
[4] P. Erdos and A. Renyi, Publicationes Mathematicae 6, 290–297 (1959); Publications of the Mathematical Institute of the Hungarian Academy of Sciences 5, 17–61 (1960); Acta Mathematica Scientia Hungary 12, 261–267 (1961).
[5] D. J. Watts and S. H. Strogatz, Nature 393, 440 (1998);
[6] D. J. Watts, Small Worlds: The Dynamics of Networks between Order and Randomness (Princeton University Press, Princeton, New Jersey, 1999).
[7] A. L. Barabasi, R. Albert, Science 286, 509 (1999); R. Albert, H. Jeong, and A. L. Barabasi, Nature 401, 130, 1999.
[8] R. Cohen, K. Erez, D. ben-Avraham and S. Havlin, Phys. Rev. Lett. 85, 4626 (2000); Phys. Rev. Lett. 86, 3682 (2001).
[9] B. Bollob´as, Modern Graph Theory (Springer Verlag, 1998); Random Graphs (Academic Press, London,1985).
[10] S.N. Dorogovtsev, A.V. Goltsev, and J.F.F. Mendes,Phys. Rev. E 65, 066122 (2002).
[12] I. Jensen and A. Guttmann, private communication (conveyed to the authors by L. Glasser).
[13] D. ben-Avraham and S. Havlin, Phys. Rev. A 29, 2309(1984).
[14] P.G. de Gennes, Scaling Concepts in Polymer Physics,(Cornell University Press, Ithaca, 1979).
[15] S. N. Dorogovtsev and J. F. F. Mendes, Phys. Rev. E 63, 062101 (2001).
[16] S. Itzkovitz, R. Milo, N. Kashtan, G. Ziv, and U. Alon, Phys. Rev. E 68, 026127 (2003).
[17] G. Bianconi and A. Capocci, Phys. Rev. Lett. 90, 078701 (2003).

Networks that change dynamically in time

Proof of Communication and Synchronization in *Instantaneously* Disconnected But Fast Moving-Neighborhood Networks

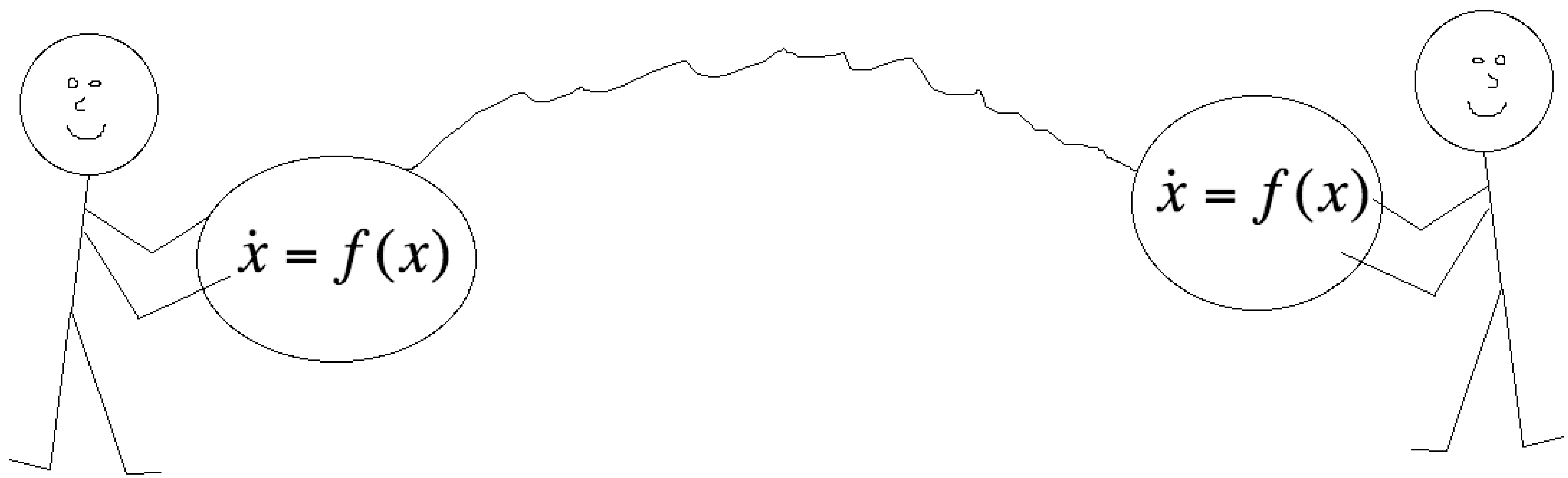
Erik Bollt, Clarkson University

Joe Skufca, USNA, U. MD, Clarkson U.

Dan Stilwell, Virginia Tech.

D. Gray Roberson, Virginia Tech.

Identical oscillators properly communicating can synchronize



Some Background: Identical Coupled Oscillators

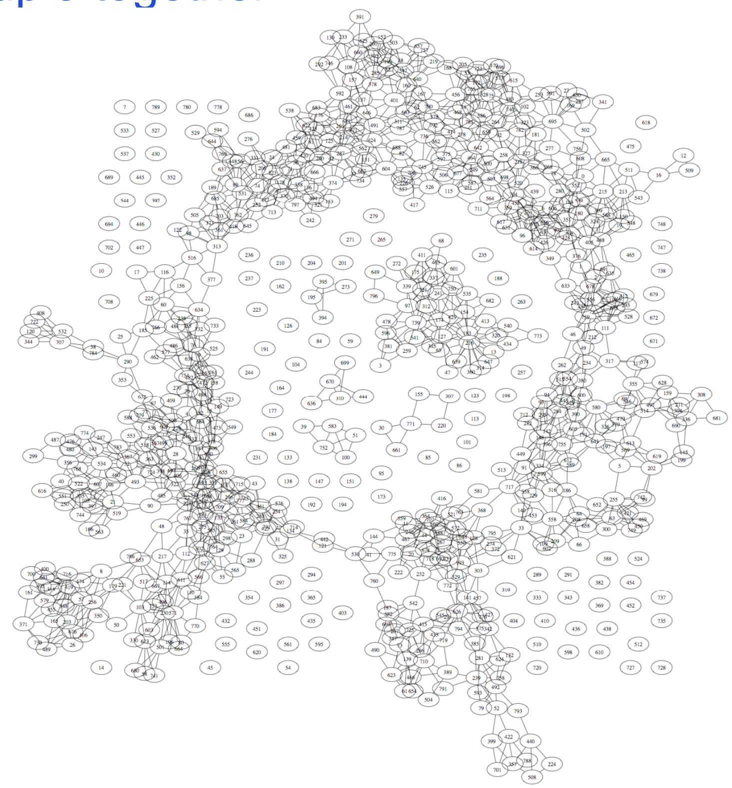
Barahona-Pecora-'02,
master stability function

Suppose a "network" of ***n***-identical oscillators which couple together according to a graph ***A***.

$$\dot{x}_i(t) = f(x_i(t)) + \sigma B \sum_{j=1}^{n} l_{i,j} x_j(t), i = 1, \ldots, n$$

$x_i(t) \in \mathbb{R}^n$ is the state of oscillator i, $B \in \mathbb{R}^{n \times n}$,

A *is* $n \times n$ *Adjacency matrix.*



Graph $A_{i,j} = 1$ *corresponding to an edge* (i,j), *else 0.*

$L = diag(d) - A$ *the graph-Laplacian*

Considering growth of variations away from a solution on an identical synchronization manifold,

$z_i(t) = x_i(t) - x^o(t)$, ***Variations from identical sync manifold***

$$\dot{z}(t) = (I_r \otimes F(t) + \sigma(I_n \otimes B)(L \otimes I_r))\, z(t) = (I_r \otimes F(t) + \sigma L \otimes B)\, z(t)$$

Synchronization of *Static Networks* of Identical Oscillators

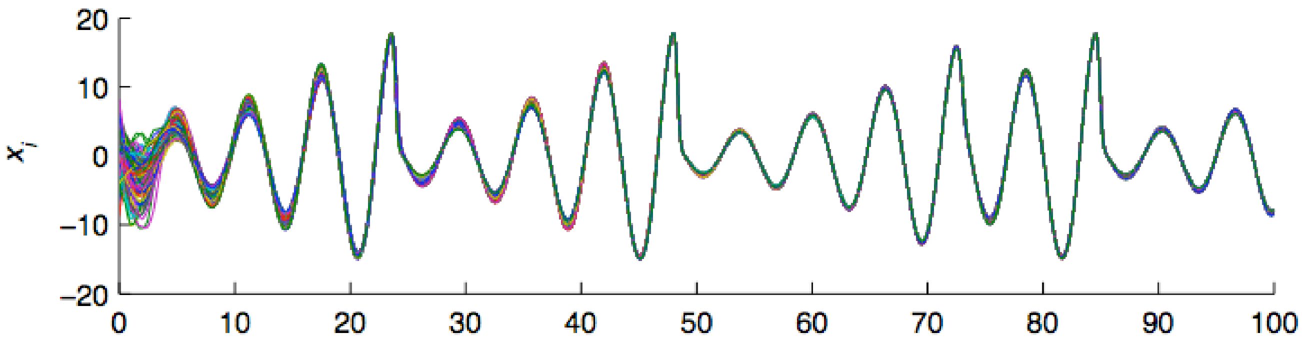
stability of the subsystems, $\dot{\xi}_i(t) = (F(t) + \sigma\lambda_i B)\xi_i(t), \qquad i = 1, \ldots, r$

$\lambda_1, \ldots, \lambda_r$ are the eigenvalues of L. the row sums of L are zero, the spectrum of L contains at least one zero eigenvalue, which we assign $\lambda_1 = 0$.

$\dot{\xi}_1(t) = F(t)\xi_1(t)$ evolves on the along the synchronization manifold,

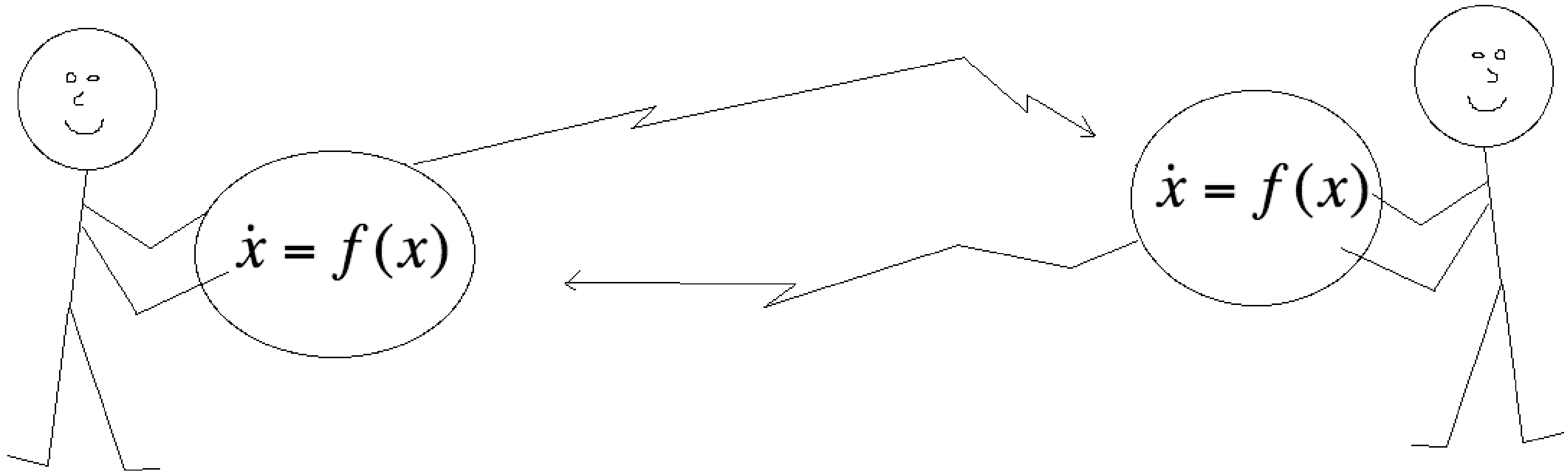
$i = 2, \ldots, r$ evolves transverse

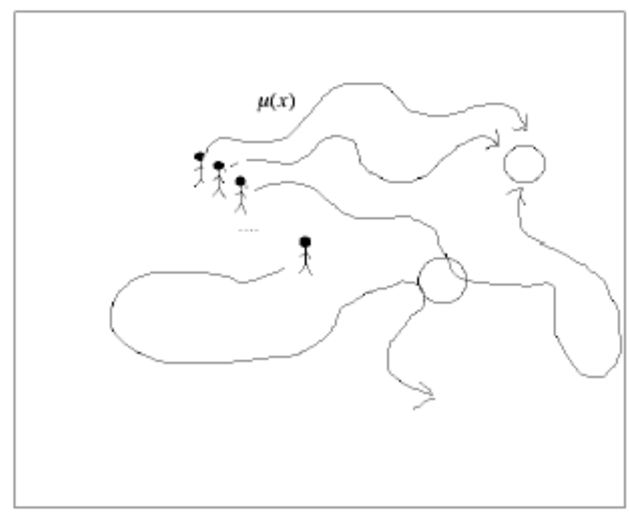
Since the oscillators are assumed identical, the (identity) synchronization manifold is invariant for all couplings, the question being its stability.



What about Time-Varying Networks?

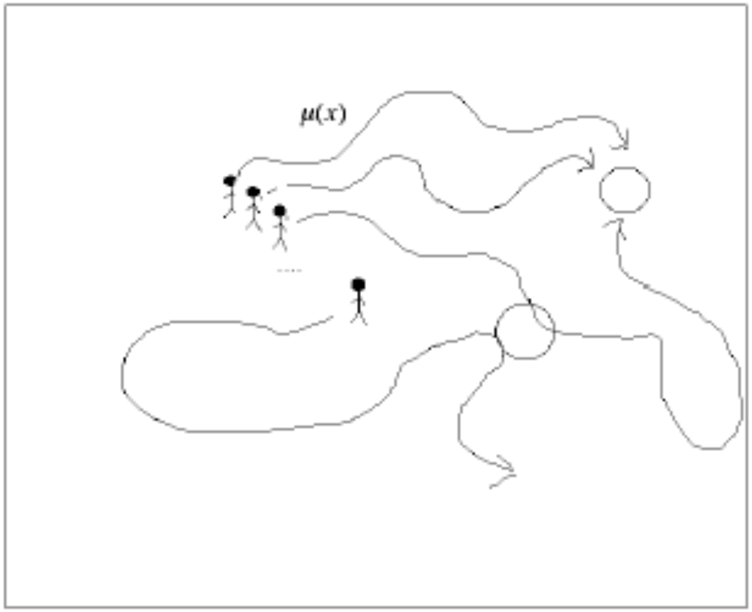
Can the Identical oscillators synchronize if they only communicate **only sometimes**?





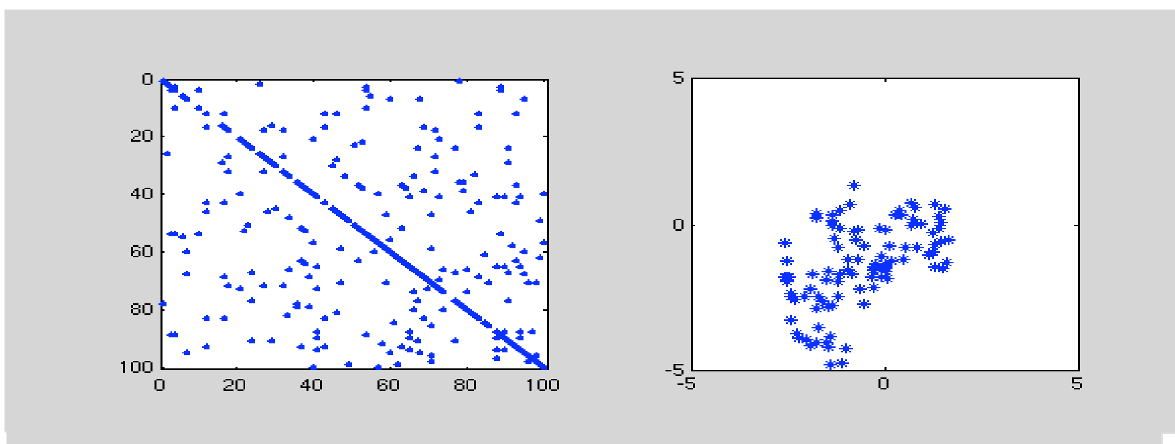
T. Stojanovski, L. Kocarev, U. Parlitz, R. Harris, "Sporadic driving of dynamical systems." Physical Review E 55: 4035, 1997.

Communication and Synchronization in Disconnected Networks with Dynamic Topology: *Moving Neighborhood Networks*



Q: Can information propagate (in time) across an Ad-hoc network?

Applications: Diseases, Control, Cooperative behavior of swarms, Synchronization, Communication



Instantaneous Adjacency Matrices

Ergodic Evolution of n moving agents

Getting ready to State Main theorem: Rigorous near the end.
First, the *Moving Neighborhood Network* Skufca, Bollt, '04

Suppose a "network" of ***n***-identical oscillators which couple together *occasionally* according to ***an ergodic process***.

$$\dot{x}_i(t) = f(x_i(t)) + \sigma B \sum_{j=1}^{n} l_{ij}(t) x_j(t), \quad i = 1, \ldots, n$$

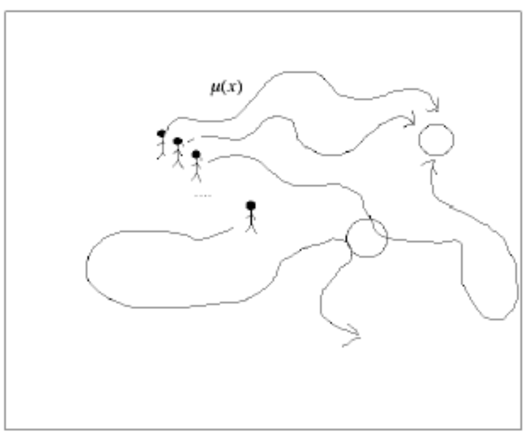
$x_i(t) \in \mathbb{R}^n$ is the state of oscillator i, $B \in \mathbb{R}^{n \times n}$,

$A(t)$ *is* $n \times n$ *Time varying adjacency matrix.*

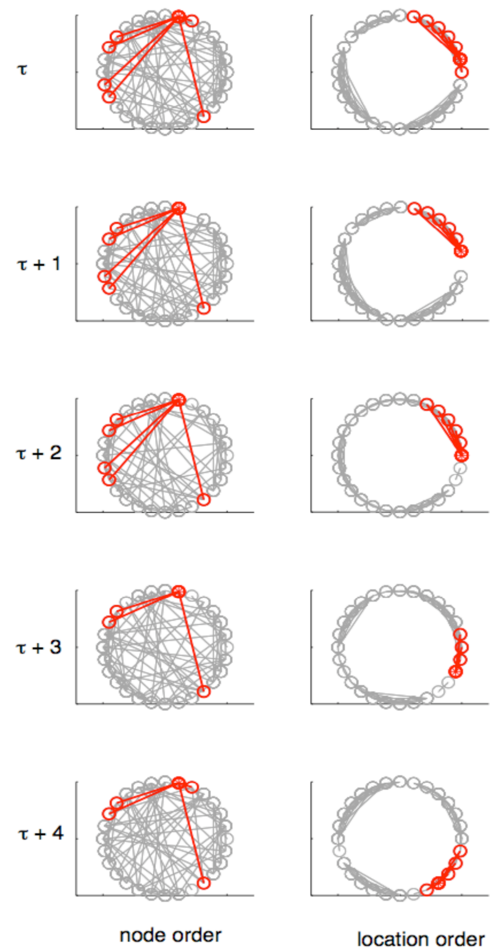
Graph corresponding to an edge (i,j) *at time* t *when*

$$|x_i(t) - x_j(t)| < \varepsilon$$

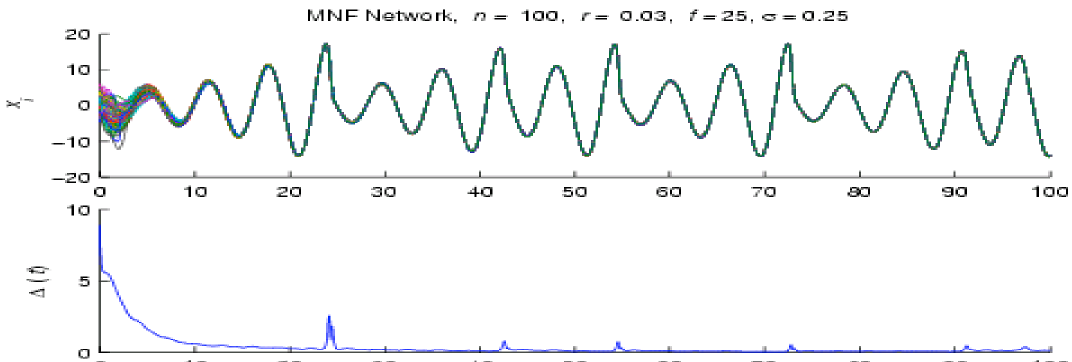
$L(t) = diag(d(t)) - A(t)$ *the **time varying** graph-Laplacian*



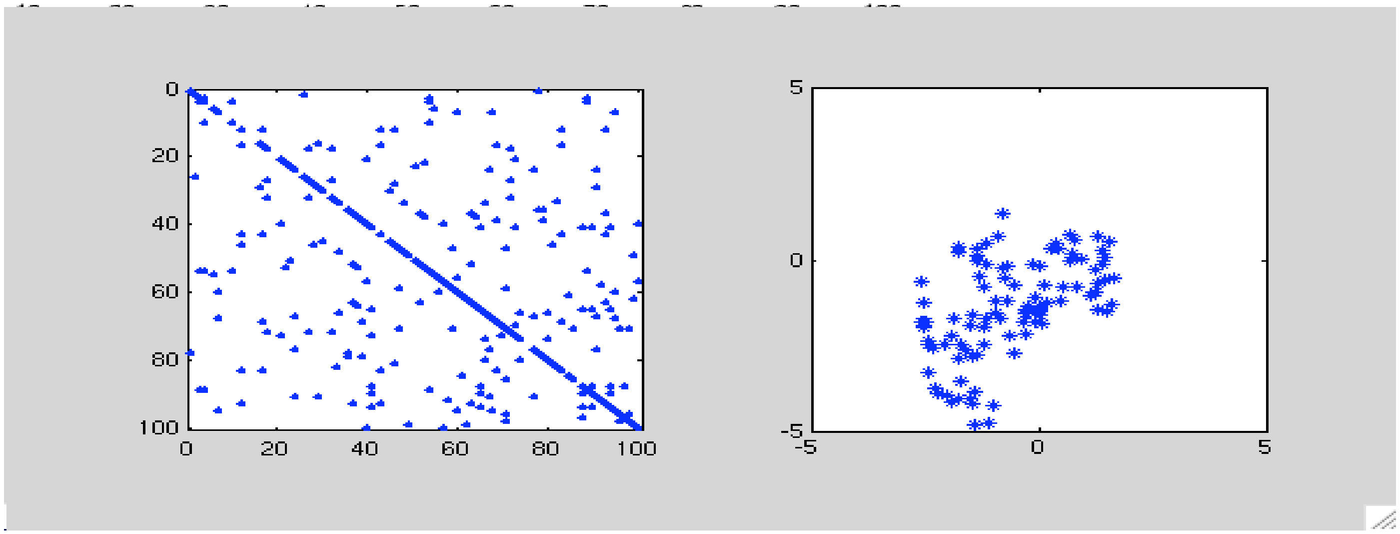
Reminds us of Barahona-Pecora-'02,
master stability function



Synchrony Probes Moving-Average Network



BUT!!! Network is
(a.) Never Instantaneously
Connected!



Instantaneous Adjacency Matrices　　**Ergodic Evolution of *n* moving agents**

Propagation of a message, biological disease, telecommunication, or any property, across a *dynamic network*, in which message relevance decays.

Two time-scales relevant for such networks:

1) Time-scale of the message

2) Time-scale of the validity of connection

Modify Master Stability Function

For dynamic networks

Barahona-Pecora-'02,
master stability function

Skufca, Bollt, '04

Moving Average Laplacian, for time varying networks - Weighting Memory

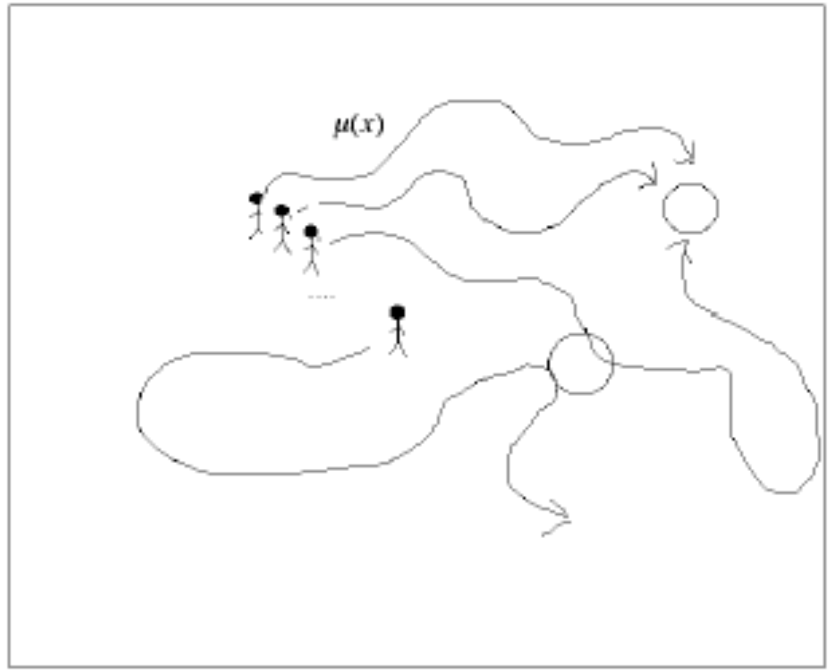
Instead of instaneous adjacency matrix, *A(t)*, use a moving averaged adjacency matrix,

$$\Lambda(t) = e^{\int_0^t qA(\tau)d\tau} z_0$$

which solves, $\dot{z} = qA(\tau)z, z(0) = z_0$ *and its Laplacian spectum*

A Time Varying Network

n-agents wandering ergodically.



Message propagation can persist in a network which is ***instantaneously never connected***

We find:

-Time-average ER, or scale-free degree distribution if latency.
-Message propagation in network which instantaneously never connected.
-Tools for such dynamic network topology: latency measuring "average moving (graph) Laplacian," generalization master-stability formalism, time-average degree distribution, ergodicity for time/space averages.

Stilwell, Bollt, Roberson, ‘05

A Related Model of A Time Varying Network

Theorem *Suppose a set of coupled oscillators with linearized dynamics*

$$\dot{z}_s(t) = \left(I_r \otimes F(t) + \sigma \bar{L} \otimes B\right) z_s(t)$$

achieves synchronization. Then there exists a positive scalar ε^* *such that the set of oscillators with linearized dynamics*

$$\dot{z}_a(t) = (I_r \otimes F(t) + \sigma L(t/\varepsilon) \otimes B)\, z_a(t)$$

and time-varying network connections $L(t)$ *achieves synchronization for all fixed* $0 < \varepsilon < \varepsilon^*$ *if* $L(t)$ *satisfies*

$$\frac{1}{T}\int_t^{t+T} L(\sigma)d\sigma = \bar{L}$$

and the column sums of $L(t)$ *are all zero for all* t*.*

If you “blink” connections fast enough, you can hope *on average* to transmit enough message to maintain synch.

Note: $\bar{L}$ *is generally not a graph Laplacian corresponding to any particular network.*

Proof is based on showing:

Lemma *Suppose the matrix-valued function $E(t)$ is such that*

$$\frac{1}{T}\int_t^{t+T} E(\sigma)d\sigma = \bar{E}$$

for all t and

$$\dot{x}(t) = (A(t) + \bar{E})x(t), \qquad x(t_o) = x_o, \qquad t \geq t_o$$

is uniformly exponentially stable. Then there exists $\varepsilon^* > 0$ such that for all fixed $\varepsilon \in (0, \varepsilon^*)$,

$$\dot{z}(t) = (A(t) + E(t/\varepsilon))z(t), \qquad z(t_o) = z_o, \qquad t \geq t_o$$

is uniformly exponentially stable.

By showing a shared Lyapunov function works sufficiently if switching is fast-enough.

Sufficient but not necessary: An example.

$$\dot{x}_i(t) = -y_i(t) - z_i(t) - \sigma \sum_{j=1}^{r} l_{ij}(t/\varepsilon) x_j(t)$$

$$\dot{y}_i(t) = x_i(t) + a y_i(t)$$

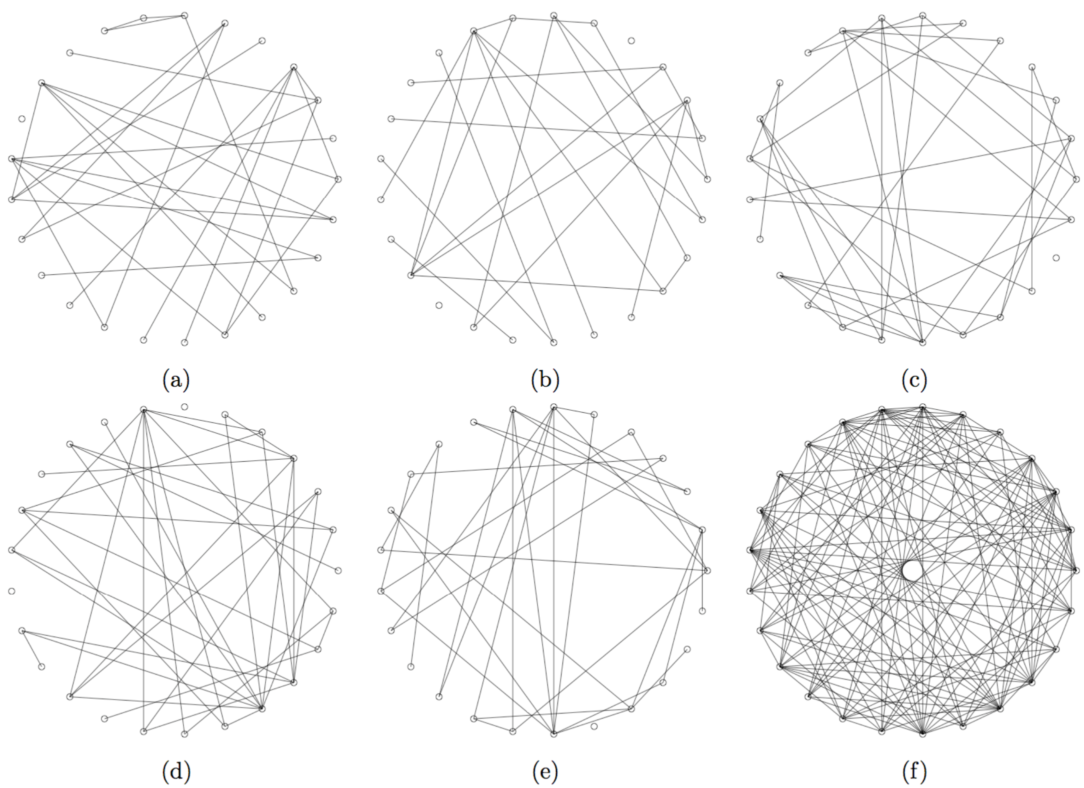
$$\dot{z}_i(t) = b + z_i(t)(x_i(t) - c)$$

$$i = 1, \ldots, r, \ a = 0.165, \ b = 0.2, \ c = 10, \ \text{and} \ \sigma = 0.3.$$

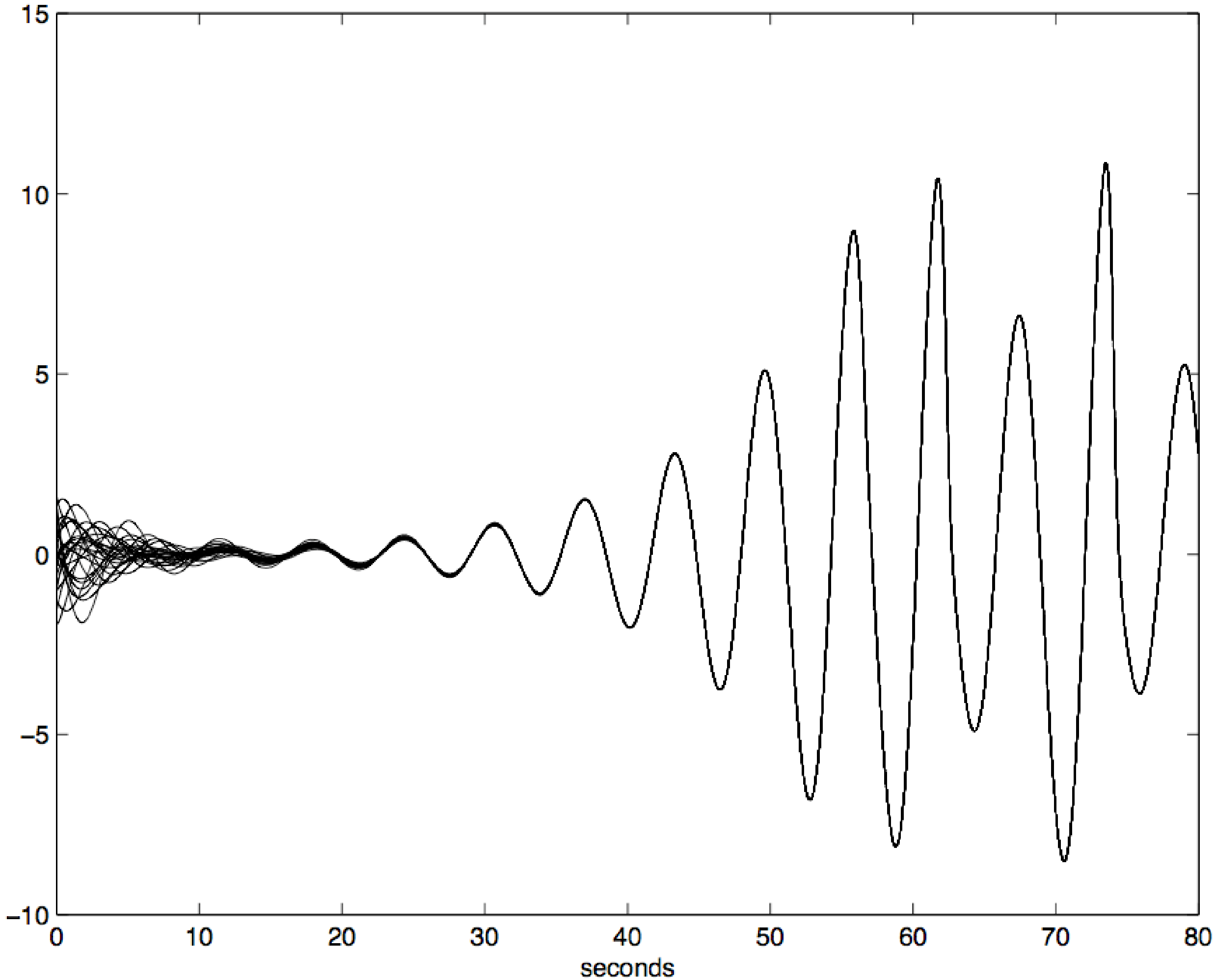
$$L(t) = \sum_{i=1}^{5} L_i \chi_{[(i-1)T/5,\, iT/5)}(t)$$

Time-averaged OF Laplacian

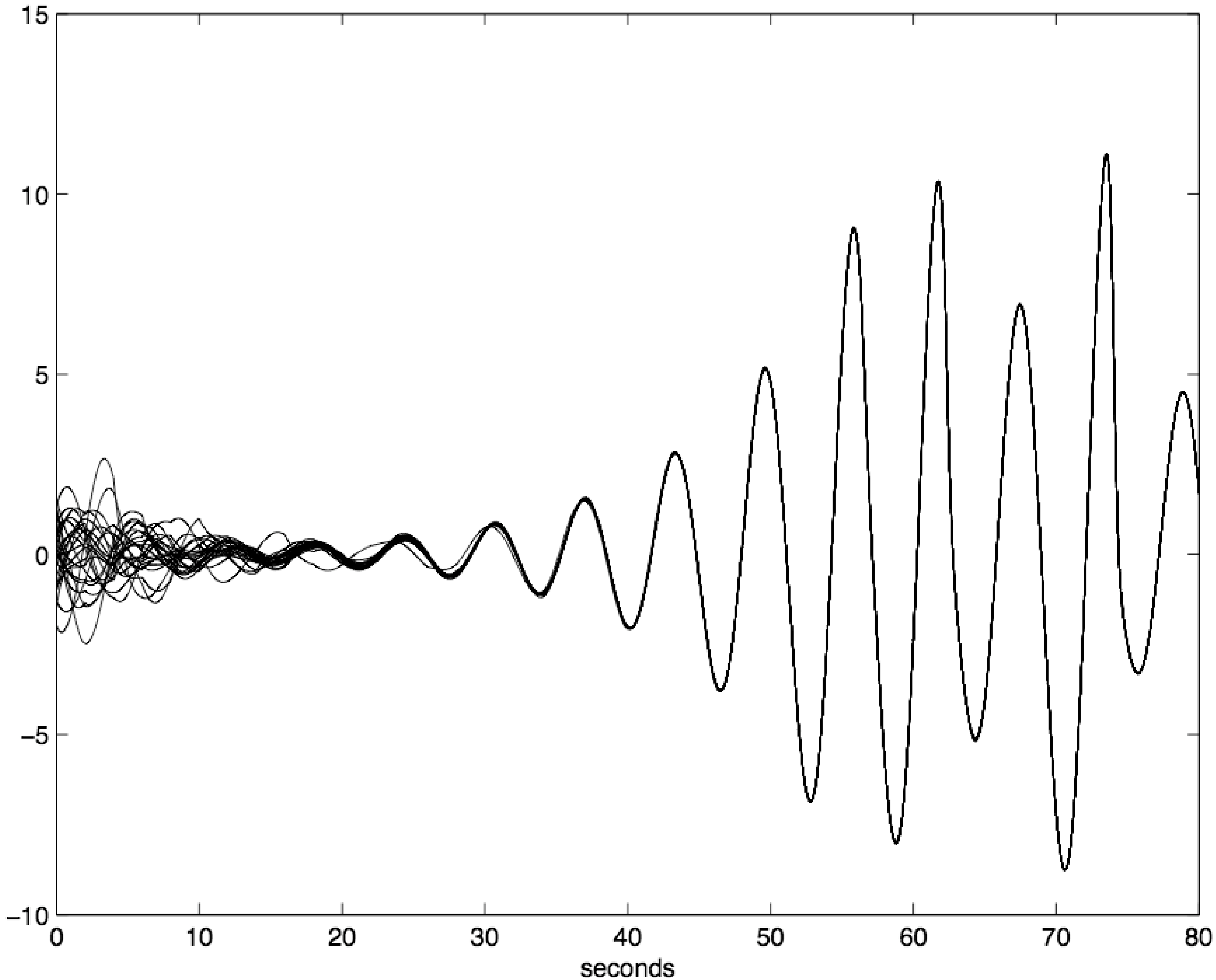
$$\bar{L} = \frac{1}{\varepsilon T} \int_0^{\varepsilon T} L(t/\varepsilon) dt$$
$$= \frac{1}{5} \sum_{i=1}^{5} L_i$$



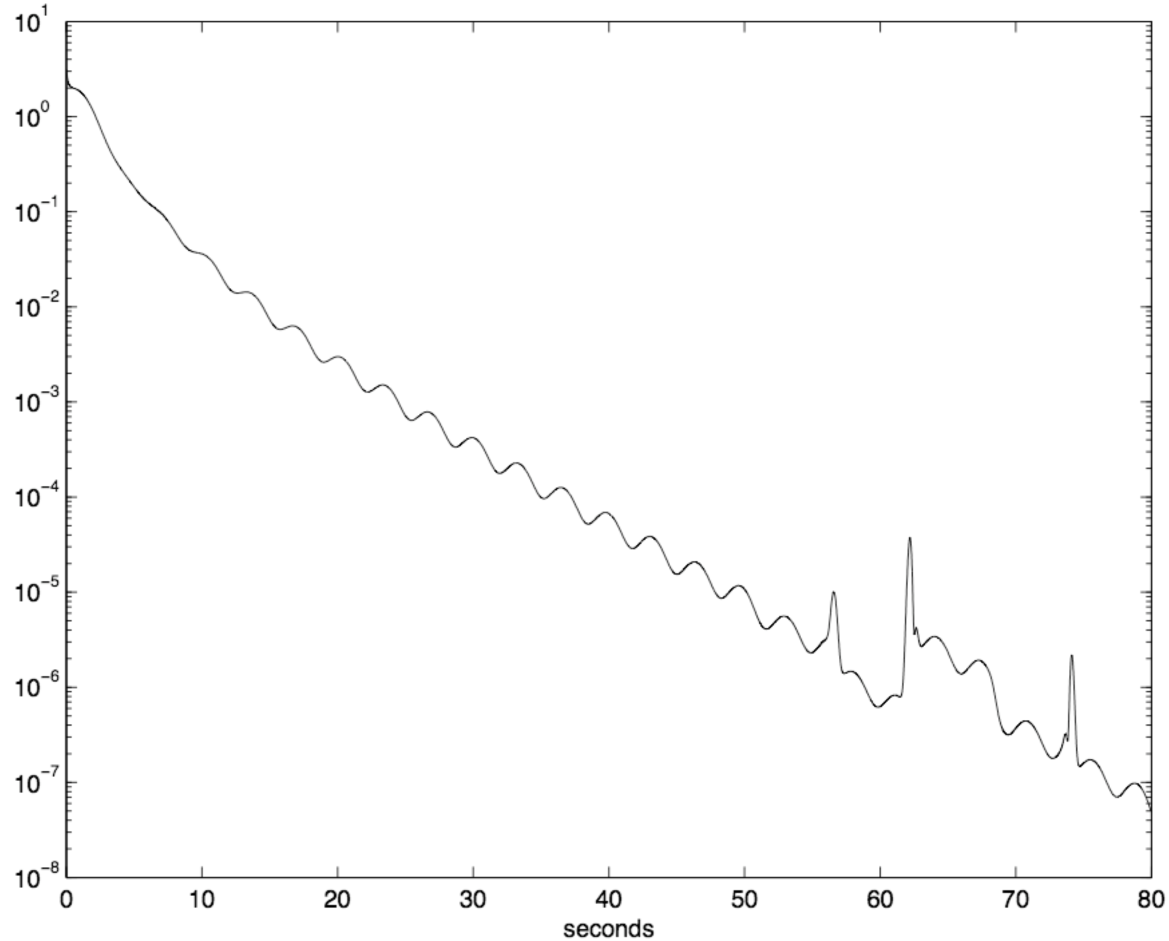
Notice that here, average over the 5 graphs is a connected Graph.



The x-coordinate for the set of coupled Rossler attractors using the average graph Laplacian.



The x-coordinate for the set of coupled Rossler attractors using the switched network where $\varepsilon = 1$.



Root-mean-squared error in for the set of coupled Rossler attractors using the average network $\bar{L}$.

$$\sum_{i=1}^{r}(x_i(t)-\mu_x(t))^2+(y_i(t)-\mu_y(t))^2+(z_i(t)-\mu_z(t))^2$$

Conclusions

-Fast switching. Sufficient but not necessary.

-Recent average must have some hope of communicating.

Future

-Stochastic and ergodic

-Other information bearing networked systems where connections are only sometimes good.

-Networked blinking control systems

Daniel J. Stilwell, Erik M. Bollt, D. Gray Roberson "Sufficient Conditions for Fast Switching Synchronization in Time Varying Network Topologies ," StilwellBolltGray05.pdf nlin.CD/0502055, Submitted to SIAM J. Dynamical Systems, (2005).

J.D. Skufca, E. Bollt, "Communication and Synchronization in Disconnected Networks with Dynamic Topology: Moving Neighborhood Networks," nlin.CD/0307010, Mathematical Biosciences and Engineering (MBE), 1 (2) 347 (2004).

More on Master Stability Function For static networks

Barahona-Pecora-'02,
master stability function

Graph Laplacian matrix *L* has *n*-eigenvalues

$$0 = \lambda_0 \leq \ldots \leq \lambda_{n-1} = \lambda_{\max}$$

Stability reduces by linear perturbation analysis to a constraint on eigenvalues of the Laplacian,

$$\sigma\lambda_i \in (\alpha_1, \alpha_2), \forall i = 1, \ldots, n-1$$

(α_1, α_2) **depends on specific oscillator**

For small, σ sync. unstable if, $\sigma\lambda_1 < \alpha_2$ As $\sigma \uparrow$ instability arises if $\sigma\lambda_{\max} > \alpha_2$

So, there is a synch coupling parameter σ if $\frac{\lambda_{\max}}{\lambda_1} < \frac{\alpha_2}{\alpha_1} := \beta$

Skufca, Bollt, '04

Moving Average Laplacian, for time varying networks - Weighting Memory

Instead of instaneous adjacency matrix, *A(t),* use a moving averaged adjacency matrix,

$$\Lambda(t) = e^{\int_0^t qA(\tau)d\tau} z_0$$

which solves, $\dot{z} = qA(\tau)z, z(0) = z_0$ *and its Laplacian spectum*