Carnegie Mellon

Large Graph Mining - Patterns, Explanations ~~and Cascade Analysis~~

Christos Faloutsos

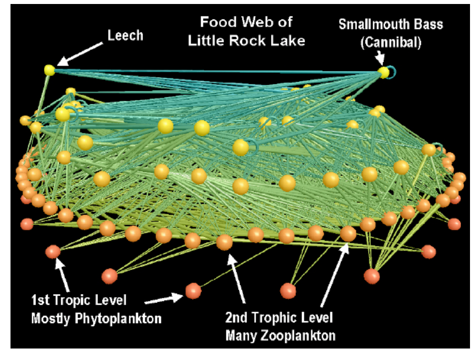
CMU

Thank you!

- Foster Provost
- Sinan Aral
- Arun Sundararajan

- Shirley Lau
- Sara Gorecki

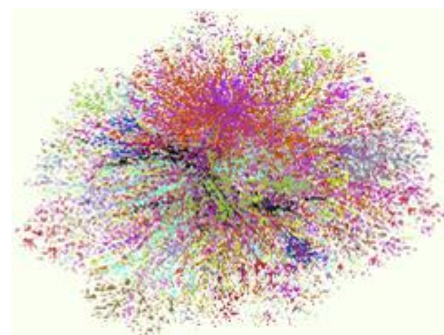
Graphs - why should we care?



Food Web
[Martinez '91]

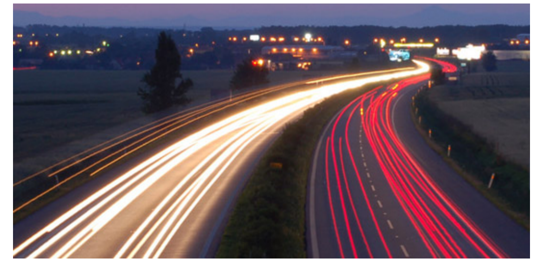
>$10B revenue

>0.5B users



Internet Map
[lumeta.com]

Roadmap

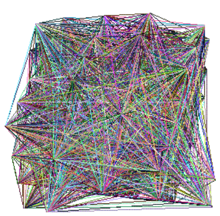


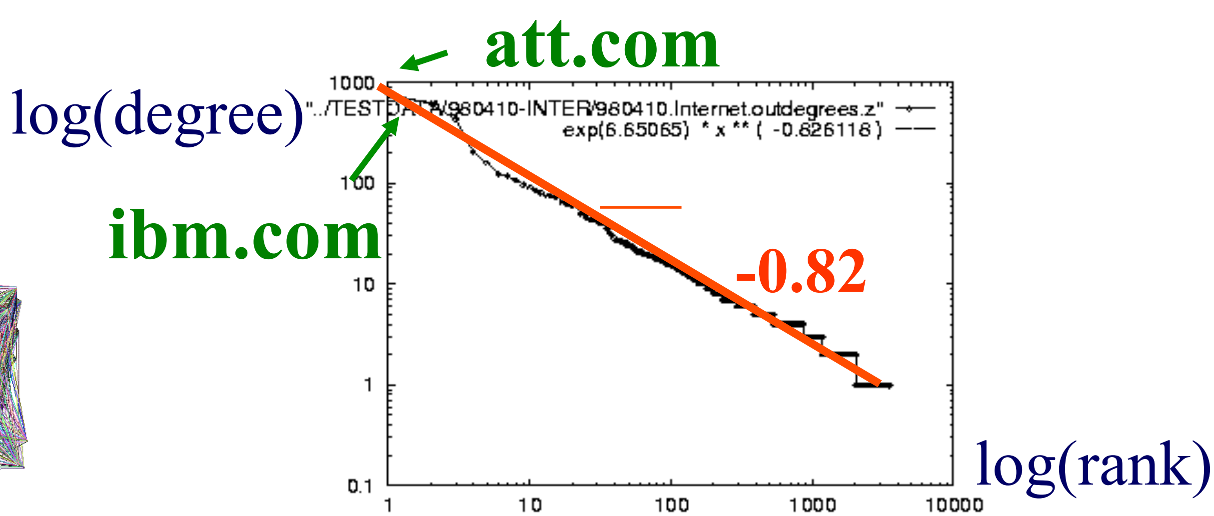
- Introduction – Motivation
- ➡ Part#1: Patterns in graphs
  - Some (power) laws
  - The 'no good cuts' shock
  - A possible explanation: fractals
- [Part#2: Cascade analysis]
- Conclusions

Solution# S.1

- Power law in the degree distribution [SIGCOMM99]

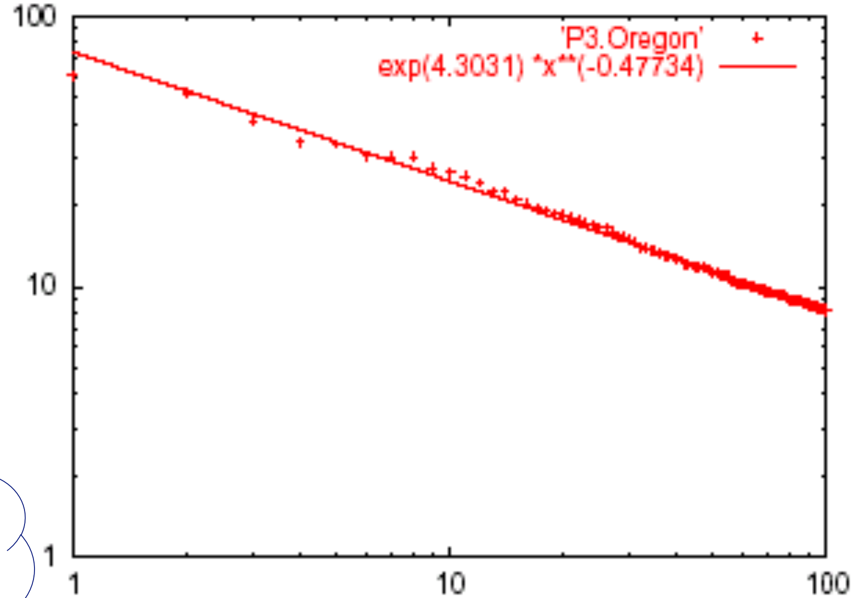
internet domains





Solution# S.2: Eigen Exponent *E*

Eigenvalue



Rank of decreasing eigenvalue

Exponent = slope

$E = -0.48$

May 2001

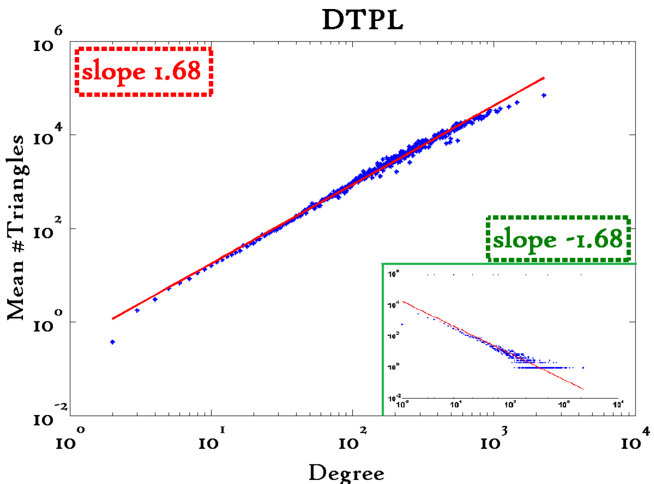
$\mathbf{A\,x} = \lambda\,\mathbf{x}$

- A2: power law in the eigenvalues of the adjacency matrix

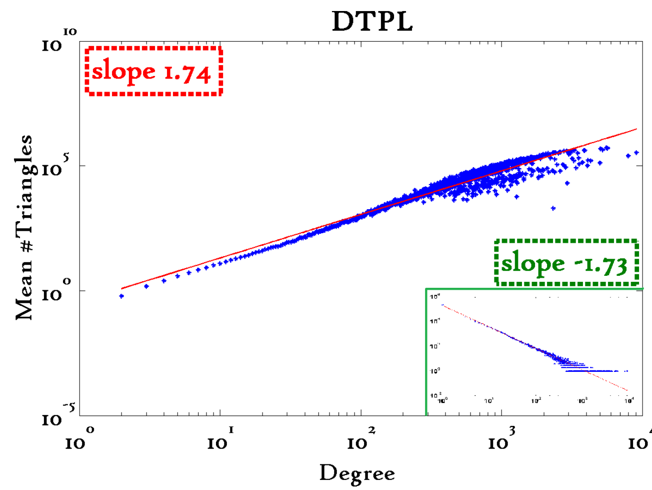
Carnegie Mellon

Triangle Law: #S.3
[Tsourakakis ICDM 2008]

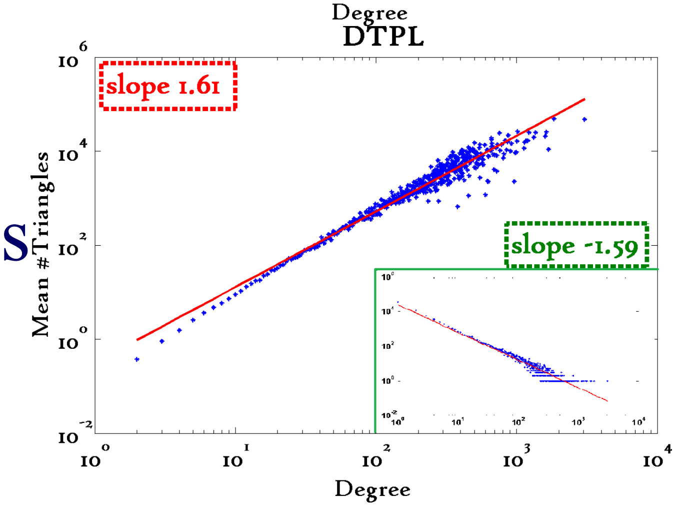
Reuters

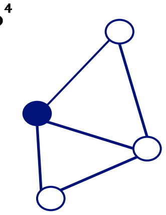


SN



Epinions





X-axis: degree

Y-axis: mean # triangles

n friends -> ~$n^{1.6}$ triangles

Carnegie Mellon

MORE Graph Patterns

	Unweighted	Weighted
Static	**L01.** Power-law degree distribution [Faloutsos et al. \`99, Kleinberg et al. \`99, Chakrabarti et al. \`04, Newman \`04] **L02.** Triangle Power Law (TPL) [Tsourakakis \`08] **L03.** Eigenvalue Power Law (EPL) [Siganos et al. \`03] **L04.** Community structure [Flake et al. \`02, Girvan and Newman \`02]	**L10.** Snapshot Power Law (SPL) [McGlohon et al. \`08]
Dynamic	**L05.** Densification Power Law (DPL) [Leskovec et al. \`05] **L06.** Small and shrinking diameter [Albert and Barabási \`99, Leskovec et al. \`05] **L07.** Constant size 2nd and 3rd connected components [McGlohon et al. \`08] **L08.** Principal Eigenvalue Power Law (λ_1PL) [Akoglu et al. \`08] **L09.** Bursty/self-similar edge/weight additions [Gomez and Santonja \`98, Gribble et al. \`98, Crovella and	**L11.** Weight Power Law (WPL) [McGlohon et al. \`08]

RTG: A Recursive Realistic Graph Generator using Random Typing Leman Akoglu and Christos Faloutsos. *PKDD*’09.

Carnegie Mellon

MORE Graph Patterns

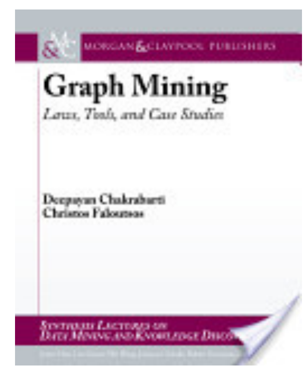
	Unweighted	Weighted
Static	✔ L01. Power-law degree distribution [Faloutsos et al. `99, Kleinberg et al. `99, Chakrabarti et al. `04, Newman `04] ✔ L02. Triangle Power Law (TPL) [Tsourakakis `08] ✔ L03. Eigenvalue Power Law (EPL) [Siganos et al. `03] L04. Community structure [Flake et al. `02, Girvan and Newman `02]	✔ L10. Snapshot Power Law (SPL) [McGlohon et al. `08]
Dynamic	✔ L05. Densification Power Law (DPL) [Leskovec et al. `05] L06. Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05] L07. Constant size 2nd and 3rd connected components [McGlohon et al. `08] ✔ L08. Principal Eigenvalue Power Law (λ_1PL) [Akoglu et al. `08] L09. Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et al. `98, Crovella and	✔ L11. Weight Power Law (WPL) [McGlohon et al. `08]

RTG: A Recursive Realistic Graph Generator using Random Typing Leman Akoglu and Christos Faloutsos. *PKDD*'09.

MORE Graph Patterns

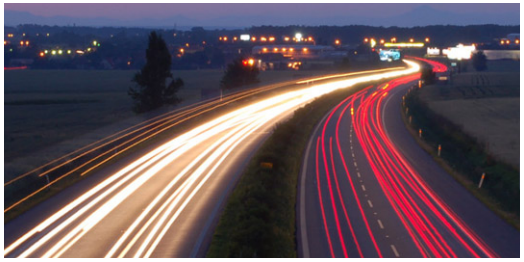
	Unweighted	Weighted
Static	L01. Power-law degree distribution [Faloutsos et al. `99, Kleinberg et al. `99, Chakrabarti et al. `04, Newman `04] L02. Triangle Power Law (TPL) [Tsourakakis `08] L03. Eigenvalue Power Law (EPL) [Siganos et al. `03] L04. Community structure [Flake et al. `02, Girvan and Newman `02]	L10. Snapshot Power Law (SPL) [McGlohon et al. `08]
Dynamic	L05. Densification Power Law (DPL) [Leskovec et al. `05] L06. Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05] L07. Constant size 2nd and 3rd connected components [McGlohon et al. `08] L08. Principal Eigenvalue Power Law (λ_1PL) [Akoglu et al. `08] L09. Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et al. `98, Crovella and Bestavros `99, McGlohon et al. `08]	L11. Weight Power Law (WPL) [McGlohon et al. `08]

- Mary McGlohon, Leman Akoglu, Christos Faloutsos. *Statistical Properties of Social Networks.* in "Social Network Data Analytics" (Ed.: Charu Aggarwal)



- Deepayan Chakrabarti and Christos Faloutsos, *Graph Mining: Laws, Tools, and Case Studies* Oct. 2012, Morgan Claypool.

Roadmap

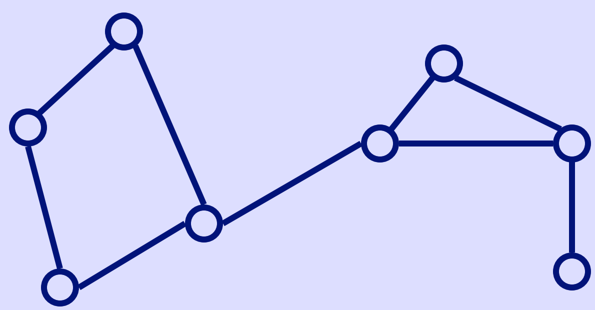


- Introduction – Motivation
- Part#1: Patterns in graphs
  - Some (power) laws
  - The 'no good cuts' shock
  - A possible explanation: fractals
- Part#2: Cascade analysis
- Conclusions

Carnegie Mellon

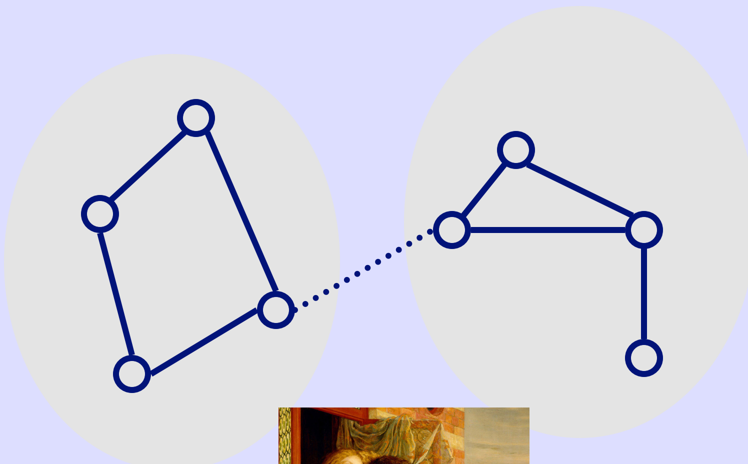
Background: Graph cut problem

- Given a graph, and *k*
- Break it into *k* (disjoint) communities

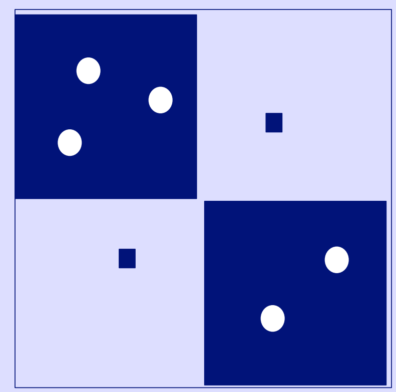


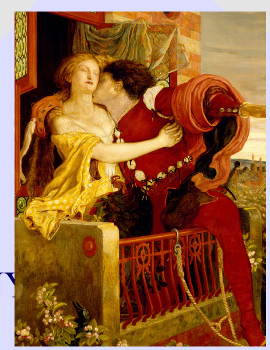
Graph cut problem

- Given a graph, and *k*
- Break it into *k* (disjoint) communities
- (assume: block diagonal = ‘cavemen’ graph)



$k = 2$





Many algo's for graph partitioning


- METIS [Karypis, Kumar +]
- 2nd eigenvector of Laplacian
- Modularity-based [Girwan+Newman]
- Max flow [Flake+]
- …
- …
- …

Carnegie Mellon

Strange behavior of min cuts

- Subtle details: next
  - Preliminaries: min-cut plots of ‘usual’ graphs

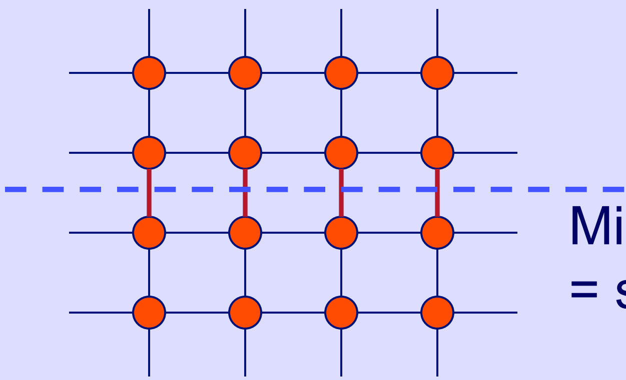
NetMine: New Mining Tools for Large Graphs, by D. Chakrabarti, Y. Zhan, D. Blandford, C. Faloutsos and G. Blelloch, in the SDM 2004 Workshop on Link Analysis, Counter-terrorism and Privacy

Statistical Properties of Community Structure in Large Social and Information Networks, J. Leskovec, K. Lang, A. Dasgupta, M. Mahoney. WWW 2008.

"Min-cut" plot

- Do min-cuts recursively.

log (mincut-size / #edges)



Mincut size
= sqrt(N)

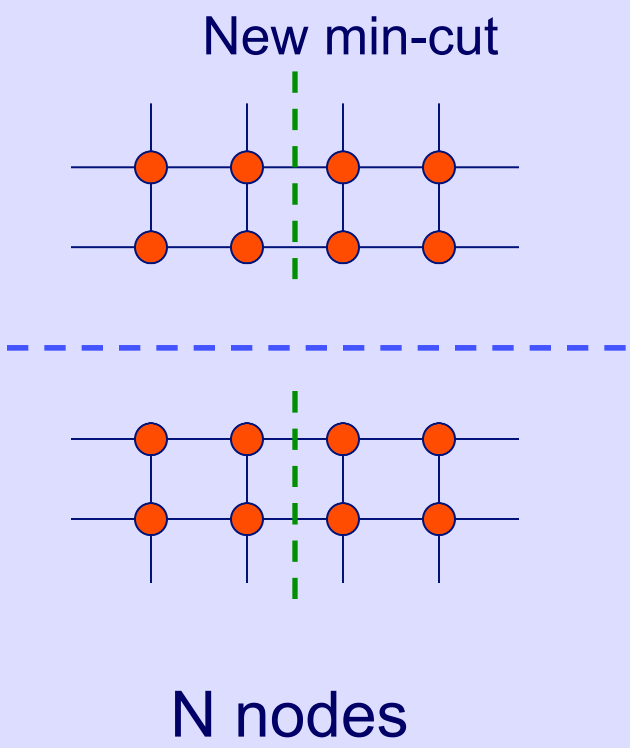
log (# edges)

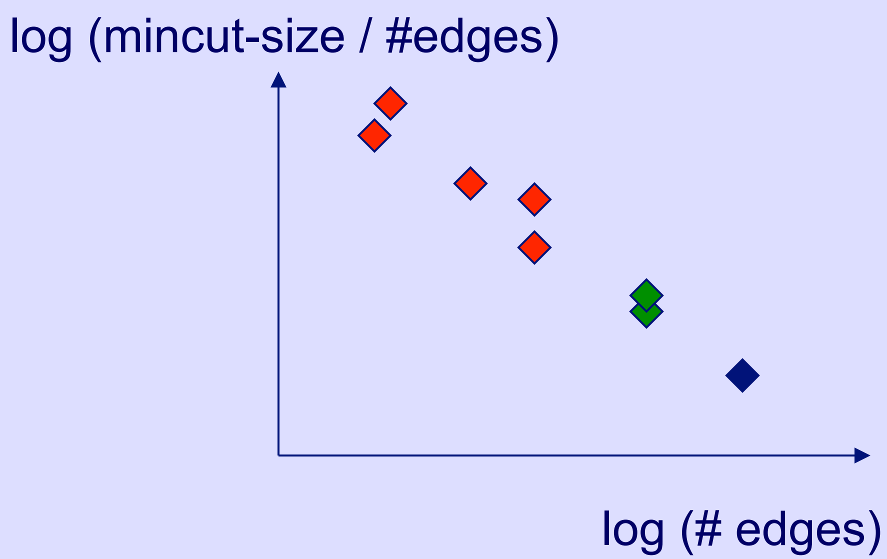
N nodes

Carnegie Mellon

"Min-cut" plot

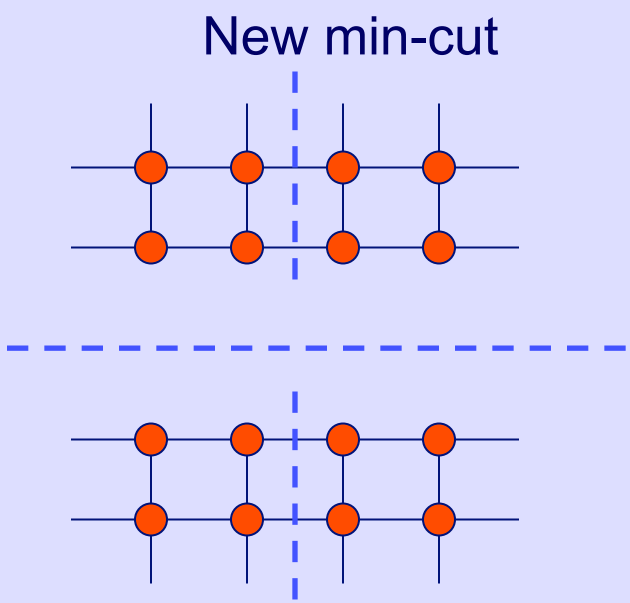
- Do min-cuts recursively.



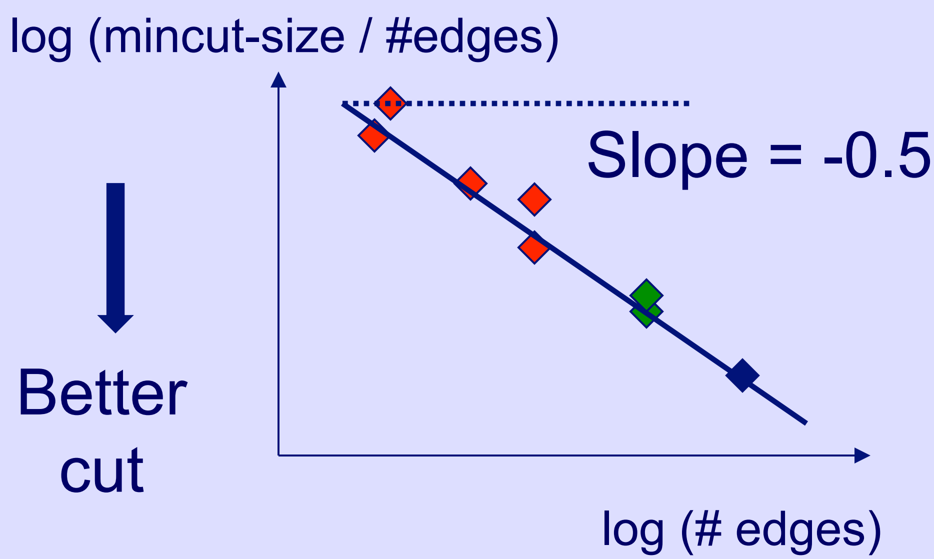


"Min-cut" plot

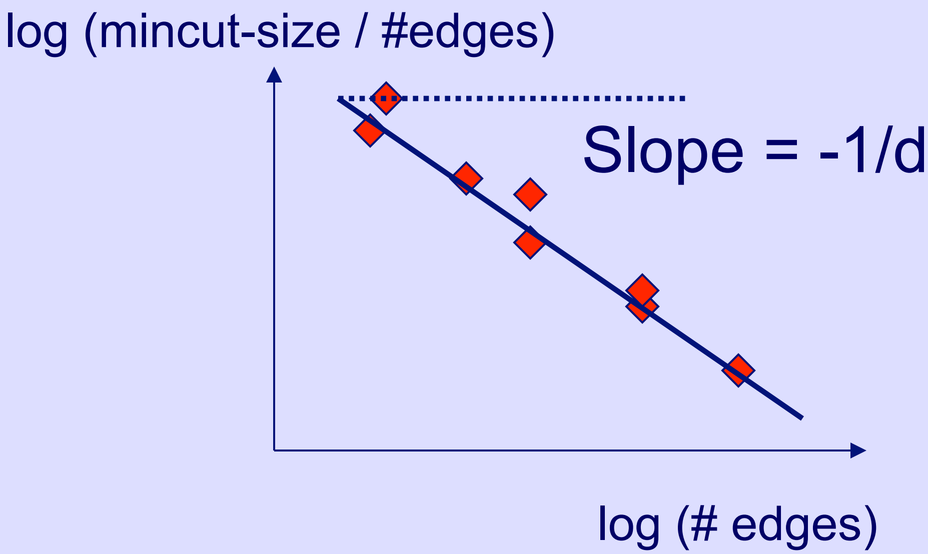
- Do min-cuts recursively.

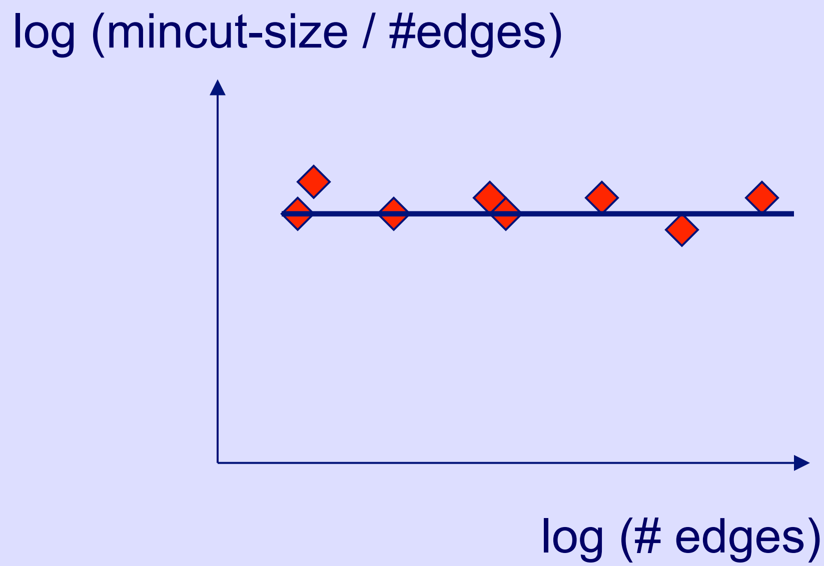


N nodes

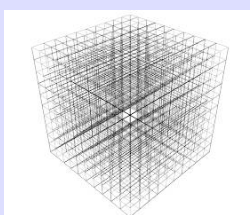


"Min-cut" plot





For a d-dimensional grid, the slope is -1/d



For a random graph (and clique), the slope is 0

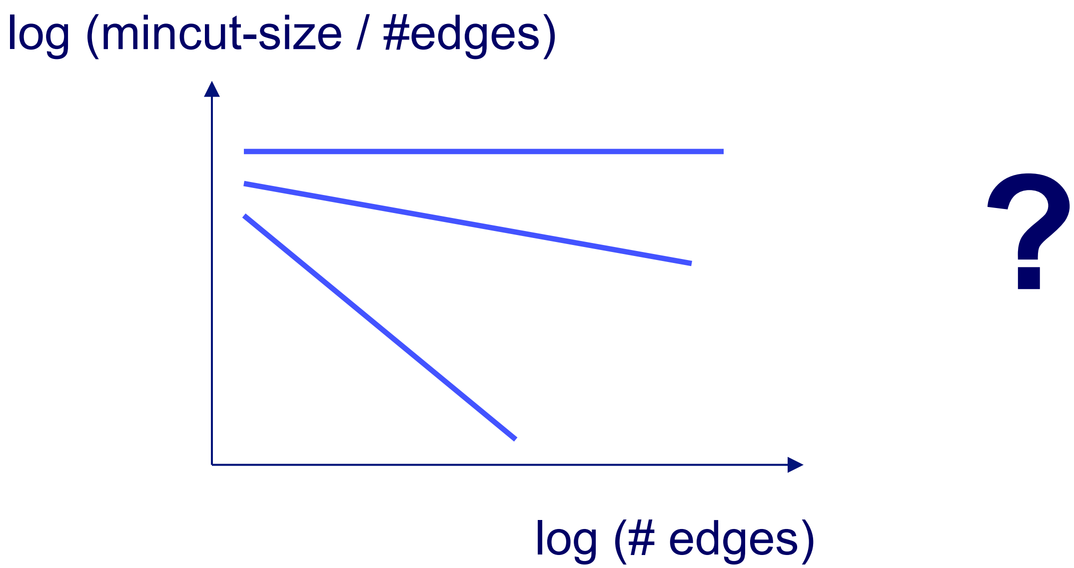
Experiments

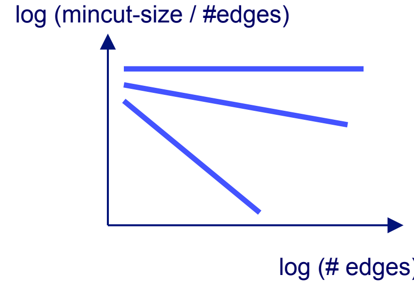
- Datasets:
  - Google Web Graph: 916,428 nodes and 5,105,039 edges
  - Lucent Router Graph: Undirected graph of network routers from www.isi.edu/scan/mercator/maps.html; 112,969 nodes and 181,639 edges
  - User → Website Clickstream Graph: 222,704 nodes and 952,580 edges

NetMine: New Mining Tools for Large Graphs, by D. Chakrabarti, Y. Zhan, D. Blandford, C. Faloutsos and G. Blelloch, in the SDM 2004 Workshop on Link Analysis, Counter-terrorism and Privacy

"Min-cut" plot

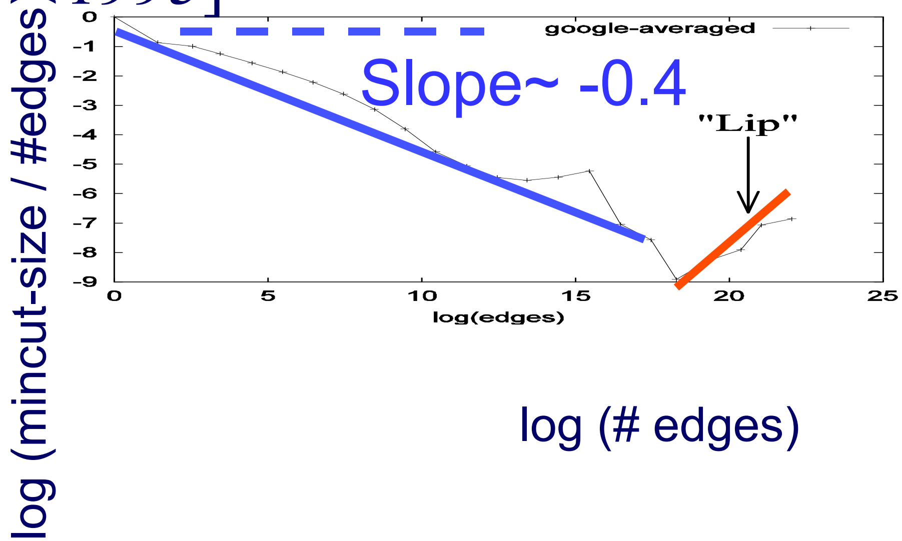
- What does it look like for a real-world graph?





Experiments

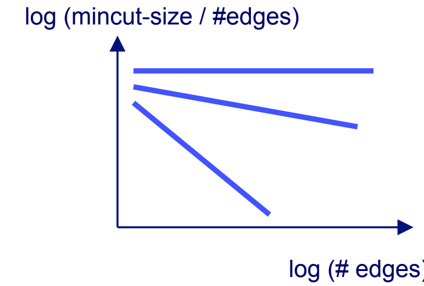
- Used the METIS algorithm [Karypis, Kumar, 1995]



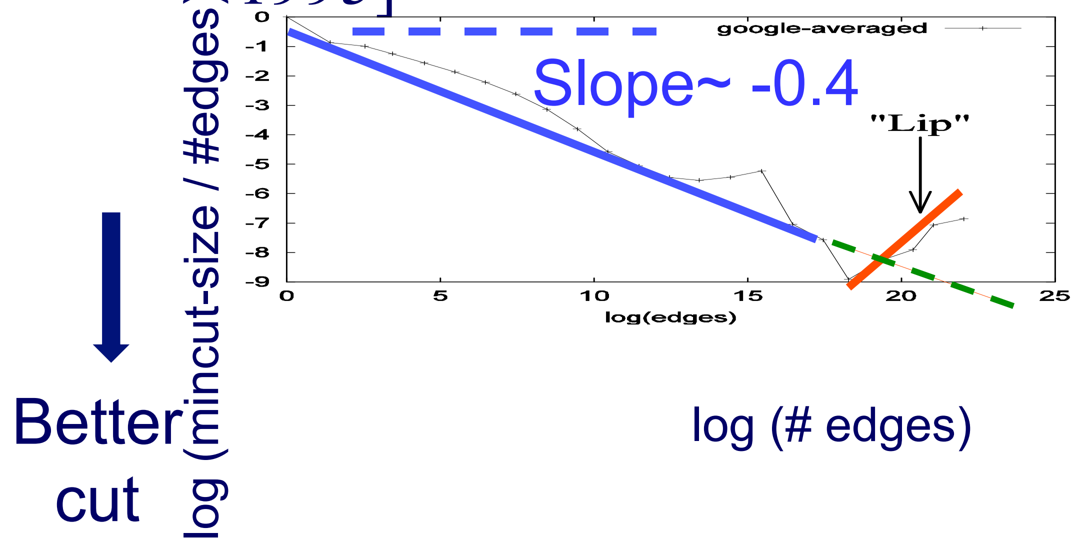
- Google Web graph
- Values along the y-axis are averaged
- "lip" for large # edges
- Slope of -0.4, corresponds to a 2.5-dimensional grid!

Carnegie Mellon

Experiments



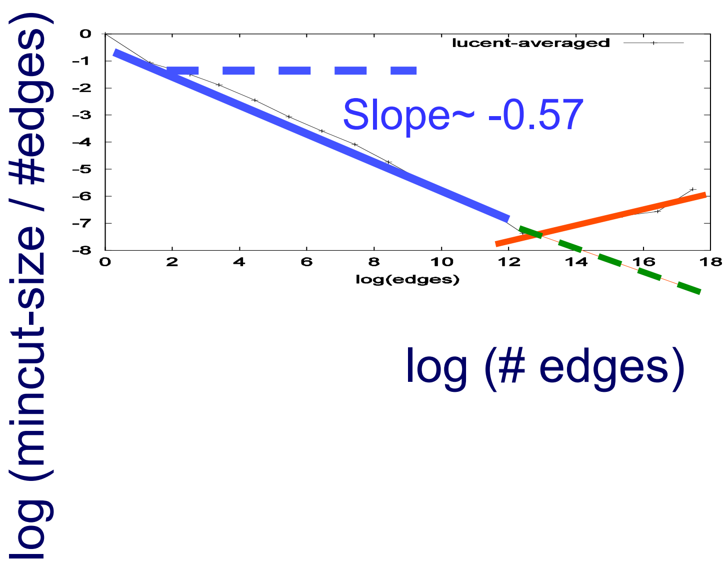
- Used the METIS algorithm [Karypis, Kumar, 1995]



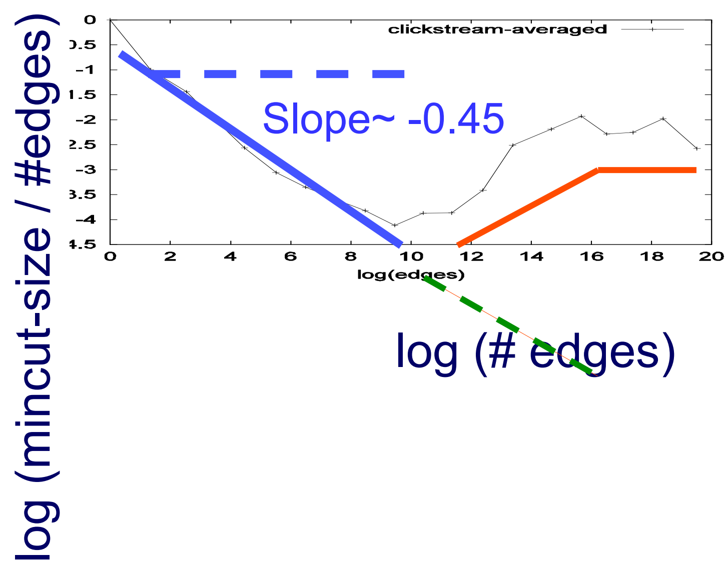
- Google Web graph
- Values along the y-axis are averaged
- "lip" for large # edges
- Slope of -0.4, corresponds to a 2.5-dimensional grid!

Experiments

- Same results for other graphs too…



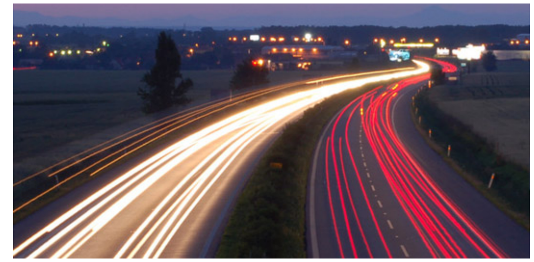
Lucent Router graph



Clickstream graph

Carnegie Mellon

Roadmap



- Introduction – Motivation
- Part#1: Patterns in graphs
  - Some (power) laws
  - The 'no good cuts' shock
  - ➡ A possible explanation: fractals
- Part#2: Cascade analysis
- Conclusions

2 Questions, one answer

- Q1: why so many power laws
- Q2: why no ‘good cuts’?

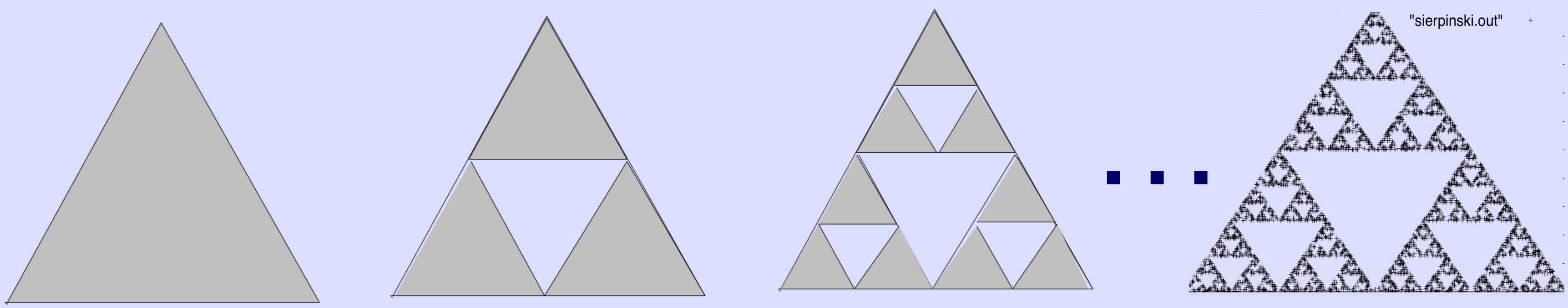
possible

2 Questions, one answer

- Q1: why so many power laws
- Q2: why no 'good cuts'?
- A: Self-similarity = fractals = 'RMAT' ~ 'Kronecker graphs'

20'' intro to fractals

- Remove the middle triangle; repeat
- -> Sierpinski triangle
- (Bonus question - dimensionality?
  - >1 (inf. perimeter – $(4/3)^\infty$)
  - <2 (zero area – $(3/4)^\infty$)



Carnegie Mellon

20'' intro to fractals

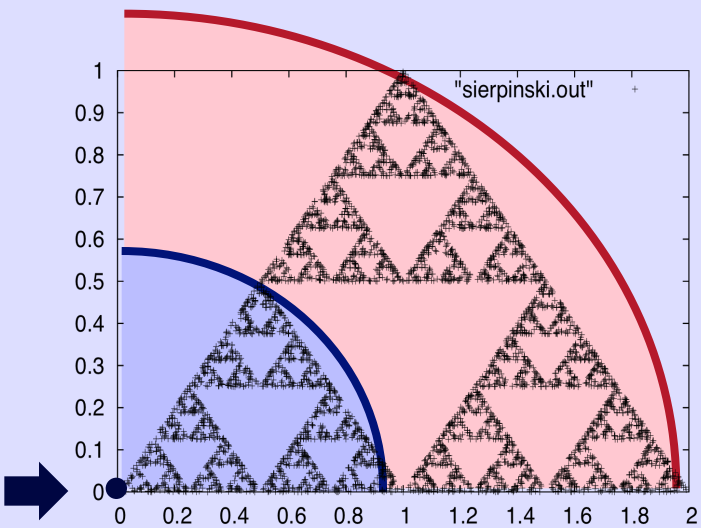
Self-similarity -> no char. scale

-> power laws, eg:

2x the radius,

3x the #neighbors nn(r)

$$nn(r) = C\ r^{\log 3/\log 2}$$



20'' intro to fractals

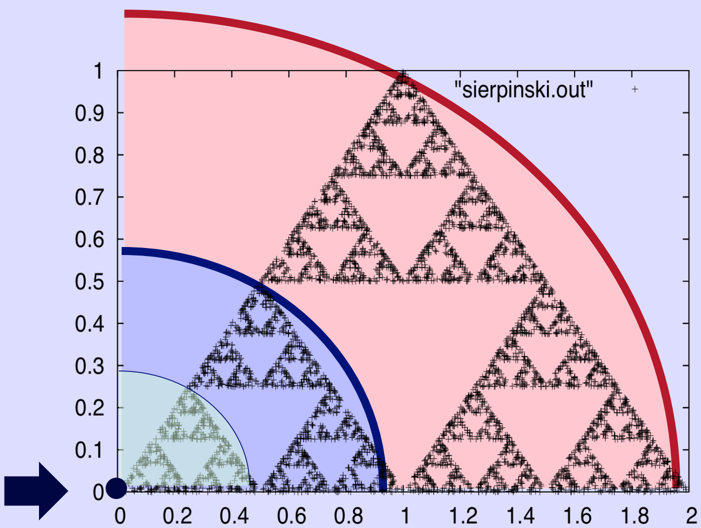
Self-similarity -> no char. scale

-> power laws, eg:

2x the radius,

3x the #neighbors nn(r)

$$nn(r) = C\ r^{\log 3/\log 2}$$



Carnegie Mellon

20'' intro to fractals

Self-similarity -> no char. scale

-> power laws, eg:

2x the radius,

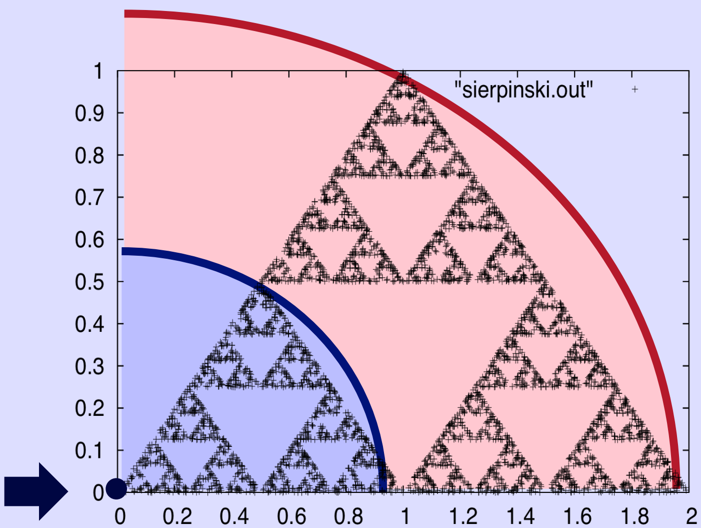
3x the #neighbors

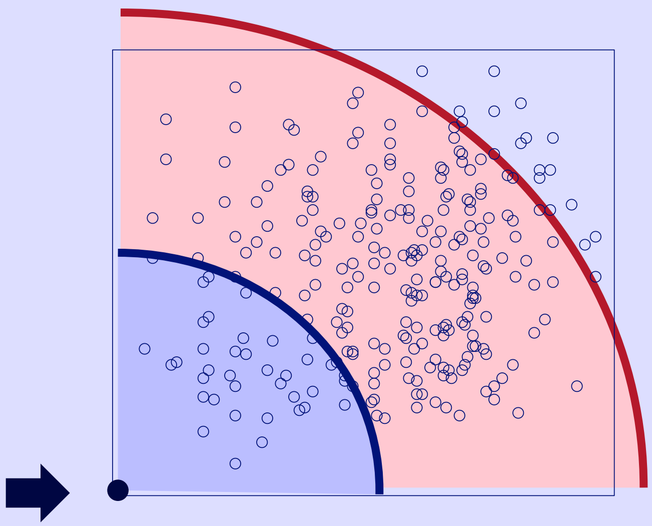
$nn = C\ r^{\log 3/\log 2}$

2x the radius,

4x neighbors

$nn = C\ r^{\log 4/\log 2} = C\ r^{2}$





20'' intro to fractals

Self-similarity -> no char. scale

-> power laws, eg:

2x the radius,

3x the #neighbors

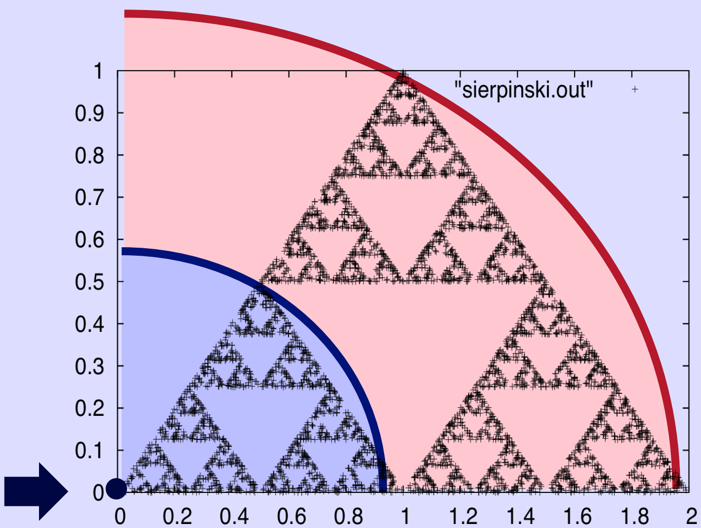
$nn = C\, r^{\log 3/\log 2}$ =1.58

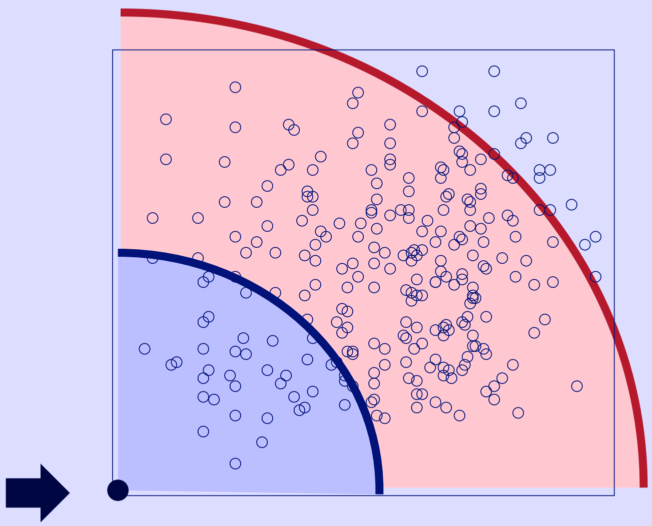
2x the radius,

4x neighbors

$nn = C\, r^{\log 4/\log 2} = C\, r^{2}$

Fractal dim.





20'' intro to fractals

Self-similarity -> no char. scale

-> **power laws**, eg:

2x the radius,

3x the #neighbors

$nn = C\ r^{\log 3/\log 2}$

2x the radius,

4x neighbors

$nn = C\ r^{\log 4/\log 2} = C\ r^{2}$

Fractal dim.

How does self-similarity help in graphs?

- A: RMAT/Kronecker generators
  - With self-similarity, we get all power-laws, automatically,
  - And small/shrinking diameter
  - And `no good cuts'

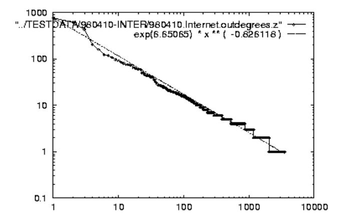
R-MAT: A Recursive Model for Graph Mining,
by D. Chakrabarti, Y. Zhan and C. Faloutsos,
SDM 2004, Orlando, Florida, USA

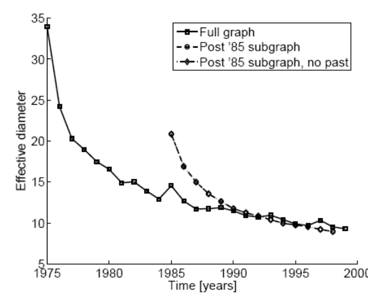
Realistic, Mathematically Tractable Graph Generation and Evolution, Using Kronecker Multiplication,
by J. Leskovec, D. Chakrabarti, J. Kleinberg,
and C. Faloutsos, in PKDD 2005, Porto, Portugal

Carnegie Mellon

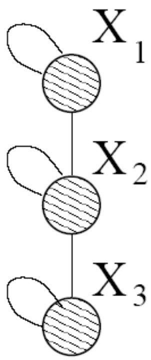
Graph gen.: Problem dfn

- Given a growing graph with count of nodes N_1, N_2, ...
- Generate a realistic sequence of graphs that will obey all the patterns
  - Static Patterns
    - S1 Power Law Degree Distribution
    - S2 Power Law eigenvalue and eigenvector distribution
    - Small Diameter
  - Dynamic Patterns
    - T2 Growth Power Law (2x nodes; 3x edges)
    - T1 Shrinking/Stabilizing Diameters





Kronecker Graphs

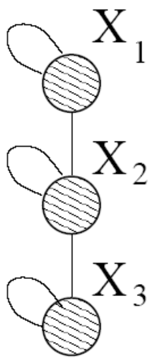


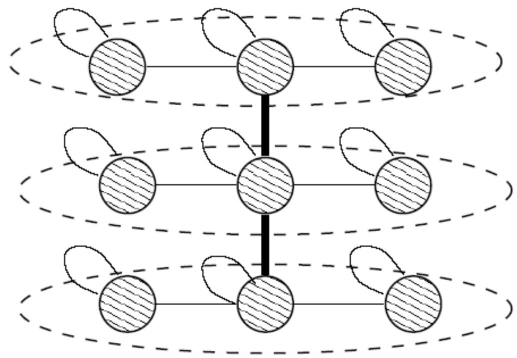
1	1	0
1	1	1
0	1	1

G_1

Adjacency matrix

Kronecker Graphs





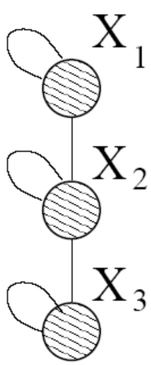
Intermediate stage

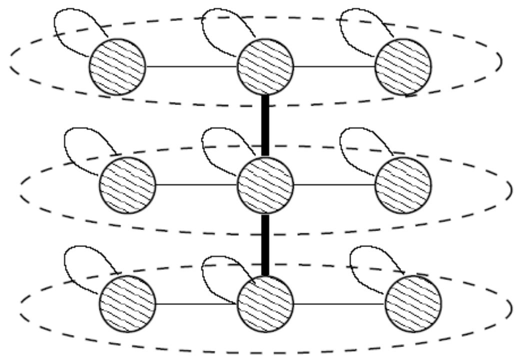
1	1	0
1	1	1
0	1	1

G_1

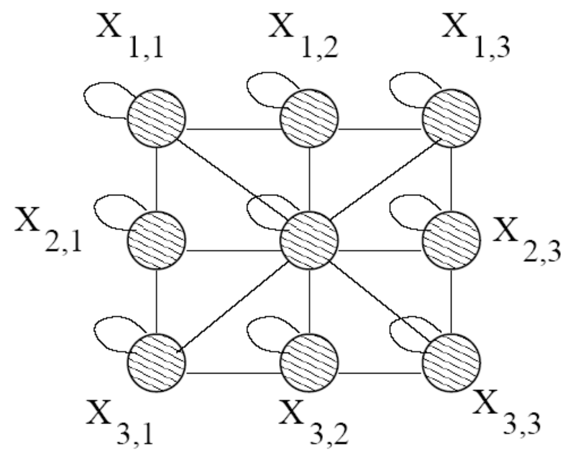
Adjacency matrix

Kronecker Graphs





Intermediate stage



1	1	0
1	1	1
0	1	1

G_1

Adjacency matrix

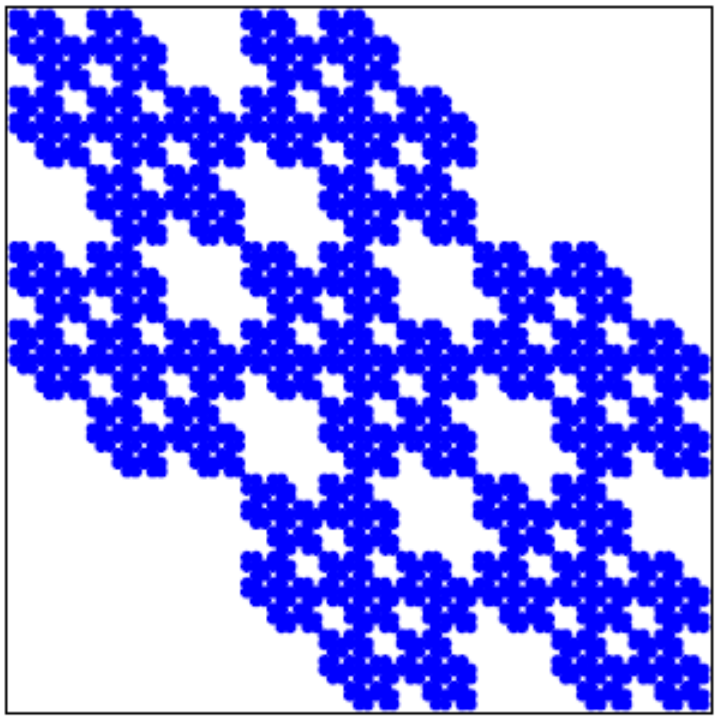
G_1	G_1	0
G_1	G_1	G_1
0	G_1	G_1

$$G_2 = G_1 \otimes G_1$$

Adjacency matrix

Kronecker Graphs

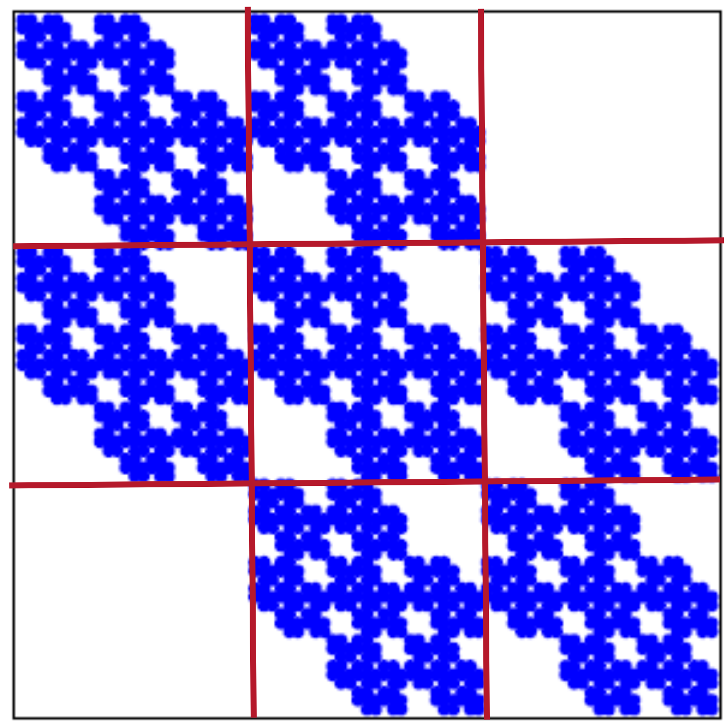
- Continuing multiplying with G_1 we obtain G_4 and so on …



G_4 adjacency matrix

Kronecker Graphs

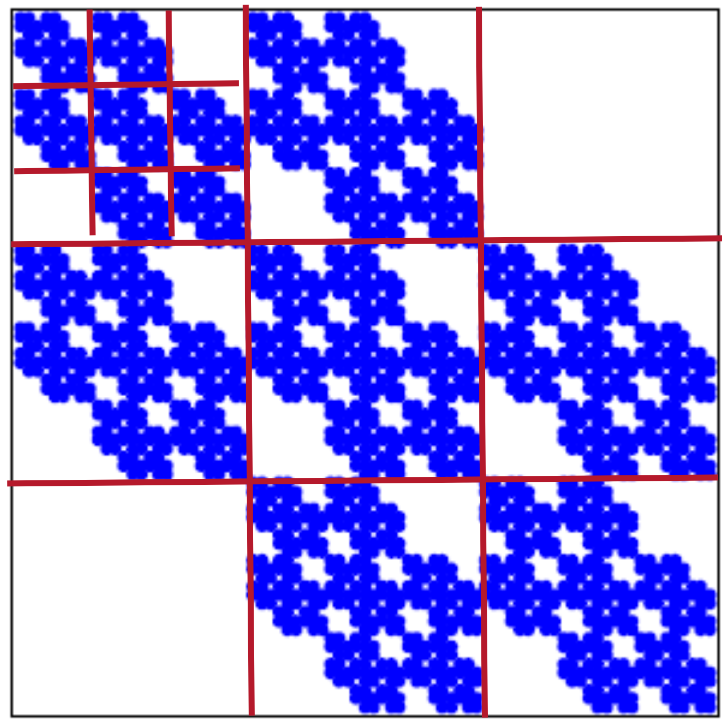
- Continuing multiplying with G_1 we obtain G_4 and so on …



G_4 adjacency matrix

Kronecker Graphs

- Continuing multiplying with G_1 we obtain G_4 and so on …

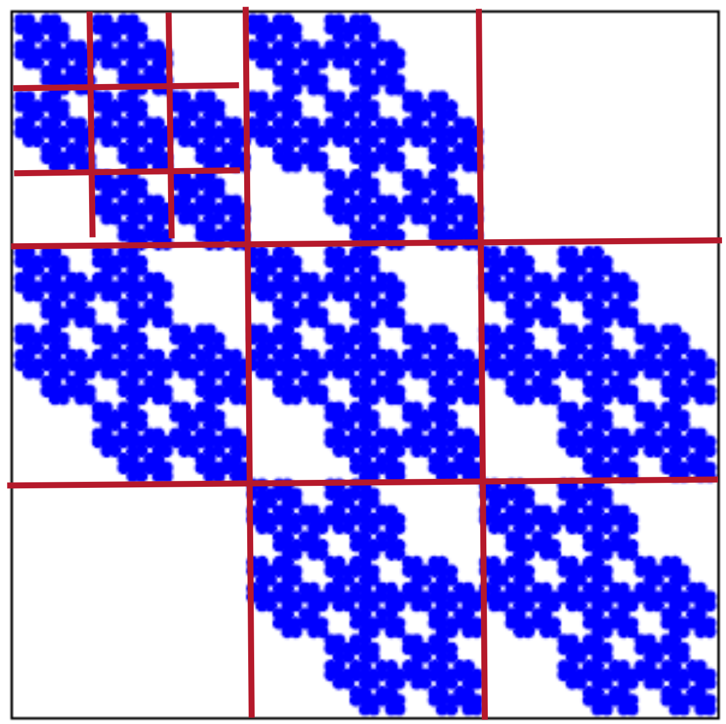


G_4 adjacency matrix

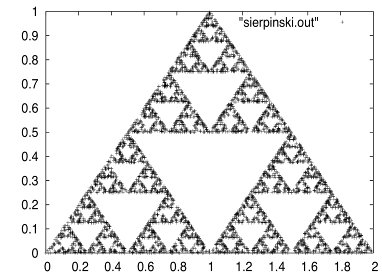
Kronecker Graphs

- Continuing multiplying with G_1 we obtain G_4 and so on …

Holes within holes; Communities within communities



G_4 adjacency matrix



Problem Definition

- Given a growing graph with nodes N_1, N_2, …
- Generate a realistic sequence of graphs that will obey all the patterns
  - Static Patterns
    - ✓ Power Law Degree Distribution
    - ✓ Power Law eigenvalue and eigenvector distribution
    - ✓ Small Diameter
  - Dynamic Patterns
    - ✓ Growth Power Law
    - ✓ Shrinking/Stabilizing Diameters
- First generator for which we can **prove** all these properties

Impact: Graph500

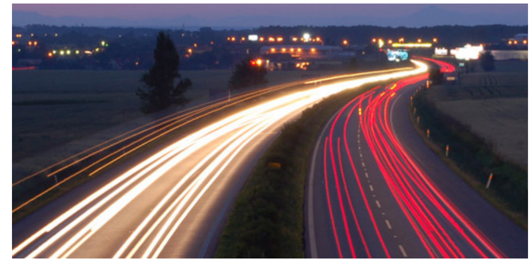
- Based on RMAT (= 2x2 Kronecker)
- Standard for graph benchmarks
- http://www.graph500.org/
- Competitions 2x year, with all major entities: LLNL, Argonne, ITC-U. Tokyo, Riken, ORNL, Sandia, PSC, …

To iterate is human, to recurse is devine

R-MAT: A Recursive Model for Graph Mining,
by D. Chakrabarti, Y. Zhan and C. Faloutsos,
SDM 2004, Orlando, Florida, USA

Carnegie Mellon

Roadmap



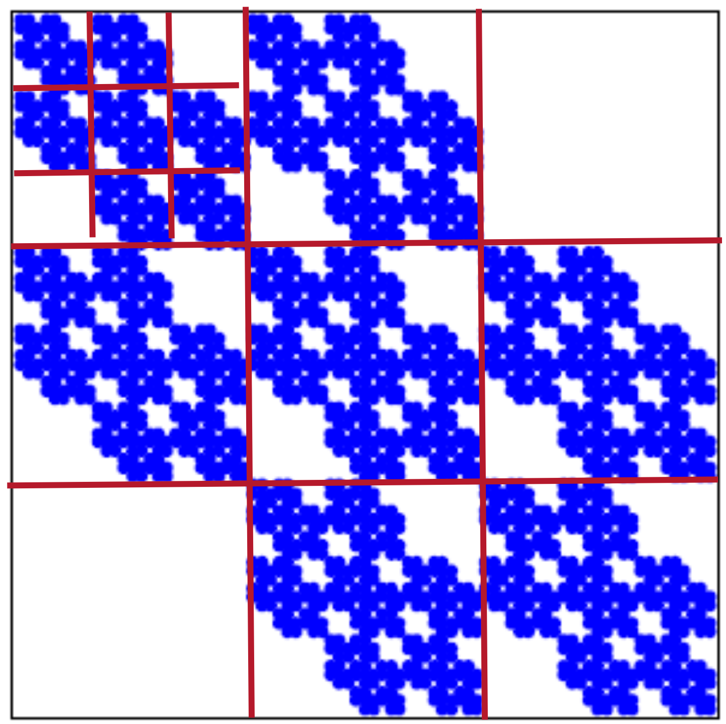
- Introduction – Motivation
- Part#1: Patterns in graphs
  - …
  - Q1: Why so many power laws?
  - ➡ Q2: Why no 'good cuts'?
- Part#2: Cascade analysis
- Conclusions

A: real graphs -> self similar -> power laws

REMINDER

Kronecker Product – a Graph

- Continuing multiplying with G_1 we obtain G_4 and so on …



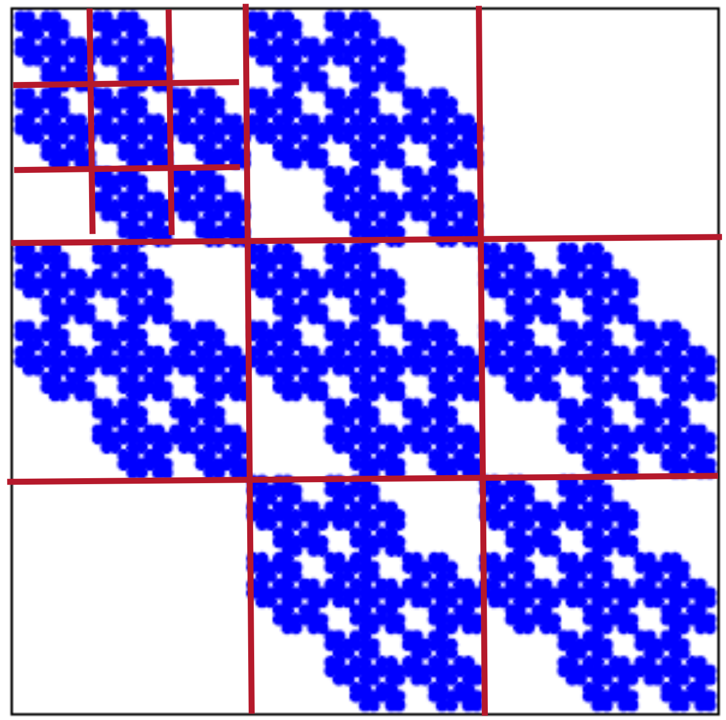
G_4 adjacency matrix

REMINDER

Kronecker Product – a Graph

- Continuing multiplying with G_1 we obtain G_4 and so on …

Communities within communities within communities …



How many Communities?
3?
9?
27?

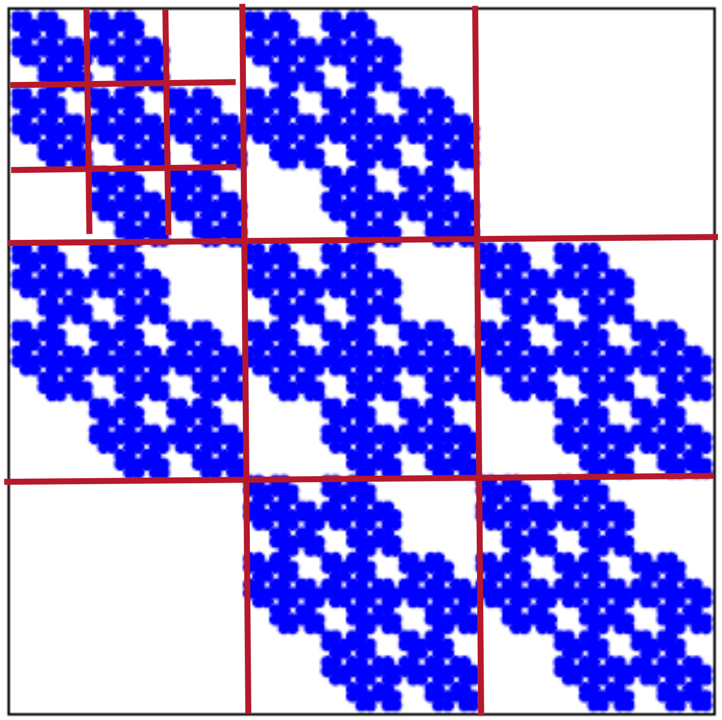
G_4 adjacency matrix

REMINDER

Kronecker Product – a Graph

- Continuing multiplying with G_1 we obtain G_4 and so on …

Communities within communities within communities …

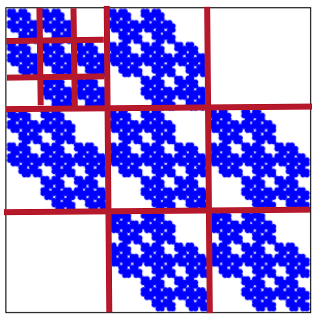


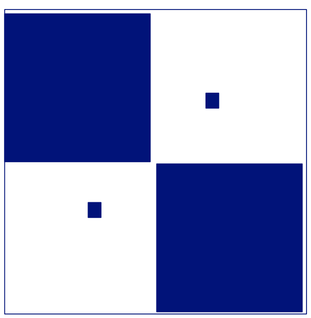
G_4 adjacency matrix

How many Communities?
3?
9?
27?

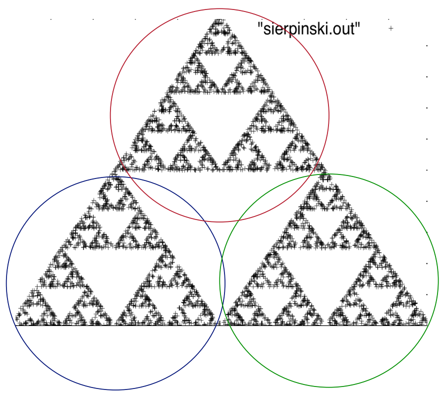
A: one – but not a typical, block-like community…

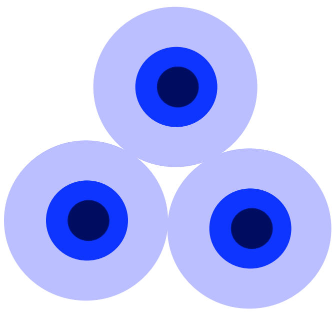
Communities?



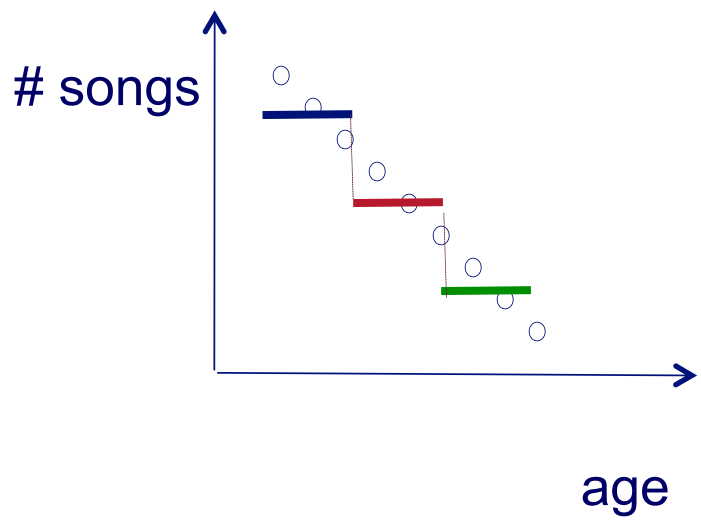


(Gaussian) Clusters?

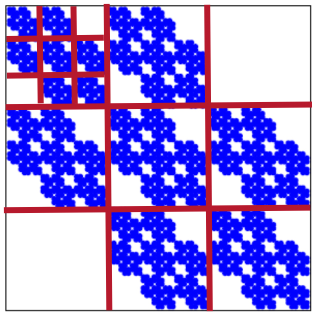


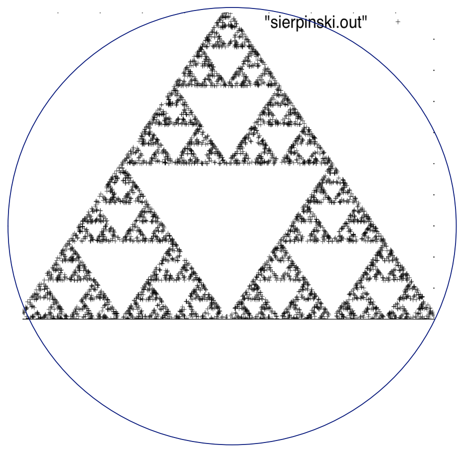


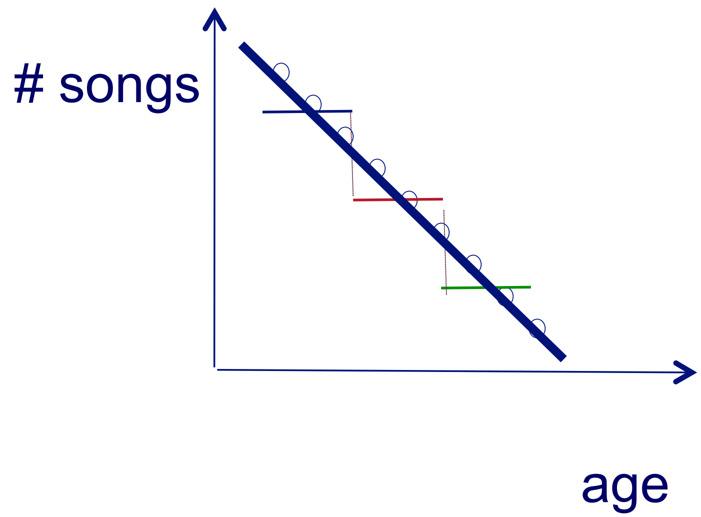
Piece-wise flat parts?

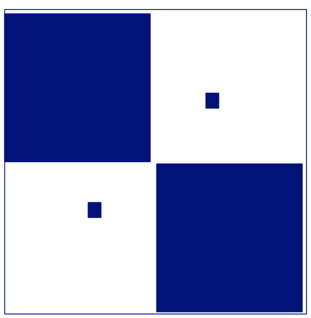


Carnegie Mellon



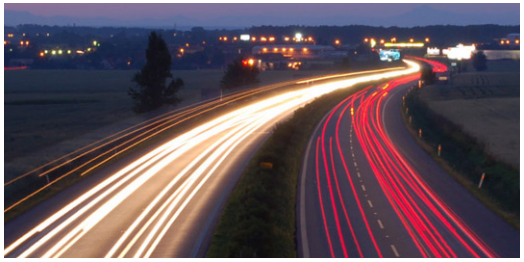






Wrong questions to ask!

Roadmap

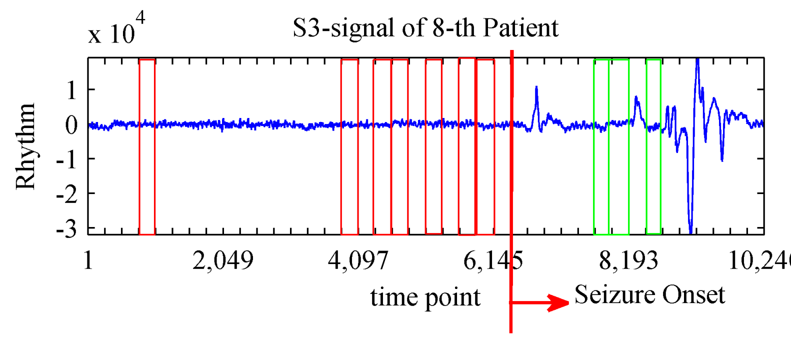


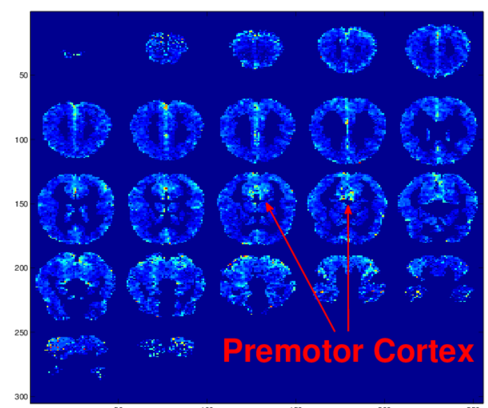
- Introduction – Motivation
- Part#1: Patterns in graphs
  - …
  - Q1: The 'no good cuts' shock
  - Q2: Why no 'good cuts'?
- ➡ What next?
- Conclusions

Carnegie Mellon

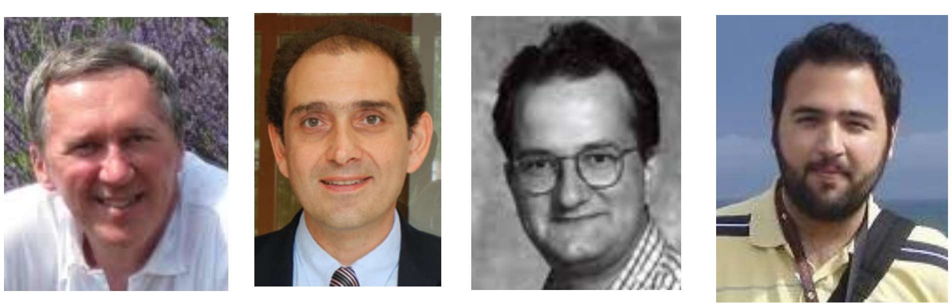
Challenge #1: ‘Connectome’ – brain wiring

- Which neurons get activated by ‘tomato’
- How wiring evolves
- Modeling epilepsy





\`glass’
\`tomato’
\`bell’

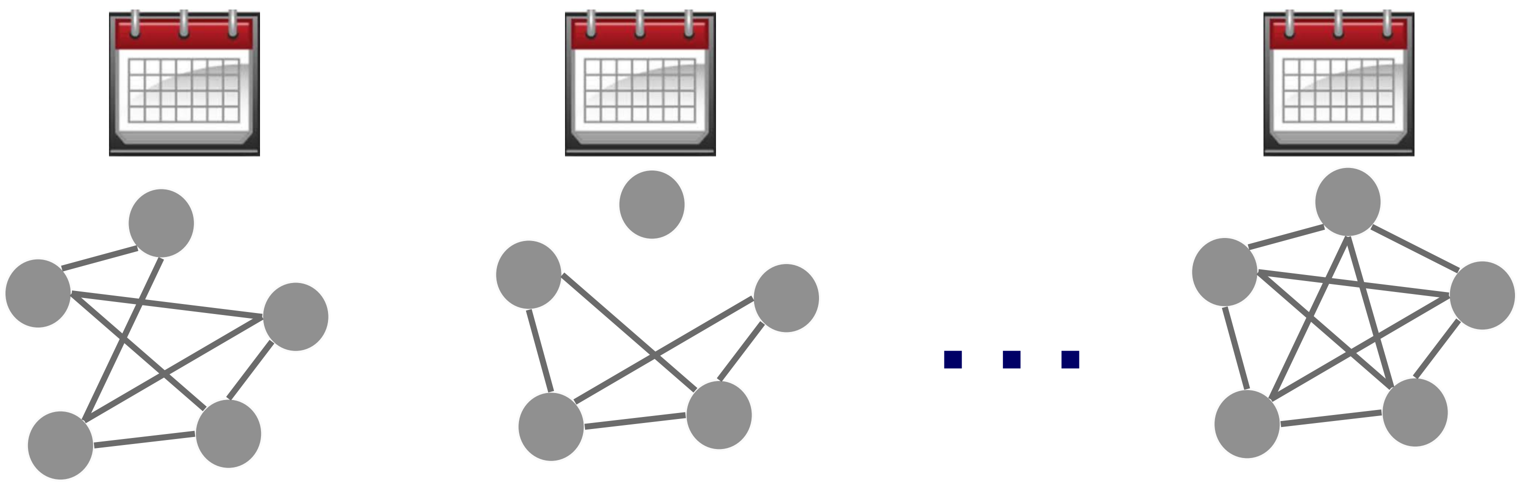


Tom Mitchell George Karypis

N. Sidiropoulos V. Papalexakis

Challenge#2: Time evolving networks / tensors

- Periodicities? Burstiness?
- What is ‘typical’ behavior of a node, over time
- Heterogeneous graphs (= nodes w/ attributes)



Carnegie Mellon

Summary

- *many* patterns in real graphs
  - Power-laws everywhere
  - ‘no good cuts’
- Self-similarity (RMAT/Kronecker): good model

Carnegie Mellon

Thanks

Disclaimer: All opinions are mine; not necessarily reflecting the opinions of the funding agencies

Thanks to: NSF IIS-0705359, IIS-0534205, CTA-INARC; Yahoo (M45), LLNL, IBM, SPRINT, Google, INTEL, HP, iLab

Project info: PEGASUS

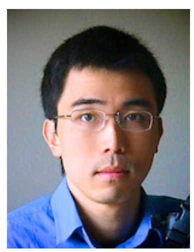
www.cs.cmu.edu/~pegasus

Results on large graphs: with Pegasus + hadoop + M45

Apache license

Code, papers, manual, video

Prof. U Kang



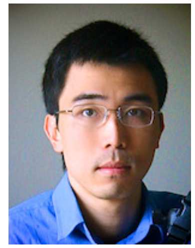
Prof. Polo Chau

Carnegie Mellon

Cast

Akoglu, Leman

Beutel, Alex



Chau, Polo

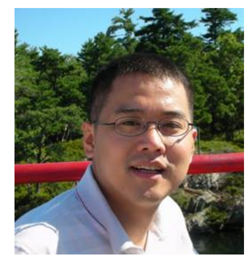
Kang, U

Koutra, Danai

McGlohon, Mary

Prakash, Aditya

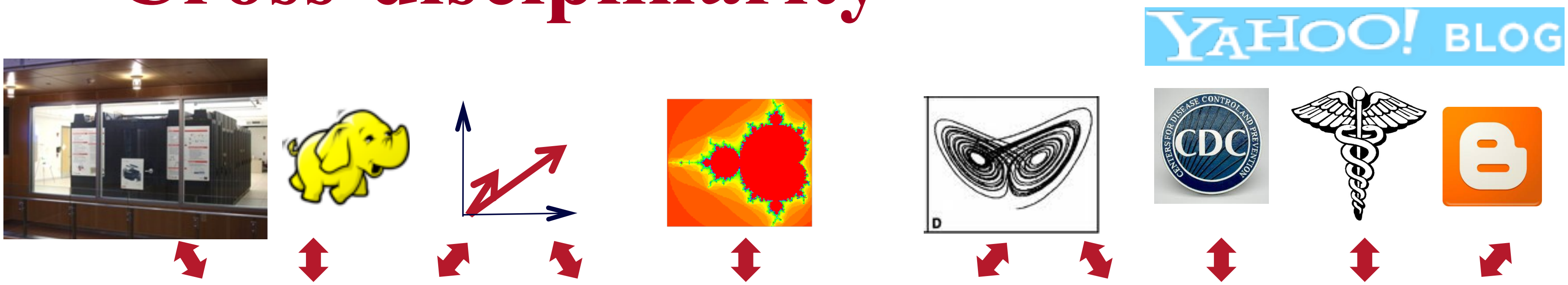
Papalexakis, Vagelis

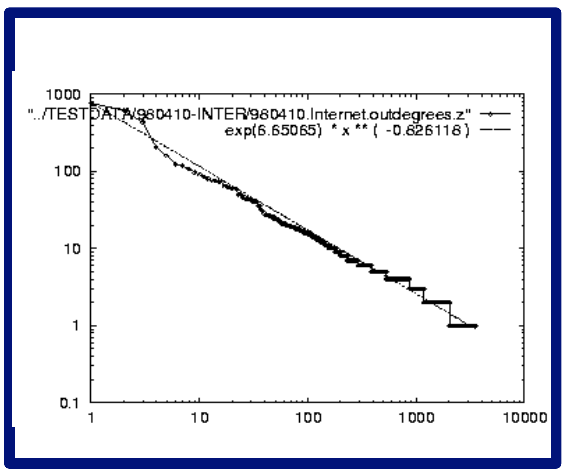


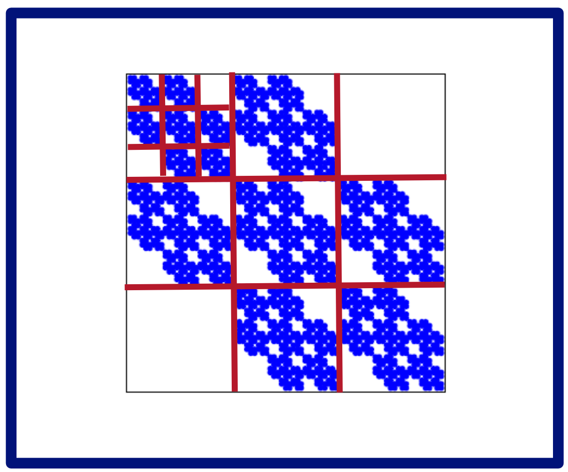
Tong, Hanghang

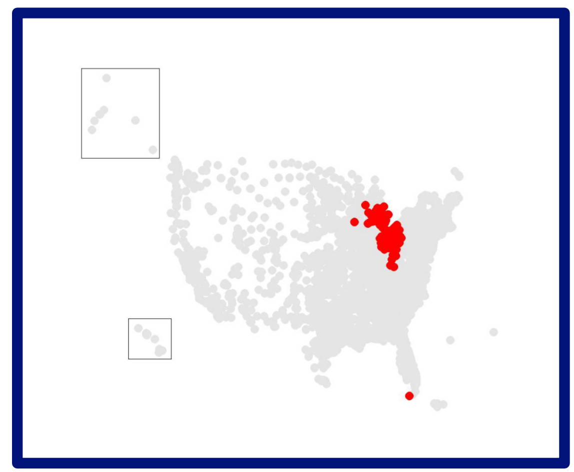
TAKE HOME MESSAGE:

Cross-disciplinarity









www.cs.cmu.edu/~christos/TALKS/13-10-WIN/