CMU SCS

Large Graph Mining: Patterns, Tools and Case Studies

Christos Faloutsos
Hanghang Tong
CMU

 Copyright: Faloutsos, Tong (2008)

CMU SCS

Thanks

- Deepayan Chakrabarti (CMU -> Yahoo)

- Michalis Faloutsos (UCR)

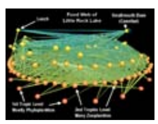
- George Siganos (UCR)

 Copyright: Faloutsos, Tong (2008)

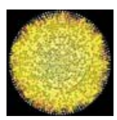
CMU SCS

Introduction

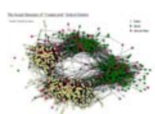
Internet Map
[lumeta.com]



Food Web
[Martinez '91]



Protein Interactions
[genomebiology.com]



Friendship Network
[Moody '01]

Graphs are everywhere!

 Copyright: Faloutsos, Tong (2008)

CMU SCS

Graph structures

- Physical networks
- Physical Internet
- Telephone lines
- Commodity distribution networks

 Copyright: Faloutsos, Tong (2008)

CMU SCS

Networks derived from "behavior"

- Telephone call patterns
- Email, Blogs, Web, Databases, XML
- Language processing
- Web of trust, epinions

 Copyright: Faloutsos, Tong (2008)

CMU SCS

Outline

- ➡ Part 1: Patterns
- Part 2: Matrix and Tensor Tools
- Part 3: Proximity
- Part 4: Case Studies

 Copyright: Faloutsos, Tong (2008)

CMU SCS

Outline

- Topology & 'laws'
- Generators
- Discussion

Motivating questions:

CMU SCS

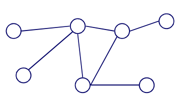
Motivating questions

- What do real graphs look like?
  - What properties of nodes, edges are important to model?
  - What local and global properties are important to measure?
- How to generate realistic graphs?

CMU SCS

Motivating questions

Given a graph:



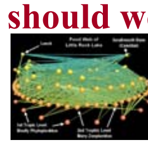
•Are there un-natural sub-graphs? (criminals' rings or terrorist cells)?

• How do P2P networks evolve?

CMU SCS

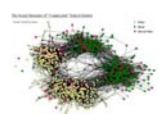
Why should we care?

Internet Map [lumeta.com]



Food Web [Martinez '91]

Protein Interactions [genomebiology.com]



Friendship Network [Moody '01]

CMU SCS

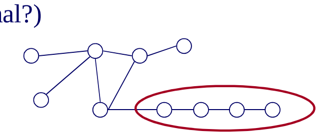
Why should we care?

- **A1: extrapolations**: how will the Internet/Web look like next year?
- **A2: algorithm design**: what is a realistic network topology,
  - to try a new routing protocol?
  - to study virus/rumor propagation, and immunization?

CMU SCS

Why should we care? (cont'd)

- **A3: Sampling**: How to get a 'good' sample of a network?
- **A4: Abnormalities**: is this sub-graph / sub-community / sub-network 'normal'? (what **is** normal?)



CMU SCS

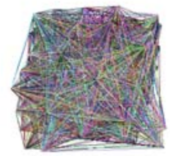
Outline

- Topology & 'laws'
  - Static graphs
  - Evolving graphs
  - Weighted graphs
- Generators
- Discussion

CMU SCS

Topology

How does the Internet look like? Any rules?



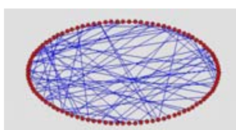
(Looks random – right?)

CMU SCS

Are real graphs random?

- random (Erdos-Renyi) graph – 100 nodes, avg degree = 2
- before layout
- after layout
- No obvious patterns

(generated with: pajek http://vlado.fmf.uni-lj.si/pub/networks/pajek/)



CMU SCS

Laws and patterns

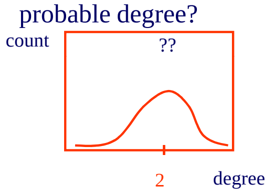
Real graphs are NOT random!!

- Diameter
- in- and out- degree distributions
- other (surprising) patterns

CMU SCS

Laws – degree distributions

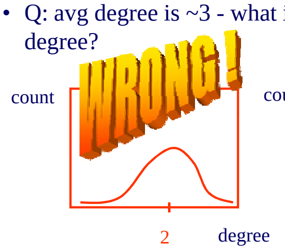
- Q: avg degree is ~2 - what is the most probable degree?

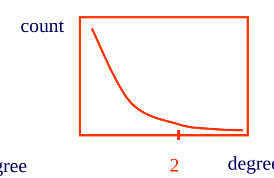


CMU SCS

Laws – degree distributions

- Q: avg degree is ~3 - what is the most probable degree?

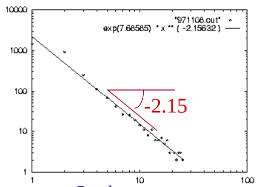




CMU SCS

I.Power-law: outdegree *O*

Frequency



Outdegree

Exponent = slope

O = -2.15

Nov'97

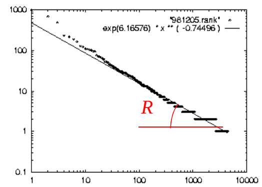
The plot is linear in log-log scale [FFF'99]

$freq = degree^{(-2.15)}$

CMU SCS

II.Power-law: rank *R*

outdegree



Rank: nodes in decreasing outdegree order

Exponent = slope

R = -0.74

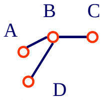
Dec'98

- The plot is a line in log-log scale

CMU SCS

III. Eigenvalues

- Let *A* be the adjacency matrix of graph
- The eigenvalue λ is:
  - $A \underline{v} = \lambda \underline{v}$, where $\underline{v}$ some vector
- Eigenvalues are strongly related to graph topology

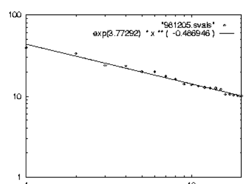


	A	B	C	D
A		1		
B	1		1	1
C		1		
D		1		

CMU SCS

III.Power-law: eigen *E*

Eigenvalue



Rank of decreasing eigenvalue

Exponent = slope

E = -0.48

Dec'98

- Eigenvalues in decreasing order (first 20)
- [Mihail+, 02]: $R = 2 * E$

CMU SCS

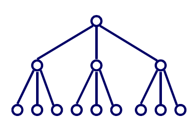
IV. The Node Neighborhood

- *N(h)* = # of pairs of nodes within *h* hops

CMU SCS

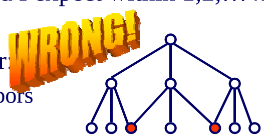
IV. The Node Neighborhood

- Q: average degree = 3 - how many neighbors should I expect within 1,2,… *h* hops?
- Potential answer:
  1 hop -> 3 neighbors
  2 hops -> 3 * 3
  …
  h hops -> 3^h



IV. The Node Neighborhood

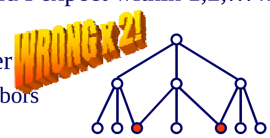
- Q: average degree = 3 - how many neighbors should I expect within 1,2,… *h* hops?
- Potential answer:
  1 hop -> 3 neighbors
  2 hops -> 3 * 3
  …
  h hops -> 3^h



WE HAVE DUPLICATES!

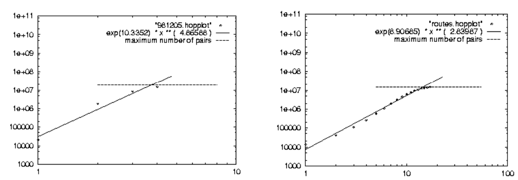
IV. The Node Neighborhood

- Q: average degree = 3 - how many neighbors should I expect within 1,2,… *h* hops?
- Potential answer:
  1 hop -> 3 neighbors
  2 hops -> 3 * 3
  …
  h hops -> 3^h



'avg' degree: meaningless!

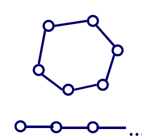
IV. Power-law: hopplot *H*



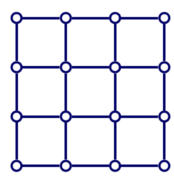
Pairs of nodes as a function of hops $N(h)= h^H$

Observation

- Q: Intuition behind 'hop exponent'?
- A: 'intrinsic=fractal dimensionality' of the network

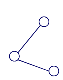


$N(h) \sim h^1$



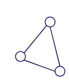
$N(h) \sim h^2$

Triangle 'Laws'



- Real social networks have a lot of triangles

Triangle 'Laws'



- Real social networks have a lot of triangles
  - Friends of friends are friends
- Any patterns?

CMU SCS

Triangles

- Naïve algo: 3-way join (slow)
- [Tsourakakis'08]: # triangles ~ sum of cubes of eigenvalues
- Thus, super-fast computation of #triangles (100x - 25,000x faster than naïve; >95% accuracy

CMU SCS

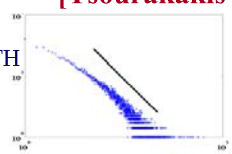
Triangles

- Easy to implement on ``hadoop'': it only needs eigenvalues (e.g., with Lanczos)

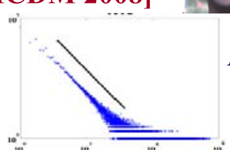
CMU SCS

Triangle Law: #1 [Tsourakakis ICDM 2008]

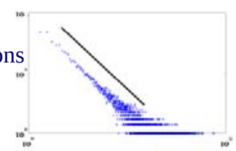
HEP-TH



ASN



Epinions

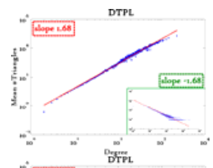


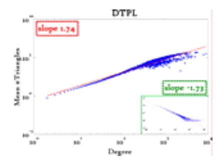
X-axis: # of Triangles a node participates in
Y-axis: count of such nodes

CMU SCS

Triangle Law: #2 [Tsourakakis ICDM 2008]

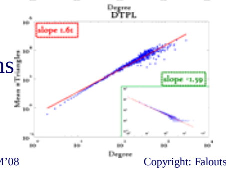
Reuters





SN

Epinions



X-axis: degree
Y-axis: mean # triangles
Notice: slope ~ degree exponent (insets)

CMU SCS

Triangle Law: Computations [Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos)
Q: Can we do that quickly?

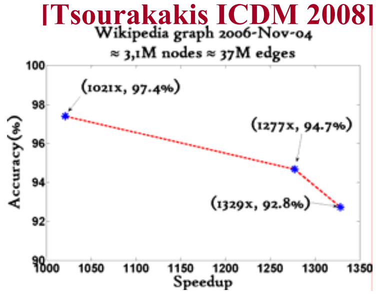
CMU SCS

Triangle Law: Computations [Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos)
Q: Can we do that quickly?
A: Yes!
#triangles = 1/6 Sum (λ_i^3)
(and, because of skewness, we only need the top few eigenvalues!

CMU SCS

Triangle Law: Computations [Tsourakakis ICDM 2008]



1000x+ speed-up, high accuracy

CIKM'08 1-37

CMU SCS

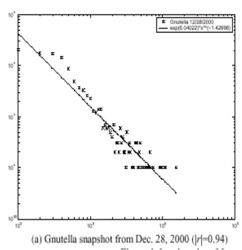
But:

- Q1: How about graphs from other domains?
- Q2: How about temporal evolution?

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-38

CMU SCS

The Peer-to-Peer Topology



[Jovanovic+]

- Frequency versus degree
- Number of adjacent peers follows a power-law

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-39

CMU SCS

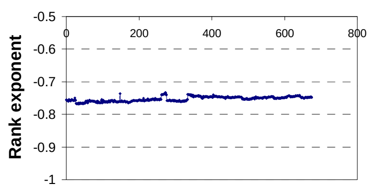
More Power laws

- Also hold for other web graphs [Barabasi+, '99], [Kumar+, '99] with additional 'rules' (bi-partite cores follow power laws)

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-40

CMU SCS

Time Evolution: rank *R*



Domain level

#days since Nov. '97

The rank exponent has not changed! [Siganos+, '03]

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-41

CMU SCS

Outline

- Topology & 'laws'
  - Static graphs
  - Evolving graphs
  - Weighted graphs
- Generators
- Discussion

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-42

CMU SCS

Any other 'laws'?

Yes!

CMU SCS

Any other 'laws'?

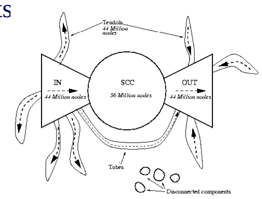
Yes!

- Small diameter (~ constant!) –
  - six degrees of separation / 'Kevin Bacon'
  - small worlds [Watts and Strogatz]

CMU SCS

Any other 'laws'?

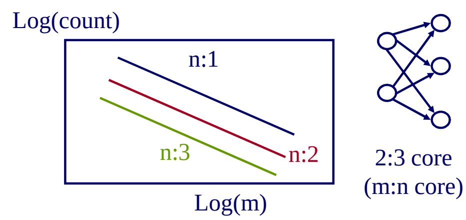
- Bow-tie, for the web [Kumar+ '99]
- IN, SCC, OUT, 'tendrils'
- disconnected components



CMU SCS

Any other 'laws'?

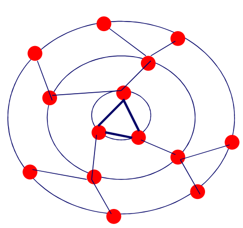
- power-laws in communities (bi-partite cores) [Kumar+, '99]



CMU SCS

Any other 'laws'?

- "Jellyfish" for Internet [Tauro+ '01]
- core: ~clique
- ~5 concentric layers
- many 1-degree nodes



CMU SCS

Summary of 'laws'

- Power laws for degree distributions
- ………….. for eigenvalues, bi-partite cores
- Triangle laws
- Small diameter ('6 degrees')
- 'Bow-tie' for web; 'jelly-fish' for internet

Outline

- Topology & 'laws'
  - Static graphs
  - ➡ Evolving graphs
  - Weighted graphs
- Generators
- Discussion

Evolution of diameter?

- Prior analysis, on power-law-like graphs, hints that

  diameter ~ O(log(N)) or

  diameter ~ O(log(log(N)))
- i.e.., slowly increasing with network size
- Q: What is happening, in reality?

Evolution of diameter?

- Prior analysis, on power-law-like graphs, hints that

  diameter ~ O(log(N)) or

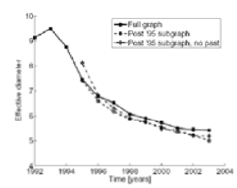
  diameter ~ O(log(log(N)))
- i.e.., slowly increasing with network size
- Q: What is happening, in reality?
- A: It **shrinks**(!!), towards a constant value

Shrinking diameter

[Leskovec+05a]

- Citations among physics papers
- 11yrs; @ 2003:
  - 29,555 papers
  - 352,807 citations
- For each month *M*, create a graph of all citations up to month *M*

diameter

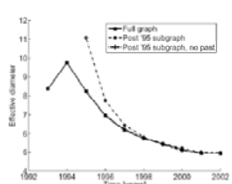


(a) arXiv citation graph

time

Shrinking diameter

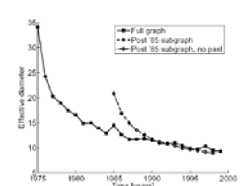
- Authors & publications
- 1992
  - 318 nodes
  - 272 edges
- 2002
  - 60,000 nodes
    - 20,000 authors
    - 38,000 papers
  - 133,000 edges



(b) Affiliation network

Shrinking diameter

- Patents & citations
- 1975
  - 334,000 nodes
  - 676,000 edges
- 1999
  - 2.9 million nodes
  - 16.5 million edges
- Each year is a datapoint



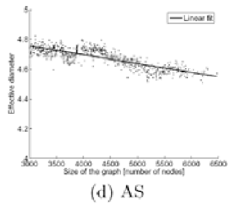
(c) Patents

CMU SCS

Shrinking diameter

- Autonomous systems
- 1997
  - 3,000 nodes
  - 10,000 edges
- 2000
  - 6,000 nodes
  - 26,000 edges
- One graph per day

diameter



N

CMU SCS

Temporal evolution of graphs

- N(t) nodes; E(t) edges at time t
- suppose that

  N(t+1) = 2 * N(t)
- Q: what is your guess for

  E(t+1) =? 2 * E(t)

CMU SCS

Temporal evolution of graphs

- N(t) nodes; E(t) edges at time t
- suppose that

  N(t+1) = 2 * N(t)
- Q: what is your guess for

  E(t+1) =? X * E(t)
- A: over-doubled!

CMU SCS

Temporal evolution of graphs

- A: over-doubled - but obeying:

  $E(t) \sim N(t)^a$ for all t

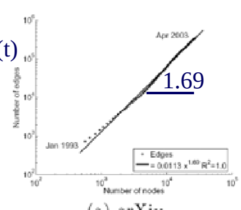
  where $1<a<2$

CMU SCS

Densification Power Law

ArXiv: Physics papers
and their citations

E(t)



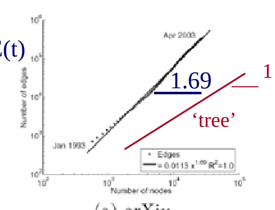
N(t)

CMU SCS

Densification Power Law

ArXiv: Physics papers
and their citations

E(t)



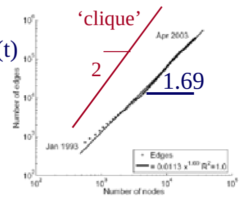
N(t)

CMU SCS

Densification Power Law

ArXiv: Physics papers and their citations

E(t)



(a) arXiv

N(t)

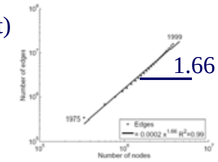
CIKM'08 Copyright: Faloutsos, Tong (2008) 1-61

CMU SCS

Densification Power Law

U.S. Patents, citing each other

E(t)



(b) Patents

N(t)

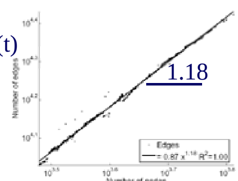
CIKM'08 Copyright: Faloutsos, Tong (2008) 1-62

CMU SCS

Densification Power Law

Autonomous Systems

E(t)



(c) Autonomous Systems

N(t)

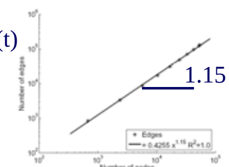
CIKM'08 Copyright: Faloutsos, Tong (2008) 1-63

CMU SCS

Densification Power Law

ArXiv: authors & papers

E(t)



(d) Affiliation network

N(t)

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-64

CMU SCS

Outline

- Topology & 'laws'
  - Static graphs
  - ➡ Evolving graphs
  - Weighted graphs
- Generators
- Discussion

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-65

CMU SCS

More on Time-evolving graphs


M. McGlohon, L. Akoglu, and C. Faloutsos
Weighted Graphs and Disconnected Components: Patterns and a Generator. SIG-KDD 2008

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-66

CMU SCS

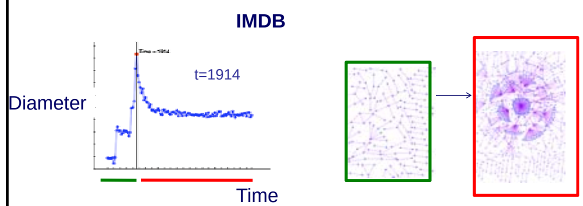
Observation 1: Gelling Point

Q1: How does the GCC emerge?

CMU SCS

Observation 1: Gelling Point

- Most real graphs display a gelling point
- After gelling point, they exhibit typical behavior. This is marked by a spike in diameter.



CMU SCS

Observation 2: NLCC behavior

Q2: How do NLCC's emerge and join with the GCC?

(``NLCC'' = non-largest conn. components)

–Do they continue to grow in size?

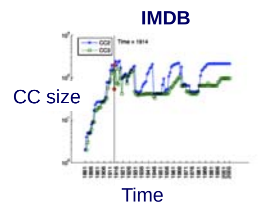
– or do they shrink?

– or stabilize?

CMU SCS

Observation 2: NLCC behavior

- After the gelling point, the GCC takes off, but NLCC's remain ~constant (actually, oscillate).



CMU SCS

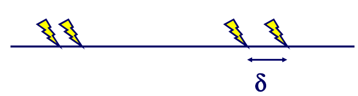
How do new edges appear?

[LBKT'08] Microscopic Evolution of Social Networks
Jure Leskovec, Lars Backstrom, Ravi Kumar, Andrew Tomkins.
(ACM KDD), 2008.

CMU SCS

How do edges appear in time? [LBKT'08]

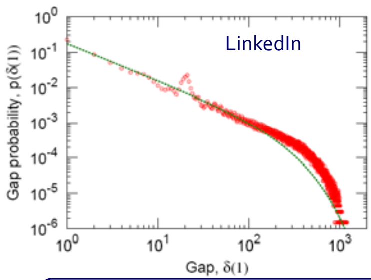
Edge gap δ(d): inter-arrival time between d^{th} and $d+1^{st}$ edge



What is the PDF of δ? Poisson**?**

CMU SCS

How do edges appear in time? [LBKT'08]

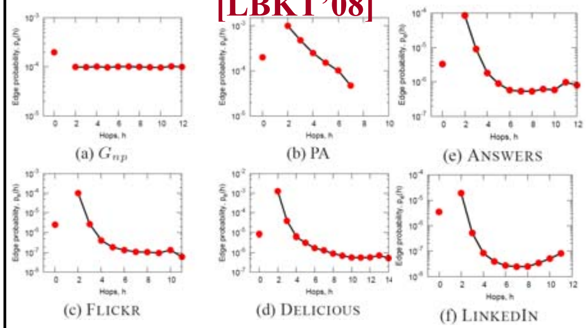


Edge gap δ(d): inter-arrival time between *d^{th}* and *d+1^{st}* edge

$$p_g(\delta(d);\alpha,\beta) \propto \delta(d)^{-\alpha} e^{-\beta\delta(d)}$$

CMU SCS

Edge locality – closing triangles [LBKT'08]



CMU SCS

Outline

- Topology & 'laws'
  - Static graphs
  - Evolving graphs
  - Weighted graphs
- Generators
- Discussion

 Copyright: Faloutsos, Tong (2008)

CMU SCS

Weighted graphs

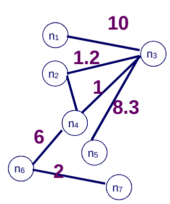
M. McGlohon, L. Akoglu, and C. Faloutsos. *Weighted Graphs and Disconnected Components: Patterns and a Generator.* KDD 2008

 Copyright: Faloutsos, Tong (2008)

CMU SCS

Networks used

- **Postnet**: Posts in blogs, hyperlinks between
- **Blognet**: Aggregated Postnet, repeated edges
- **Patent:** Patent citations
- **NIPS**: Academic citations
- **Arxiv**: Academic citations
- **NetTraffic**: Packets, repeated edges
- Autonomous Systems (**AS**): Packets, repeated edges

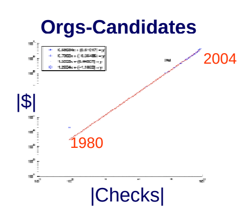


 Copyright: Faloutsos, Tong (2008)

CMU SCS

Observation 1: Fortification Effect

- $ = C* # checks ?



 Copyright: Faloutsos, Tong (2008)

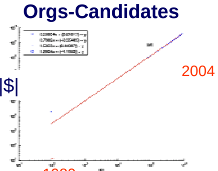
CMU SCS

Observation 1: Fortification Effect – Weight P.L. (WPL)

- Weight additions follow a power law with respect to the number of edges:

$$W(t) \propto E(t)^{w}$$

  - *W(t):* total weight of graph at *t*
  - *E(t)*: total edges of graph at *t*
  - *w* is PL exponent
  - 1.01 < w < 1.5 = super-linear!
  - (more checks, even more $)



CMU SCS

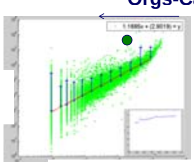
Observation 2

Q2: How do the weights of nodes relate to degree?

CMU SCS

Observation 2: Snapshot Power Law

- At any time, total incoming weight of a node is proportional to in-degree with PL exponent 'iw':
  - *i.e. 1.01 < iw < 1.26, super-linear*
- *More donors, even more $*



CMU SCS

Summary of 'laws'

- Power laws for degree distributions
- ………….. for eigenvalues, bi-partite cores
- Triangle laws
- Small diameter ('6 degrees')
- 'Bow-tie' for web; 'jelly-fish' for internet
- Oscillating NLCC's
- WPL (weight grows superlinearly to degree)
- ~Power-law between edge generations

CMU SCS

Outline

- Topology & 'laws'
- ➡ Generators
  - Erdos – Renyi
  - Degree-based
  - Process-based
  - Recursive generators
- Discussion

CMU SCS

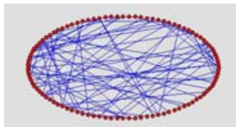
Generators

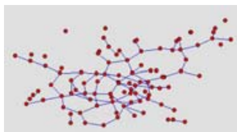
- How to generate random, realistic graphs?
  - Erdos-Renyi model: beautiful, but unrealistic
  - degree-based generators
  - process-based generators
  - recursive/self-similar generators

CMU SCS

Erdos-Renyi

- random graph – 100 nodes, avg degree = 2
- Fascinating properties (phase transition)
- But: unrealistic (Poisson degree distribution != power law)



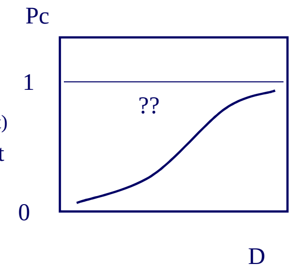


CIKM'08 Copyright: Faloutsos, Tong (2008) 1-85

CMU SCS

E-R model & Phase transition

- vary avg degree D
- watch Pc =
  Prob(there is a giant connected component)
- How do you expect it to be?

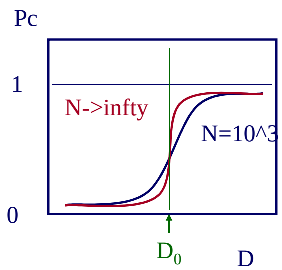


CIKM'08 Copyright: Faloutsos, Tong (2008) 1-86

CMU SCS

E-R model & Phase transition

- vary avg degree D
- watch Pc =
  Prob(there is a giant connected component)
- How do you expect it to be?



CIKM'08 Copyright: Faloutsos, Tong (2008) 1-87

CMU SCS

Degree-based

- Figure out the degree distribution (eg., 'Zipf')
- Assign degrees to nodes
- Put edges, so that they match the original degree distribution

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-88

CMU SCS

Process-based

- Barabasi; Barabasi-Albert: Preferential attachment -> power-law tails!
  - 'rich get richer'
- [Kumar+]: preferential attachment + mimick
  - Create 'communities'

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-89

CMU SCS

Process-based (cont'd)

- [Fabrikant+, '02]: H.O.T.: connect to closest, high connectivity neighbor
- [Pennock+, '02]: Winner does NOT take all

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-90

CMU SCS

Outline

- Topology & ‘laws’
- Generators
  - Erdos – Renyi
  - Degree-based
  - Process-based
  - Recursive generators
- Discussion and tools

CMU SCS

Recursive generators

- (RMAT [Chakrabarti+,’04])
- Kronecker product

CMU SCS

Problem Definition

- Given a growing graph with count of nodes N_1, N_2, …
- Generate a realistic sequence of graphs that will obey all the patterns
- **Idea: Self-similarity**
  - Leads to power laws
  - Communities within communities
  - …

CMU SCS

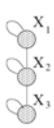
Wish list for a generator:

- Power-law-tail in- and out-degrees
- Power-law-tail scree plots
- **shrinking/constant** diameter
- Densification Power Law
- communities-within-communities

Q: how to achieve all of them?

CMU SCS

Kronecker Product – a Graph



1	1	0
1	1	1
0	1	1

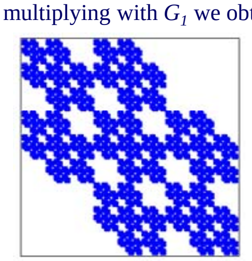
G_1

Adjacency matrix

CMU SCS

Kronecker Product – a Graph

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

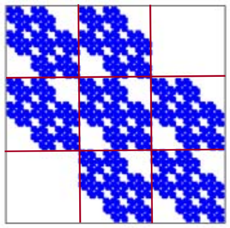


G_4 adjacency matrix

CMU SCS

Kronecker Product – a Graph

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



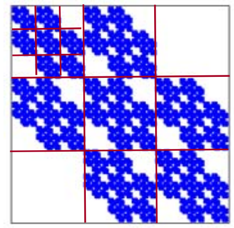
G_4 adjacency matrix

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-97

CMU SCS

Kronecker Product – a Graph

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



G_4 adjacency matrix

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-98

CMU SCS

Properties:

- We can PROVE that
  - Degree distribution is multinomial ~ power law
  - Diameter: constant
  - Eigenvalue distribution: multinomial
  - First eigenvector: multinomial
- See [Leskovec+, PKDD'05] for proofs

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-99

CMU SCS

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 and only generator for which we can **prove** all these properties

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-100

CMU SCS

Experiments

- How well can we match real graphs?
  - Arxiv: physics citations:
    - 30,000 papers, 350,000 citations
    - 10 years of data
  - U.S. Patent citation network
    - 4 million patents, 16 million citations
    - 37 years of data
  - Autonomous systems – graph of internet
    - Single snapshot from January 2002
    - 6,400 nodes, 26,000 edges
- We show both static and temporal patterns

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-101

CMU SCS

skip

Stochastic Kronecker Graphs

- Create $N_1 \times N_1$ probability matrix P_1
- Compute the k^{th} Kronecker power P_k
- For each entry p_{uv} of P_k include an edge (u,v) with probability p_{uv}

0.4	0.2
0.1	0.3

P_1

Kronecker multiplication →

0.16	0.08	0.08	0.04
0.04	0.12	0.02	0.06
0.04	0.02	0.12	0.06
0.01	0.03	0.03	0.09

P_k

→ flip biased coins → Instance Matrix G_2

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-102

CMU SCS

(Q: how to fit the parm's?)

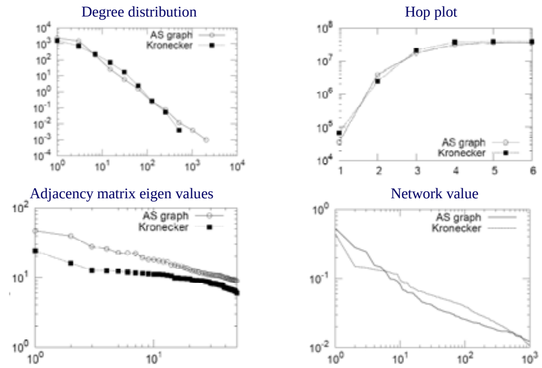
A:

- Stochastic version of Kronecker graphs +
- Max likelihood +
- Metropolis sampling
- [Leskovec+, ICML'07]

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-103

CMU SCS

Experiments on real AS graph



CMU SCS

Conclusions

- Kronecker graphs have:
  - All the static properties
    - ✓Heavy tailed degree distributions
    - ✓Small diameter
    - ✓Multinomial eigenvalues and eigenvectors
  - All the temporal properties
    - ✓Densification Power Law
    - ✓Shrinking/Stabilizing Diameters
  - We can formally prove these results

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-105

CMU SCS

Outline

- Topology & 'laws'
- Generators
  - Erdos – Renyi
  - Degree-based
  - ➡ Process-based
  - Recursive generators
- Discussion and tools

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-106

CMU SCS

Recent, process-based generators

- Triad closing [LBKT'08]
- 'butterfly' model [McGlohon+, KDD'08]

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-107

CMU SCS

Goals of model

a) Emergent, intuitive behavior

b) Shrinking diameter

c) Constant NLCC's

d) Densification power law

e) Power-law degree distribution

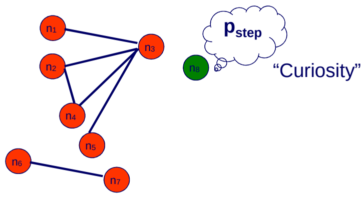
= "Butterfly" Model

CIKM'08 Copyright: Faloutsos, Tong (2008) 1-108

CMU SCS

Butterfly model in action

- A node joins a network, with own parameter.

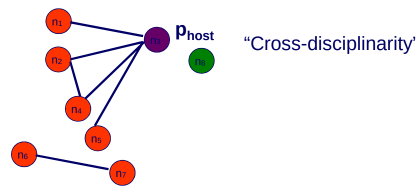


CIKM'08 Copyright: Faloutsos, Tong (2008) 1-109

CMU SCS

Butterfly model in action

- A node joins a network, with own parameter.
- With (global) p_{host}, chooses a random host

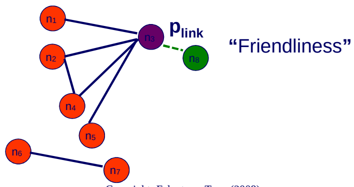


CIKM'08 Copyright: Faloutsos, Tong (2008) 1-110

CMU SCS

Butterfly model in action

- A node joins a network, with own parameters.
- With (global) p_{host}, chooses a random host
  - With (global) p_{link}, creates link

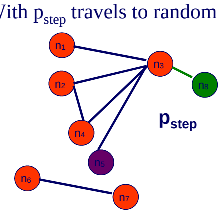


CIKM'08 Copyright: Faloutsos, Tong (2008) 1-111

CMU SCS

Butterfly model in action

- A node joins a network, with own parameters.
- With (global) p_{host}, chooses a random host
  - With (global) p_{link}, creates link
  - With p_{step} travels to random neighbor

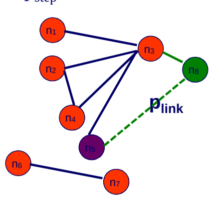


CIKM'08 1-112

CMU SCS

Butterfly model in action

- A node joins a network, with own parameters.
- With (global) p_{host}, chooses a random host
  - With (global) p_{link}, creates link
  - With p_{step} travels to random neighbor. Repeat.

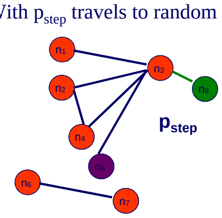


CIKM'08 1-113

CMU SCS

Butterfly model in action

- A node joins a network, with own parameters.
- With (global) p_{host}, chooses a random host
  - With (global) p_{link}, creates link
  - With p_{step} travels to random neighbor. Repeat.

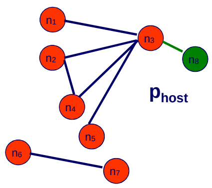


CIKM'08 1-114

CMU SCS

Butterfly model in action

- Once there are no more “steps”, repeat “host” procedure:
  - With p_{host}, choose new host, possibly link, etc.

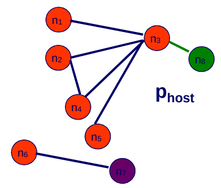


CIKM’08 1-115

CMU SCS

Butterfly model in action

- Once there are no more “steps”, repeat “host” procedure:
  - With p_{host}, choose new host, possibly link, etc.

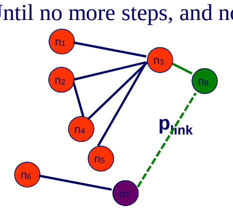


CIKM’08 1-116

CMU SCS

Butterfly model in action

- Once there are no more “steps”, repeat “host” procedure:
  - With p_{host}, choose new host, possibly link, etc.
  - Until no more steps, and no more hosts.

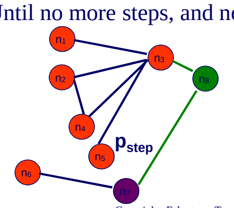


CIKM’08 1-117

CMU SCS

Butterfly model in action

- Once there are no more “steps”, repeat “host” procedure:
  - With p_{host}, choose new host, possibly link, etc.
  - Until no more steps, and no more hosts.



CIKM’08 Copyright: Faloutsos, Tong (2008) 1-118

CMU SCS

Emergent, intuitive behavior

Novelties of model:

- Nodes link with probability p_{link}
  - May choose host, but not link (start new component)
- Incoming nodes are “social butterflies”
  - May have several hosts (merges components)
- Some nodes are friendlier than others
  - p_{step} different for each node
  - This creates power-law degree distribution (Theorem)

CIKM’08 Copyright: Faloutsos, Tong (2008) 1-119

CMU SCS

Outline

- Topology & ‘laws’
- Generators
- ➡ Discussion

CIKM’08 Copyright: Faloutsos, Tong (2008) 1-120

CMU SCS

Power laws

- Q1: Why so many?
- A1:

CMU SCS

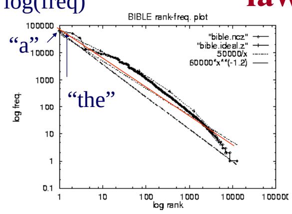
Power laws

- Q1: Why so many?
- A1: self-similarity; 'rich-get-richer', etc - see Newman's paper
  http://arxiv.org/abs/cond-mat/0412004v3

Other settings with power laws?

CMU SCS

A famous power law: Zipf's law



- Bible - **rank** vs **frequency** (log-log)

CMU SCS

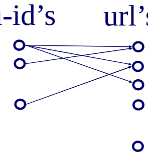
Power laws, cont'ed

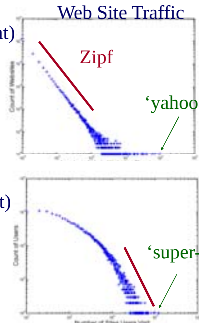
- web hit counts [Huberman]
- Click-stream data [Montgomery+01]

CMU SCS

Click-stream data

u-id's url's



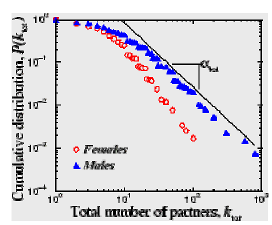


CMU SCS

Swedish sex-web

Nodes: people (Females; Males)
Links: sexual relationships

Albert Laszlo Barabasi
http://www.nd.edu/~networks/Publication%20Categories/04%20Talks/2005-norway-3hours.ppt



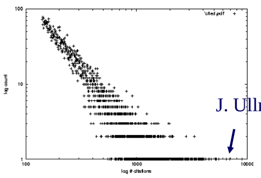
4781 Swedes; 18-74;
59% response rate.

Liljeros et al. *Nature* 2001

CMU SCS

Lotka's law

(Lotka's law of publication count); and citation counts: (*citeseer.nj.nec.com* 6/2001)



CMU SCS

Conclusions - Laws

'Laws' and patterns:

- Power laws for degrees, eigenvalues, 'communities'/cores
- Small diameter and shrinking diameter
- Bow-tie; jelly-fish
- densification power law

CMU SCS

Conclusions - Generators

- Preferential attachment (Barabasi)
- Variations + extensions:
  - copying model
  - triad-closing
  - Butterfly model
- Recursion – Kronecker graphs

CMU SCS

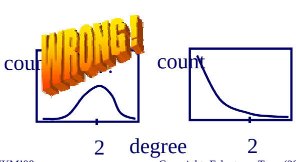
Conclusions - Tools

- Power laws –
  - rank/frequency plots ~ log-log NCDF
  - log-log PDF
- Self-similarity / recursion / fractals

CMU SCS

Conclusions - Tools (cont'd)

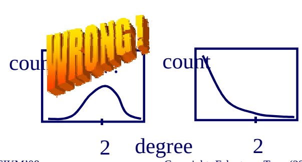
- Real settings/graphs: skewed distributions
  - 'mean' is meaningless
  - slope of power law, instead

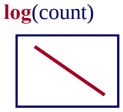


CMU SCS

Conclusions - Tools (cont'd)

- Real settings/graphs: skewed distributions
  - 'mean' is meaningless
  - slope of power law, instead

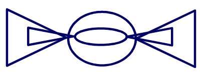




CMU SCS

Conclusions - Tools (cont'd)

- Recursion/self-similarity
  - May reveal non-obvious patterns (e.g., bow-ties within bow-ties within bow-ties) [Dill+, '01]



"To iterate is human, to recurse is divine"

CMU SCS

Resources

Generators:
- Kronecker (christos@cs.cmu.edu)
- BRITE http://www.cs.bu.edu/brite/
- INET: http://topology.eecs.umich.edu/inet

CMU SCS

Other resources

Visualization - graph algo's:
- Graphviz: http://www.graphviz.org/
- pajek: http://vlado.fmf.uni-lj.si/pub/networks/pajek/

Kevin Bacon web site:
http://www.cs.virginia.edu/oracle/

CMU SCS

References

- [Aiello+, '00] William Aiello, Fan R. K. Chung, Linyuan Lu: *A random graph model for massive graphs*. STOC 2000: 171-180
- [Albert+] Reka Albert, Hawoong Jeong, and Albert-Laszlo Barabasi: *Diameter of the World Wide Web*, Nature 401 130-131 (1999)
- Réka Albert and Albert-László Barabási Statistical mechanics of complex networks, Reviews of Modern Physics, 74, 47 (2002).
- [Barabasi, '03] Albert-Laszlo Barabasi *Linked: How Everything Is Connected to Everything Else and What It Means* (Plume, 2003)

CMU SCS

References, cont'd

- [Barabasi+, '99] Albert-Laszlo Barabasi and Reka Albert. *Emergence of scaling in random networks.* Science, 286:509--512, 1999
- [Broder+, '00] Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. *Graph structure in the web*, WWW, 2000

CMU SCS

References, cont'd

- [Chakrabarti+, '04] *RMAT: A recursive graph generator*, D. Chakrabarti, Y. Zhan, C. Faloutsos, SIAM-DM 2004
- D. Chakrabarti and C. Faloutsos, *Graph Mining: Laws, Generators and Algorithms*, in ACM Computing Surveys, 38(1), 2006
- [Dill+, '01] Stephen Dill, Ravi Kumar, Kevin S. McCurley, Sridhar Rajagopalan, D. Sivakumar, Andrew Tomkins: *Self-similarity in the Web*. VLDB 2001: 69-78

CMU SCS

References, cont'd

- [Fabrikant+, '02] A. Fabrikant, E. Koutsoupias, and C.H. Papadimitriou. *Heuristically Optimized Trade-offs: A New Paradigm for Power Laws in the Internet*. ICALP, Malaga, Spain, July 2002
- [FFF, 99] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "*On power-law relationships of the Internet topology*," in SIGCOMM, 1999.

CMU SCS

References, cont'd

- [Jovanovic+, '01] M. Jovanovic, F.S. Annexstein, and K.A. Berman. *Modeling Peer-to-Peer Network Topologies through "Small-World" Models and Power Laws*. In TELFOR, Belgrade, Yugoslavia, November, 2001
- [Kumar+ '99] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Andrew Tomkins: *Extracting Large-Scale Knowledge Bases from the Web*. VLDB 1999: 639-650

CMU SCS

References, cont'd

- [Leland+, '94] W. E. Leland, M.S. Taqqu, W. Willinger, D.V. Wilson, *On the Self-Similar Nature of Ethernet Traffic*, IEEE Transactions on Networking, 2, 1, pp 1-15, Feb. 1994.
- [Leskovec+05a] Jure Leskovec, Jon Kleinberg, Christos Faloutsos, *Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations* (KDD 2005)

CMU SCS

References, cont'd

- [Leskovec+05b] Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos *Realistic, Mathematically Tractable Graph Generation and Evolution, Using Kronecker Multiplication* (ECML/PKDD 2005), Porto, Portugal, 2005.

CMU SCS

References, cont'd

- [Leskovec+07] Jure Leskovec and Christos Faloutsos, Scalable Modeling of Real Graphs using Kronecker Multiplication, ICML 2007.
- [Leskovec+ 08] Jure Leskovec, Lars Backstrom, Ravi Kumar, Andrew Tomkins. *Microscopic Evolution of Social Networks*, KDD, 2008.
- [McGlohon+ '08] M. McGlohon, L. Akoglu, and C. Faloutsos. *Weighted Graphs and Disconnected Components*: Patterns and a Generator. KDD 2008

CMU SCS

References, cont'd

- [Mihail+, '02] Milena Mihail, Christos H. Papadimitriou: *On the Eigenvalue Power Law*. RANDOM 2002: 254-262
- [Milgram '67] Stanley Milgram: *The Small World Problem*, Psychology Today 1(1), 60-67 (1967)
- [Montgomery+, '01] Alan L. Montgomery, Christos Faloutsos: *Identifying Web Browsing Trends and Patterns*. IEEE Computer 34(7): 94-95 (2001)

CMU SCS

References, cont'd

- [Palmer+, '01] Chris Palmer, Georgos Siganos, Michalis Faloutsos, Christos Faloutsos and Phil Gibbons *The connectivity and fault-tolerance of the Internet topology* (NRDM 2001), Santa Barbara, CA, May 25, 2001
- [Pennock+, '02] David M. Pennock, Gary William Flake, Steve Lawrence, Eric J. Glover, C. Lee Giles: *Winners don't take all: Characterizing the competition for links on the web* Proc. Natl. Acad. Sci. USA 99(8): 5207-5211 (2002)

CMU SCS

References, cont'd

- [Schroeder, '91] Manfred Schroeder *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise* W H Freeman & Co., 1991

CMU SCS

References, cont'd

- [Siganos+, '03] G. Siganos, M. Faloutsos, P. Faloutsos, C. Faloutsos *Power-Laws and the AS-level Internet Topology*, Transactions on Networking, August 2003.
- [Watts+ Strogatz, '98] D. J. Watts and S. H. Strogatz *Collective dynamics of 'small-world' networks*, Nature, 393:440-442 (1998)
- [Watts, '03] Duncan J. Watts *Six Degrees: The Science of a Connected Age* W.W. Norton & Company; (February 2003)