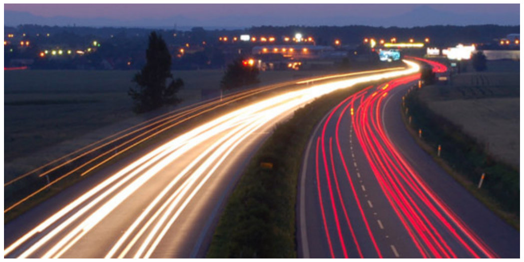
Carnegie Mellon

# Large Graph Mining - Patterns, Explanations and Cascade Analysis

*Christos Faloutsos*

CMU

# Roadmap



- ➡ A case for cross-disciplinarity
- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns in graphs
- Part#2: Cascade analysis
- Conclusions

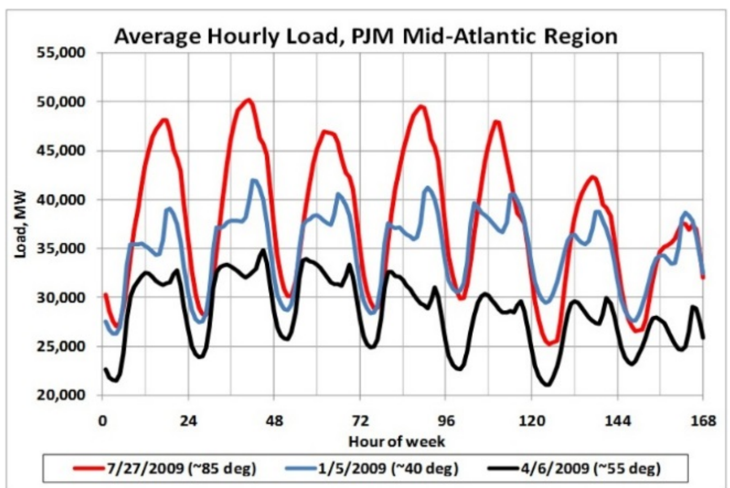
# **Speed up** of Data-Driven State Estimation Using Low-Complexity Indexing Method

## Data-Driven State Estimation

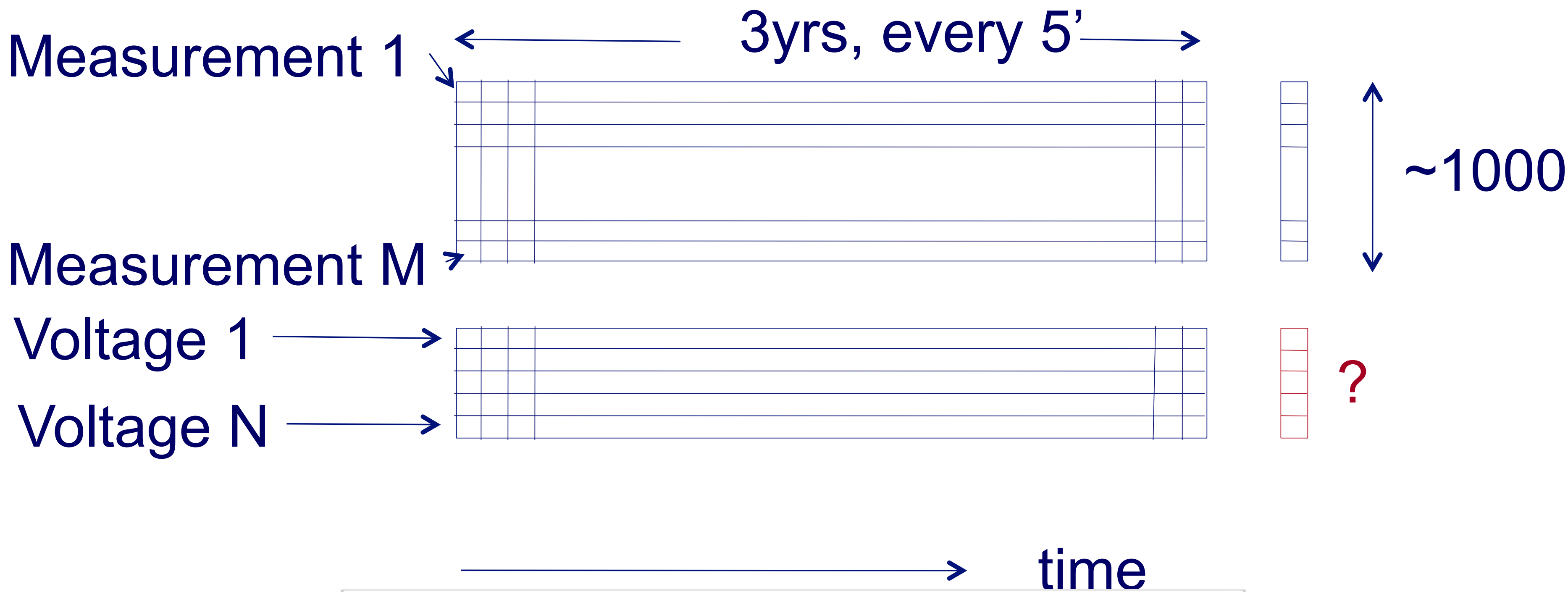
- Historical Similar Data Measurements, States ⇐ Time Consuming

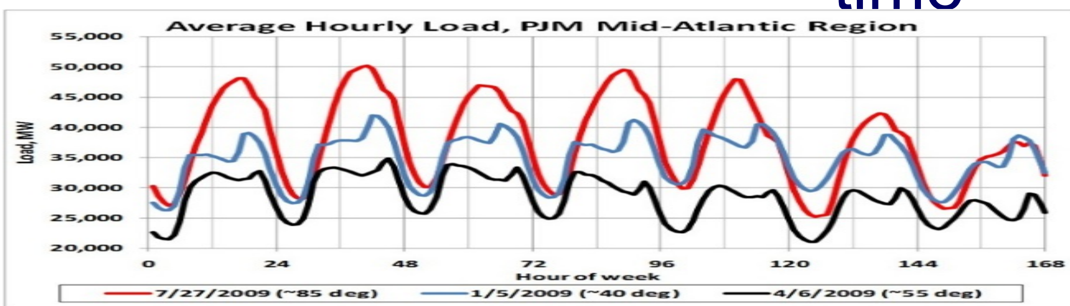
Observation:

- Redundancies & correlations

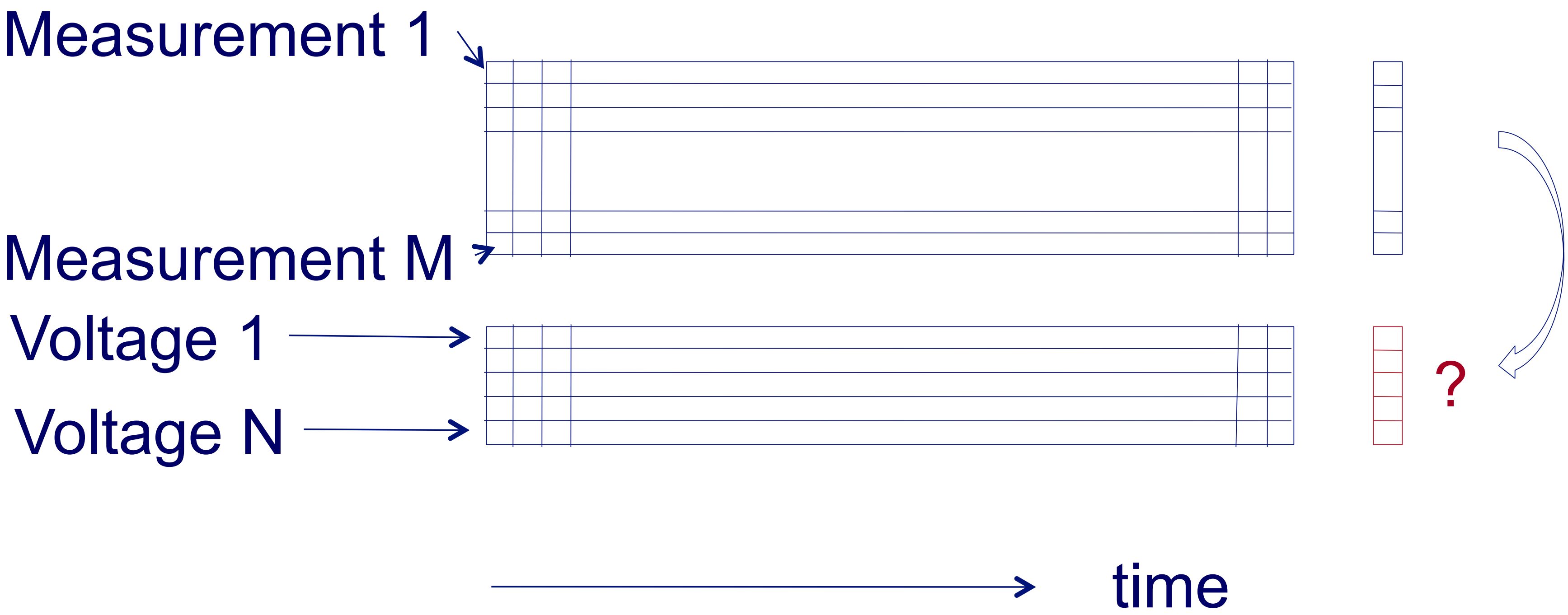


# Problem dfn


Measurement 1
3yrs, every 5'
~1000
Measurement M
Voltage 1
Voltage N
?
time

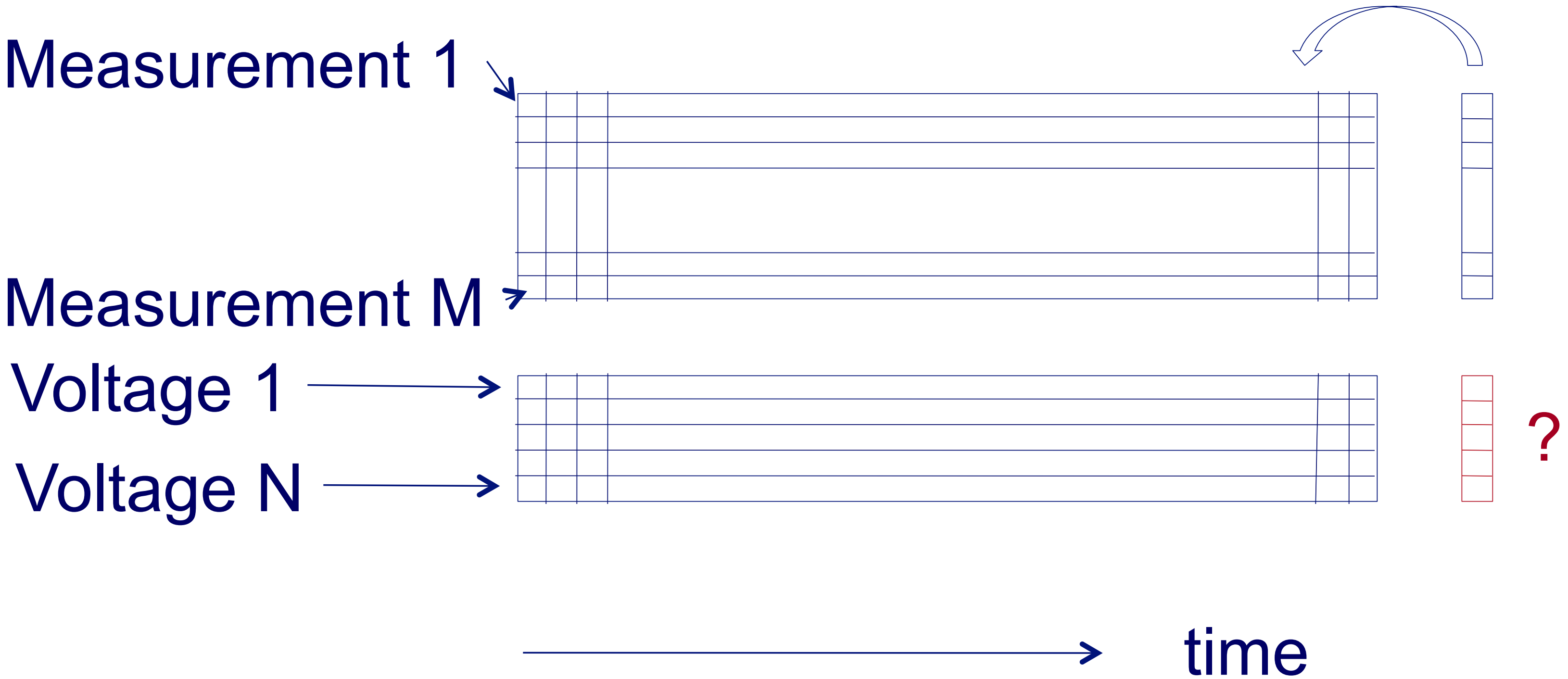

Average Hourly Load, PJM Mid-Atlantic Region
Load, MW
55,000
50,000
45,000
40,000
35,000
30,000
25,000
20,000
0
24
48
72
96
120
144
168
Hour of week
7/27/2009 (~85 deg)
1/5/2009 (~40 deg)
4/6/2009 (~55 deg)

# Problem dfn



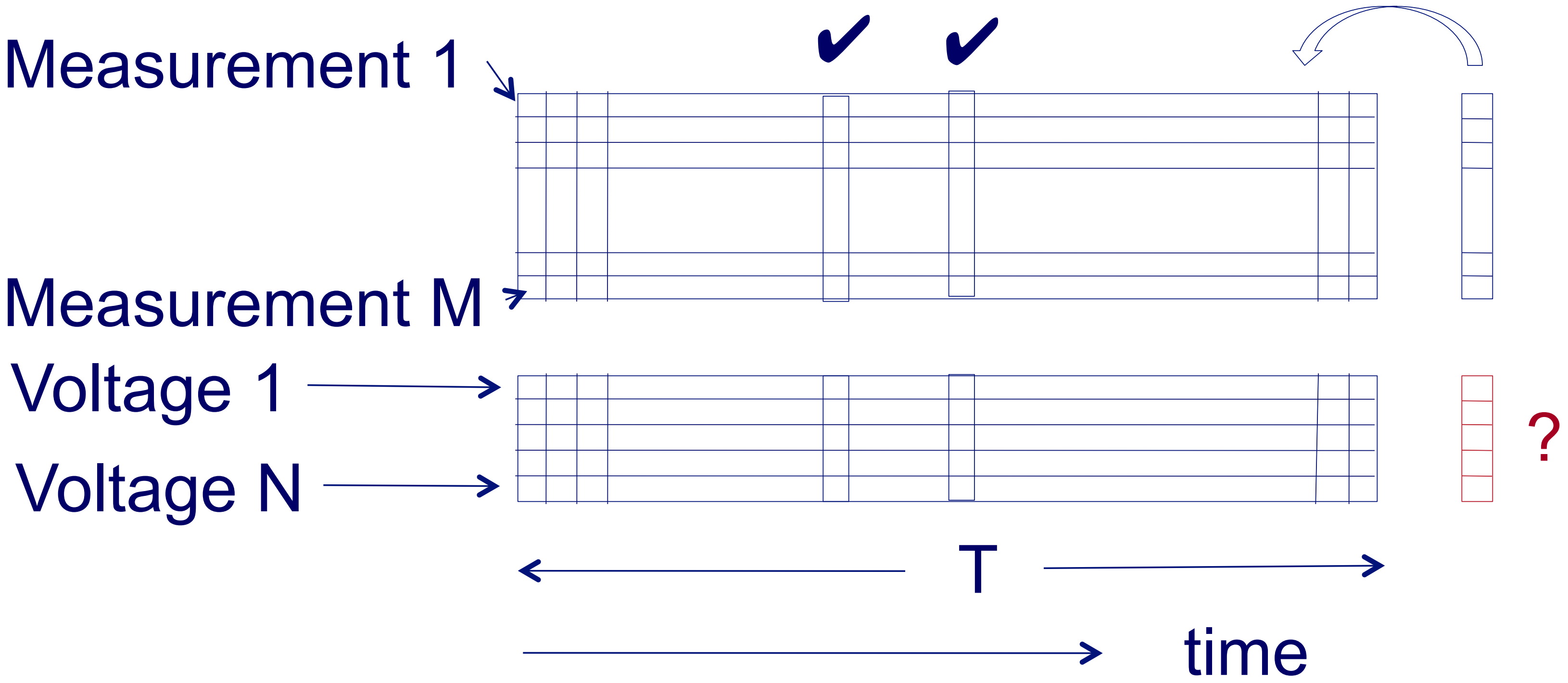

Direct solution:
Slow (Kirchoff's eq.)

# Problem dfn




Look for near-neighbors
And use *their* voltages

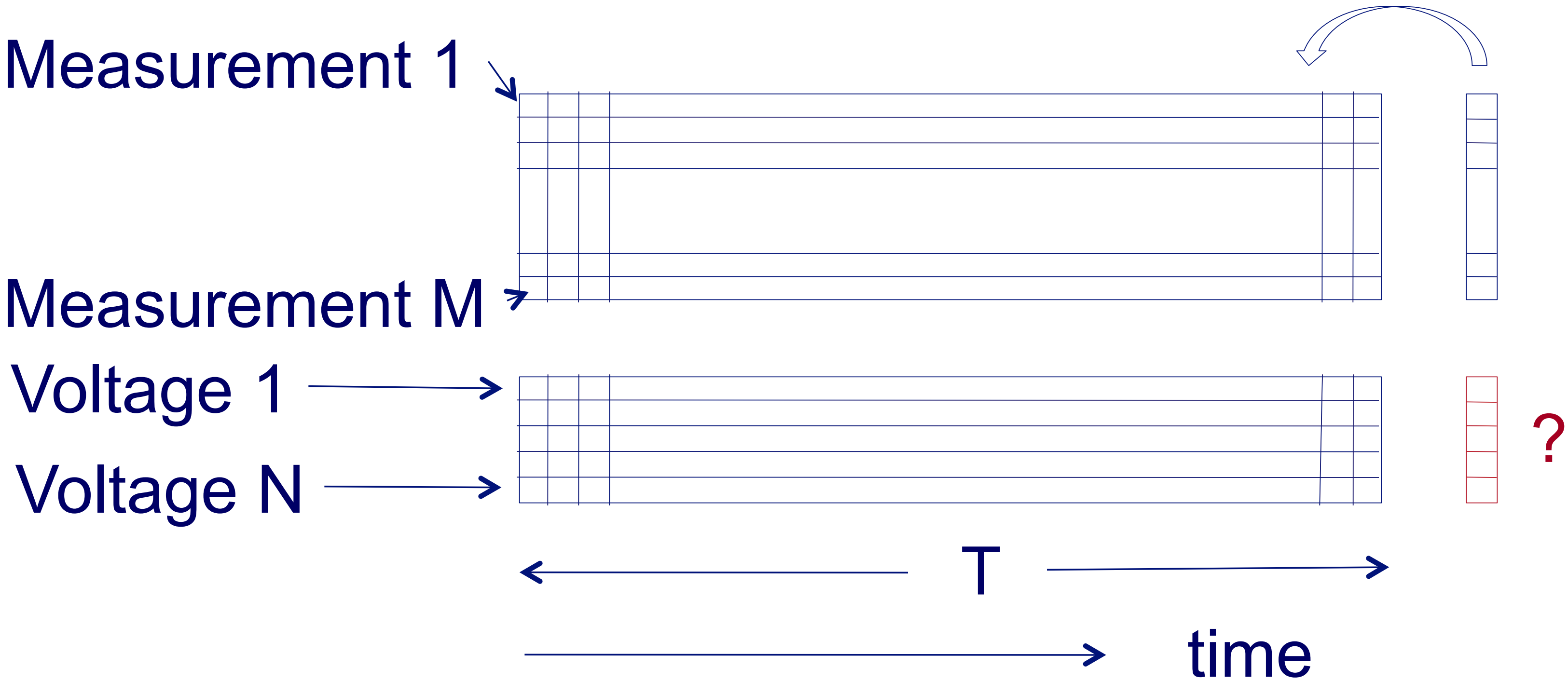
# Problem dfn




But sequential scan
Is slow, too (MxT)
Can we do better?

Look for near-neighbors
And use *their* voltages

# Problem dfn




But sequential scan
Is slow, too (MxT)
Can we do better?

A: yes!
We can reduce both
- T, and
- M

Carnegie Mellon

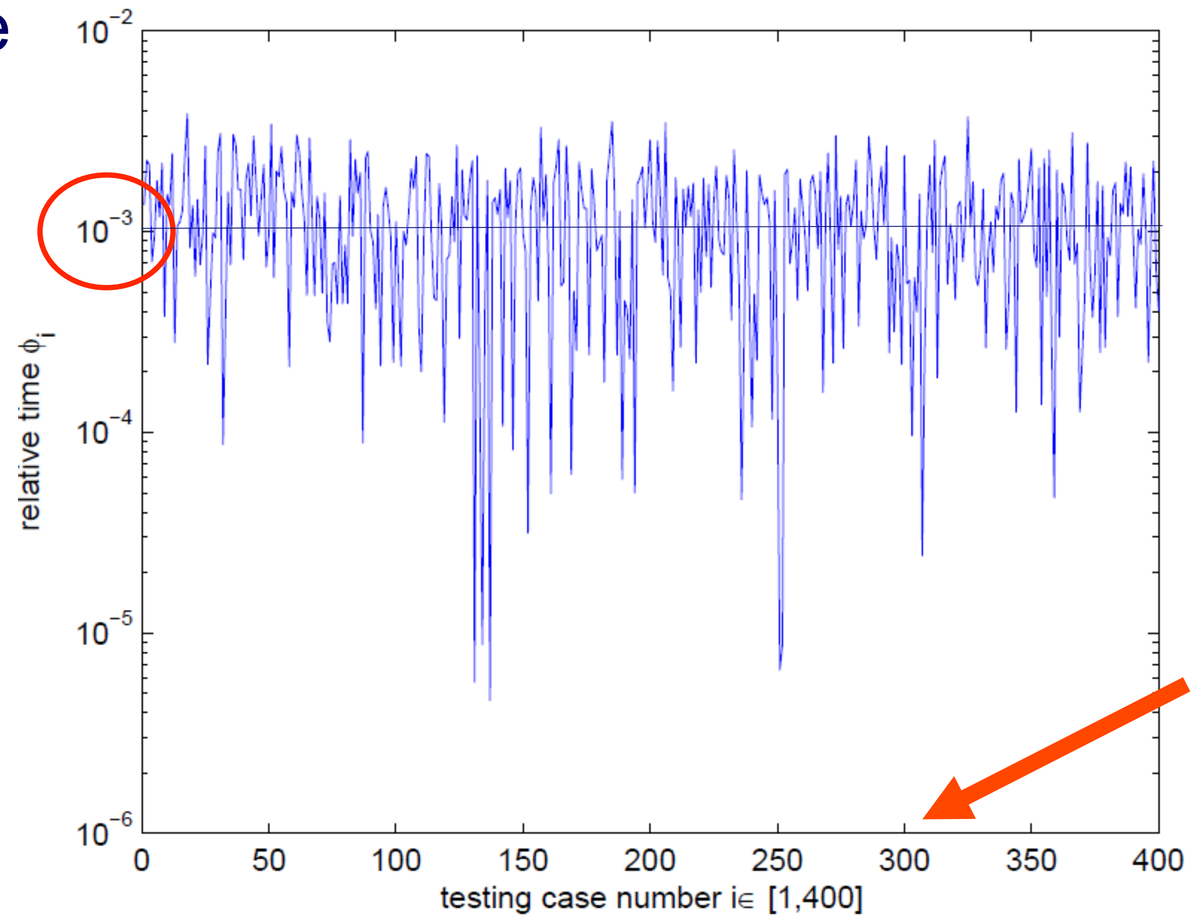
# Simulation Results

- Same accuracy, **100x – 100K x faster**

Relative Search Time:

**1000 x**

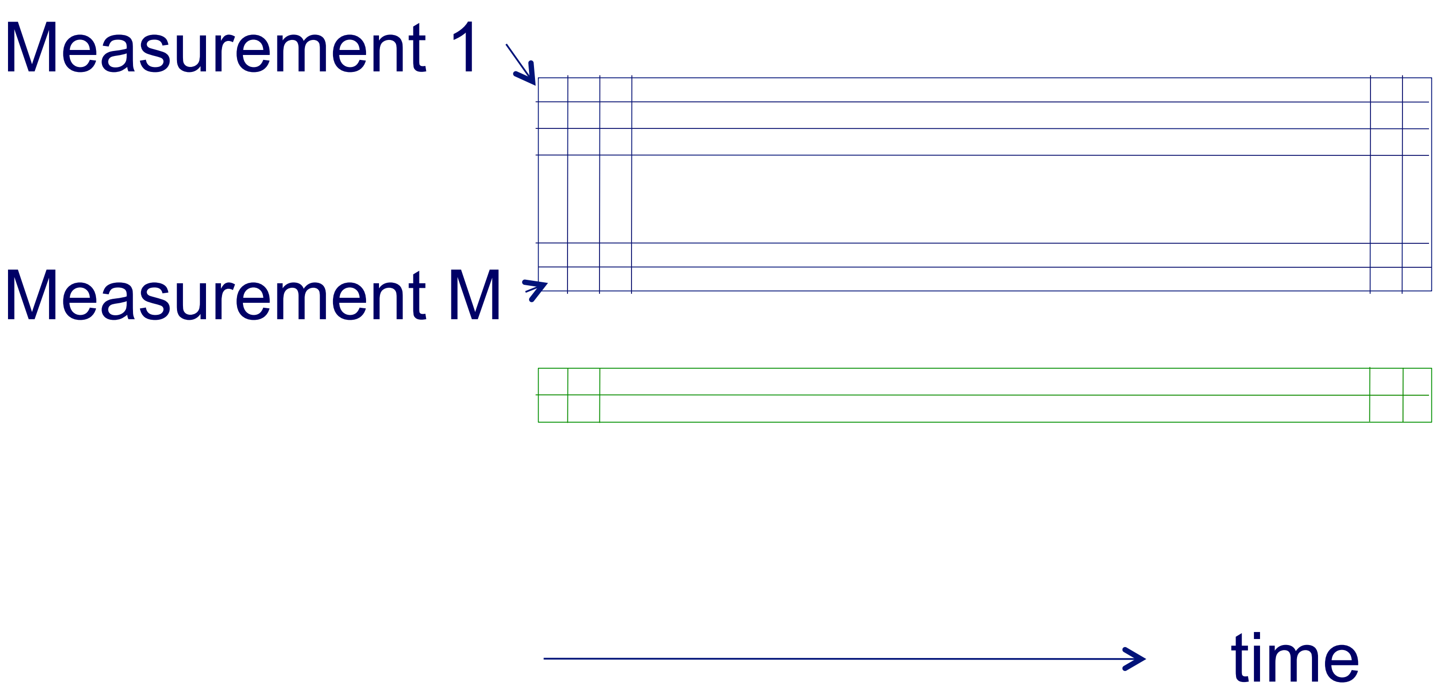
1 sec
vs 15’
vs 1 day

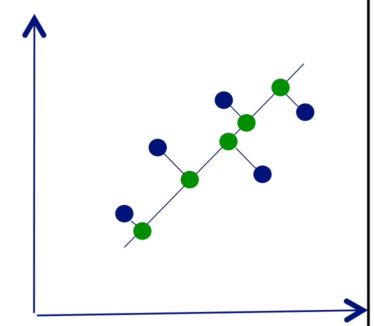



Many simulations


[1] Yang Weng, Christos Faloutsos, Marija D. Ili´c, and Rohit Negi, Speed up of Data-Driven State Estimation Using Low-Complexity Indexing Method, IEEE PES-General Meeting, (accepted), 2014

# Step1: Reducing dimensionality M



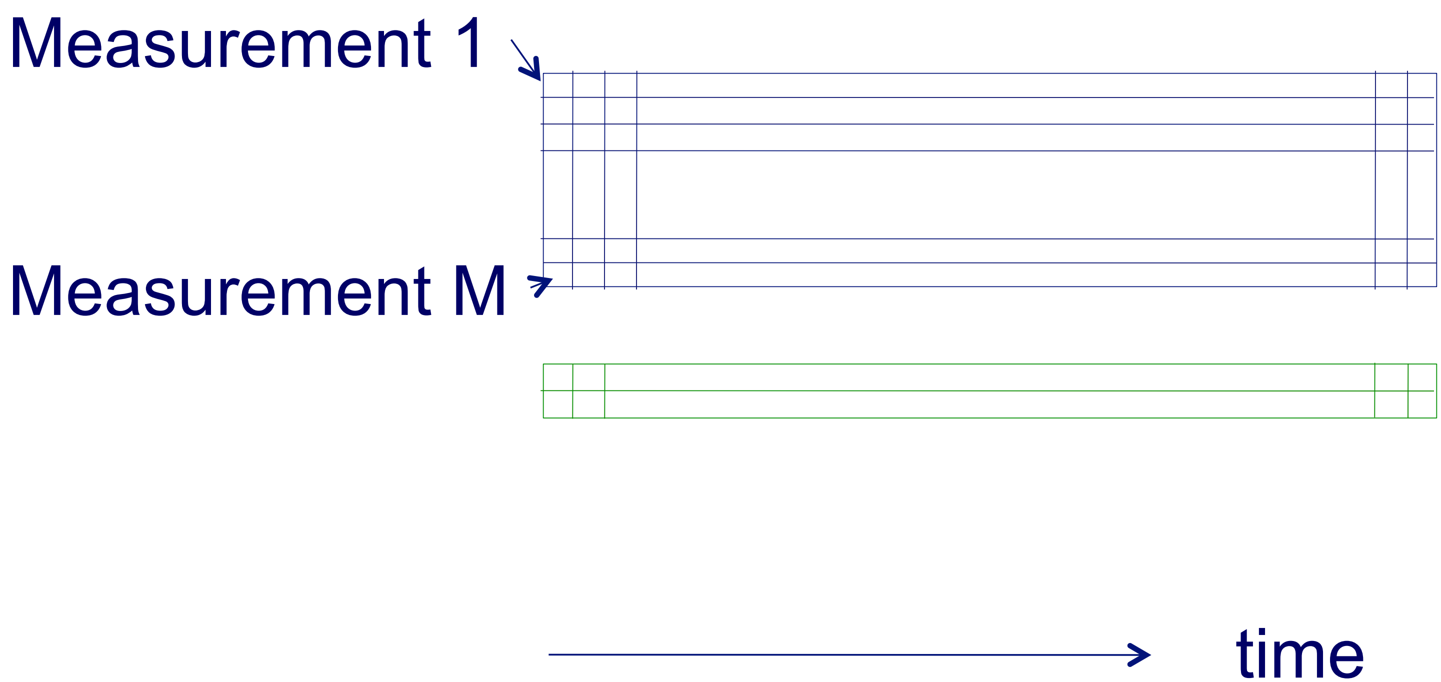



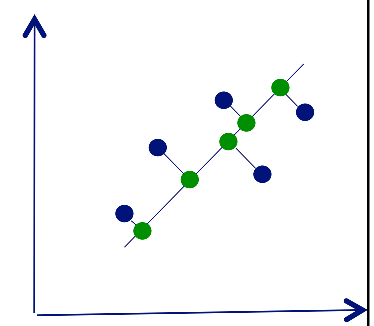
But sequential scan
Is slow, too (MxT)
Can we do better?

A: yes!
We can reduce both
- T, and
- M

SVD

# Step2: Faster than T timeticks



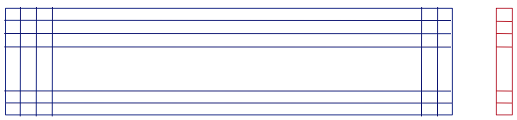



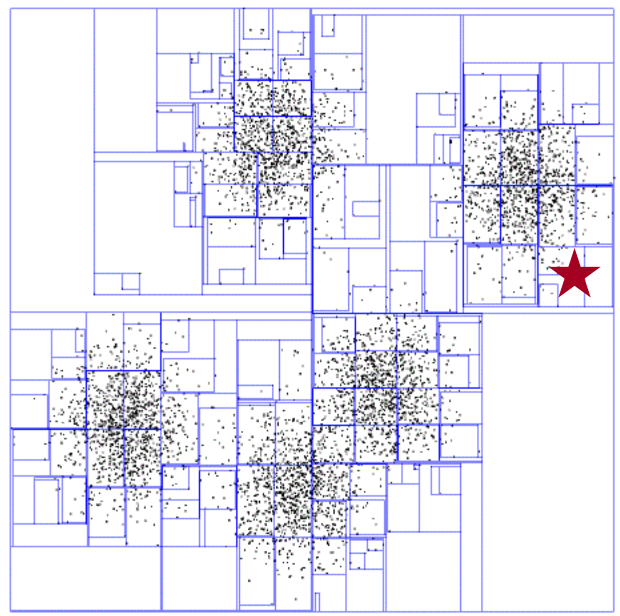
But sequential scan
Is slow, too (MxT)
Can we do better?

A: yes!
We can reduce both
- T, and
- M

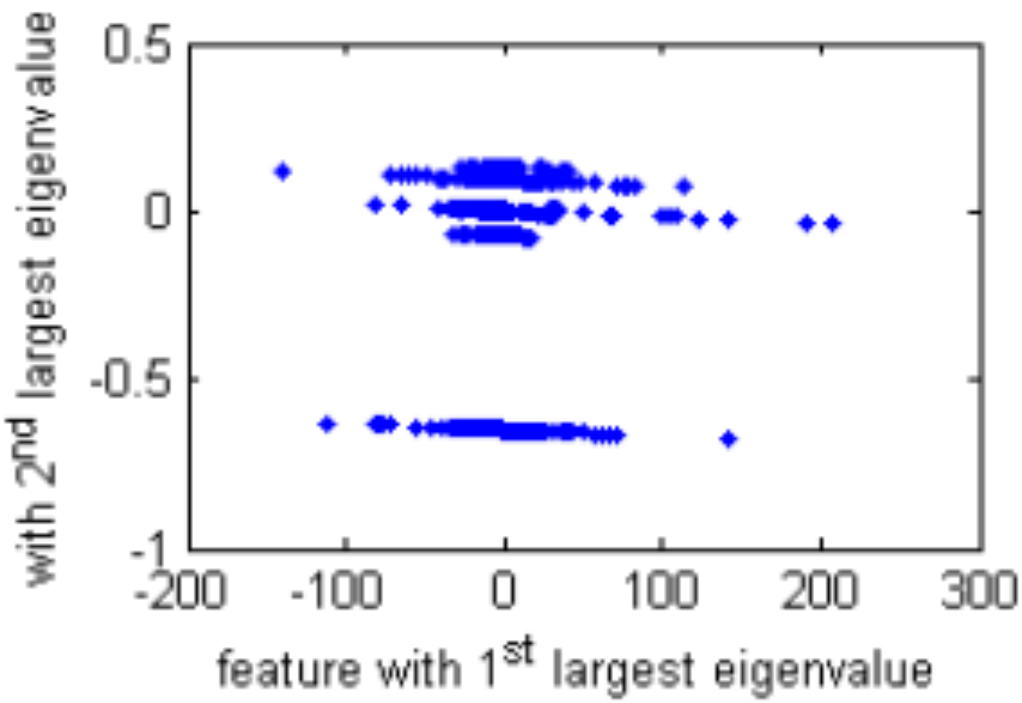
K-d trees
SVD

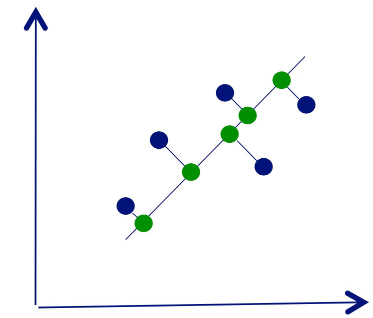
# Faster than seq. scan: K-d trees





# Thanks to SVD: VISUALIZATION!



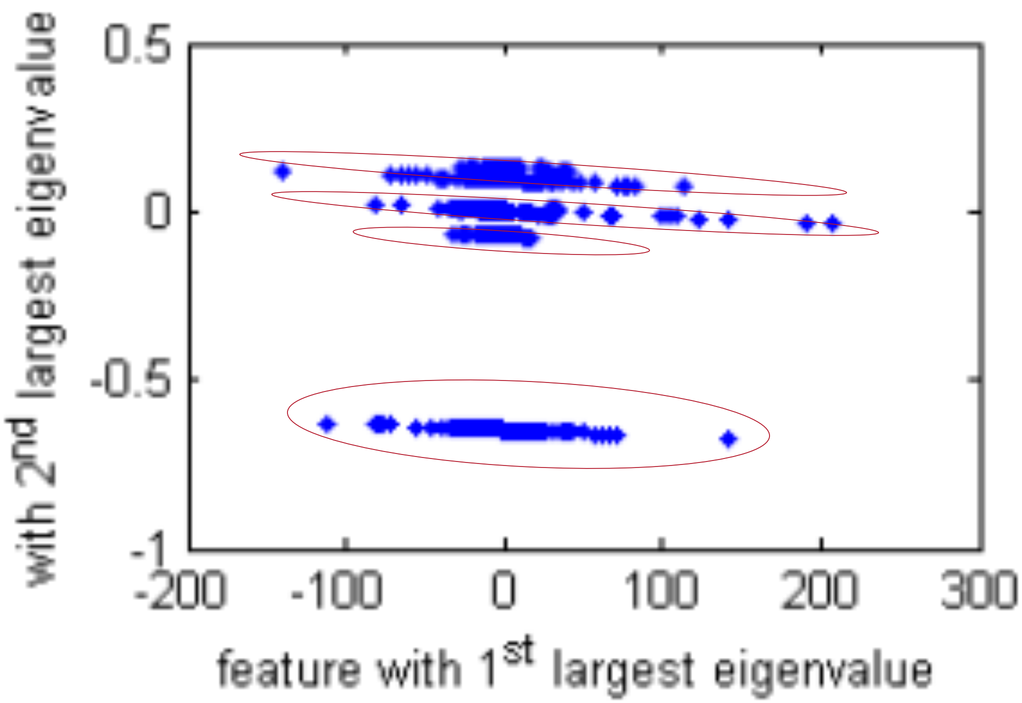



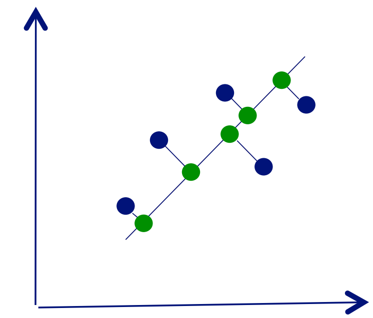
- Projection of measurements on to singular vectors of measurement matrix

[1] Yang Weng, Christos Faloutsos, Marija D. Ili´c, and Rohit Negi, Speed up of Data-Driven State Estimation Using Low-Complexity Indexing Method, IEEE PES-General Meeting, (accepted), 2014

Carnegie Mellon

# Thanks to SVD: VISUALIZATION!



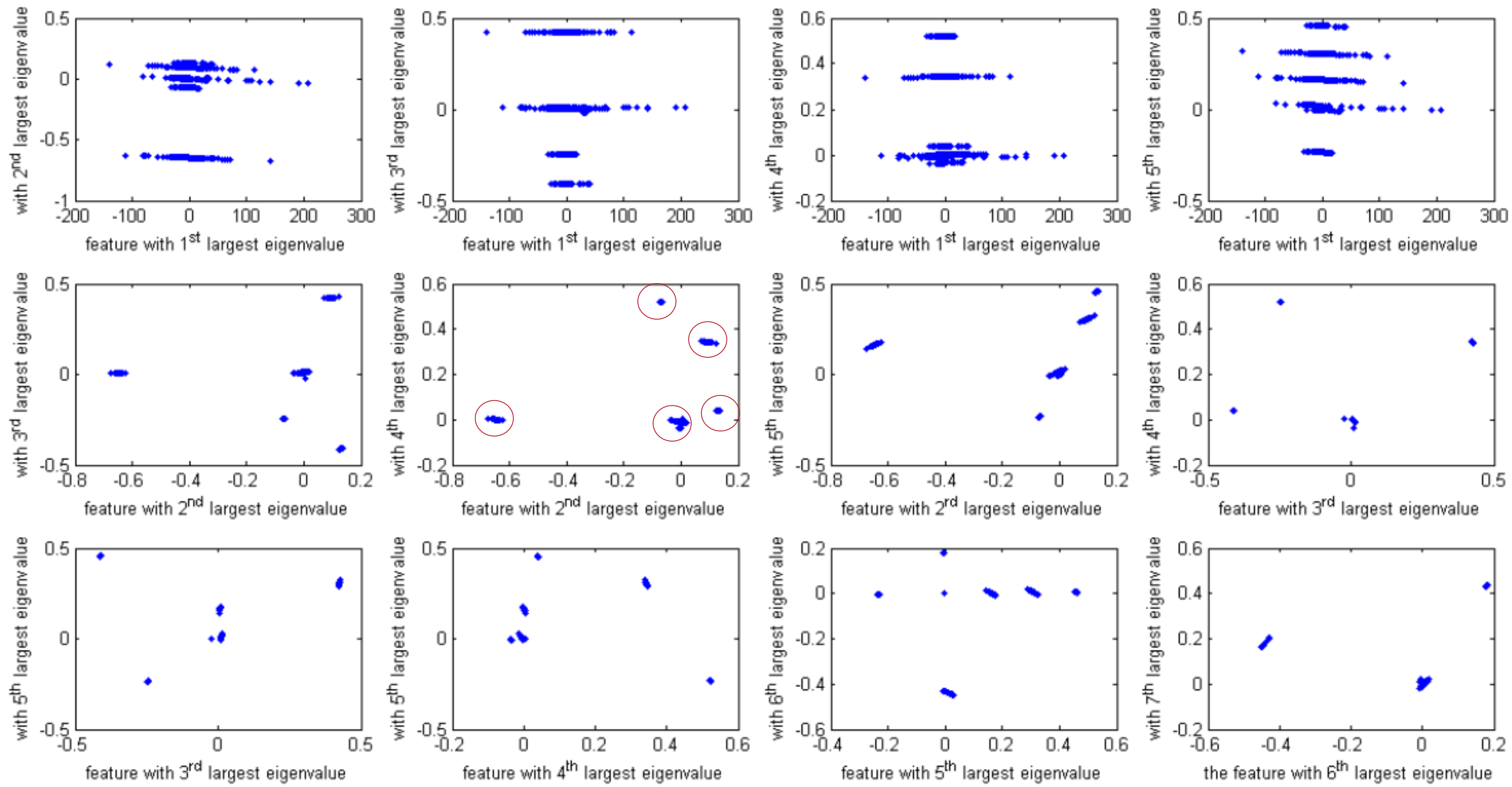



4 (or 5) groups of behavior!

- Projection of measurements on to singular vectors of measurement matrix

[1] Yang Weng, Christos Faloutsos, Marija D. Ili´c, and Rohit Negi, Speed up of Data-Driven State Estimation Using Low-Complexity Indexing Method, IEEE PES-General Meeting, (accepted), 2014

# Thanks to SVD: VISUALIZATION!



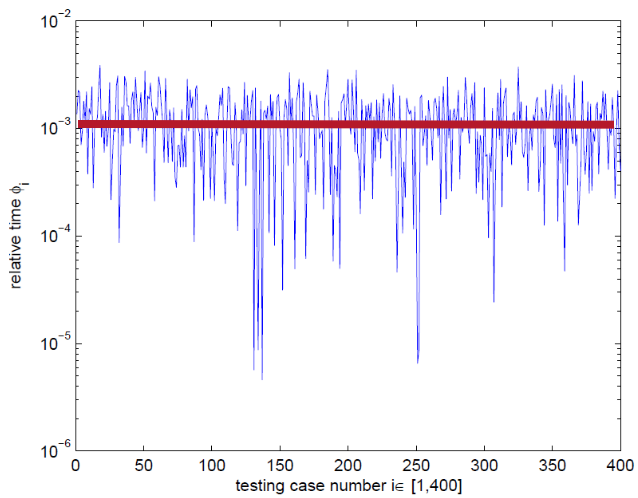

- Projection of measurements on to singular vectors of measurement matrix

[1] Yang Weng, Christos Faloutsos, Marija D. Ili´c, and Rohit Negi, Speed up of Data-Driven State Estimation Using Low-Complexity Indexing Method, IEEE PES-General Meeting, (accepted), 2014

# Crossdisciplinarity: Already started paying off

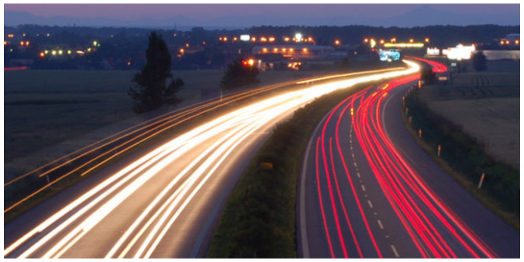
- Same accuracy, **100x – 100K x faster**

**1000 x faster**




**[1] Yang Weng, Christos Faloutsos, Marija D. Ili´c, and Rohit Negi, Speed up of Data-Driven State Estimation Using Low-Complexity Indexing Method, IEEE PES-General Meeting, (accepted), 2014**

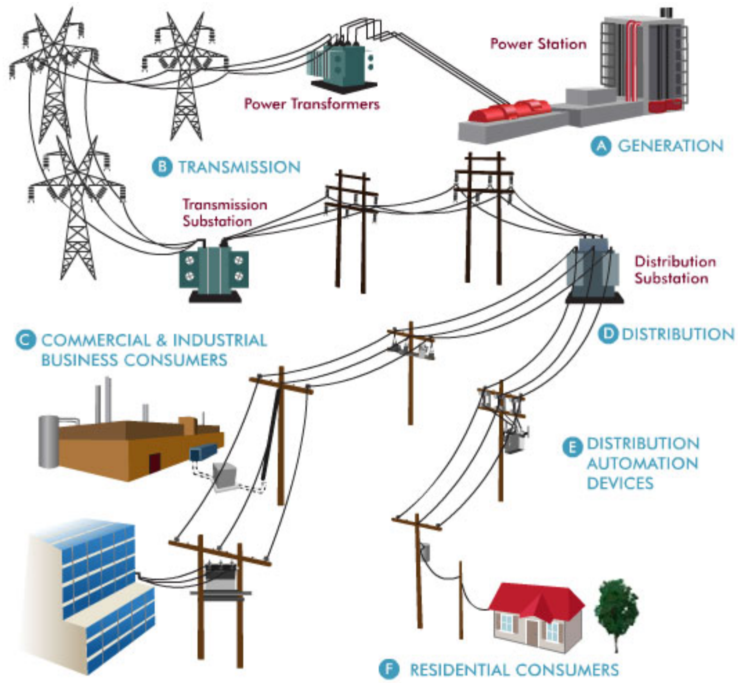
# Roadmap



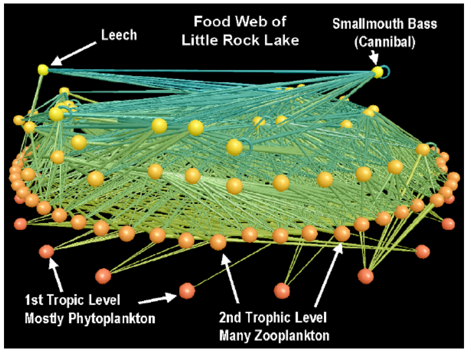
- A case for cross-disciplinarity
- ➡ Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Patterns in graphs
- Part#2: Cascade analysis
- Conclusions

# Graphs - why should we care?

- Power-grid!
  - Nodes: (plants/ consumers)
  - Edges: power lines




# Graphs - why should we care?




Food Web
[Martinez '91]

>$10B revenue

>0.5B users

Internet Map
[lumeta.com]

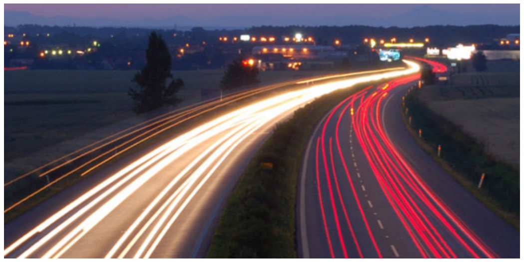
# Graphs - why should we care?

- web-log ('blog') news propagation
- computer network security: email/IP traffic and anomaly detection
- Recommendation systems
- ....

- Many-to-many db relationship -> graph

Carnegie Mellon

# Roadmap

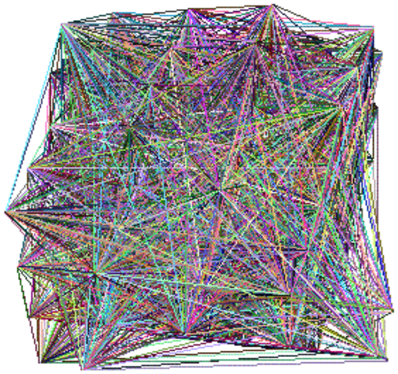


- A case for cross-disciplinarity
- Introduction – Motivation
  - Why study (big) graphs?
- ➡ Part#1: Patterns in graphs
- Part#2: Cascade analysis
- Conclusions

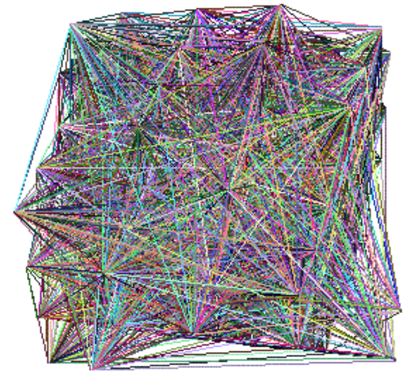
# Part 1: Patterns & Laws

Carnegie Mellon

# Laws and patterns



- Q1: Are real graphs random?



# Laws and patterns

- Q1: Are real graphs random?
- A1: NO!!
  - – Diameter
  - – in- and out- degree distributions
  - – other (surprising) patterns
- Q2: why so many power laws?
- A2: <self-similarity – stay tuned>

- So, let’s look at the data

# Solution# S.1

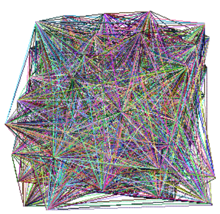
- Power law in the degree distribution [SIGCOMM99]

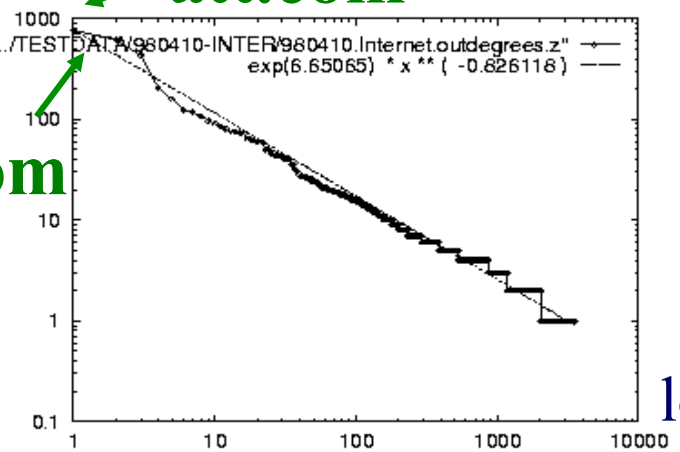
**internet domains**

**att.com**

log(degree)

**ibm.com**



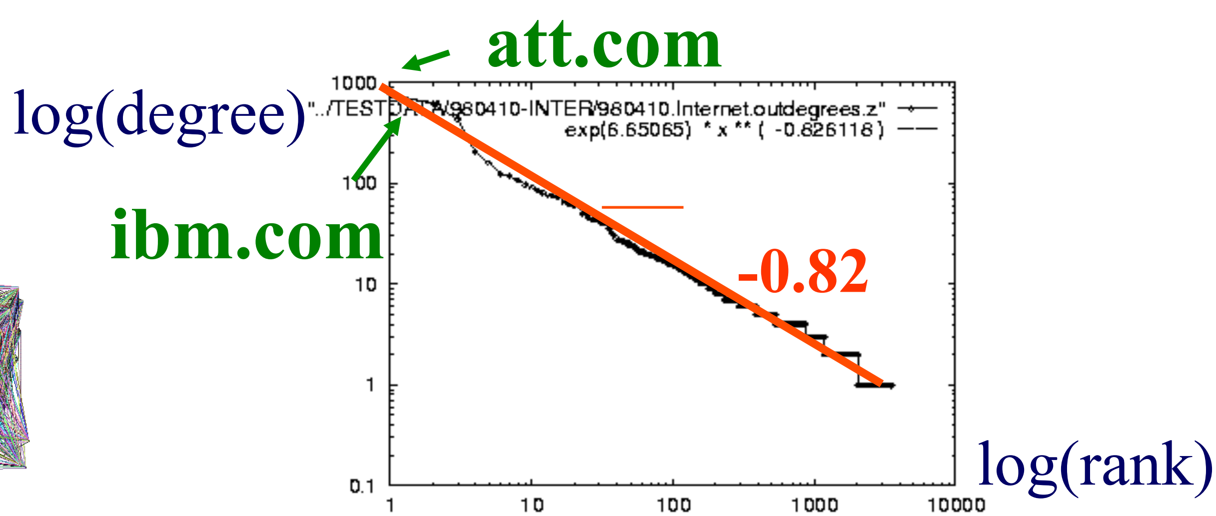


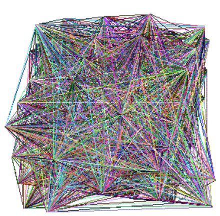

log(rank)

# Solution# S.1

- Power law in the degree distribution [SIGCOMM99]

**internet domains**

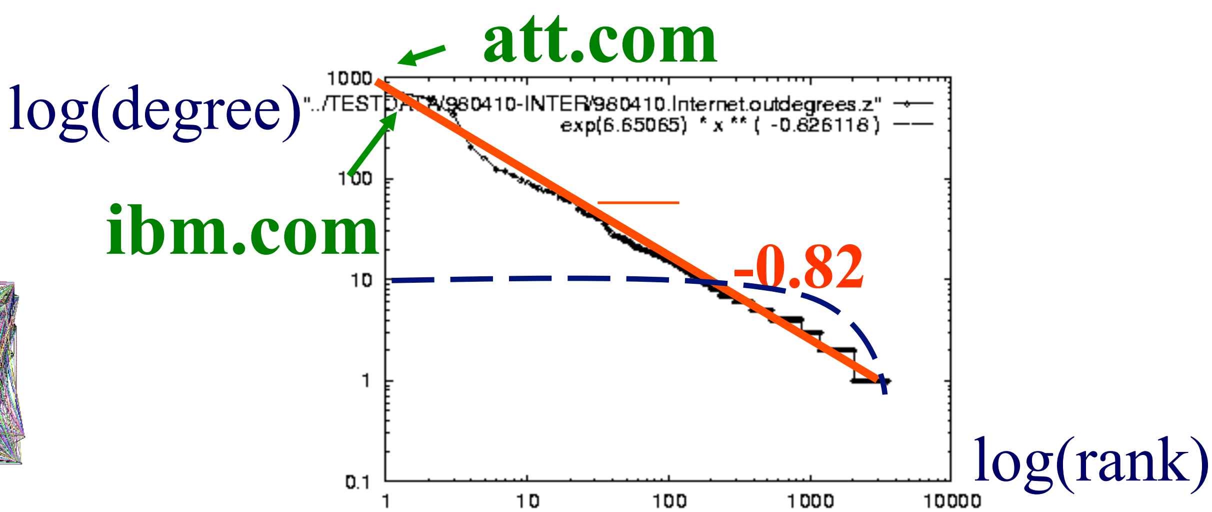


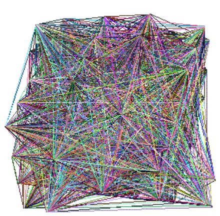



# Solution# S.1

- Q: So what?

**internet domains**



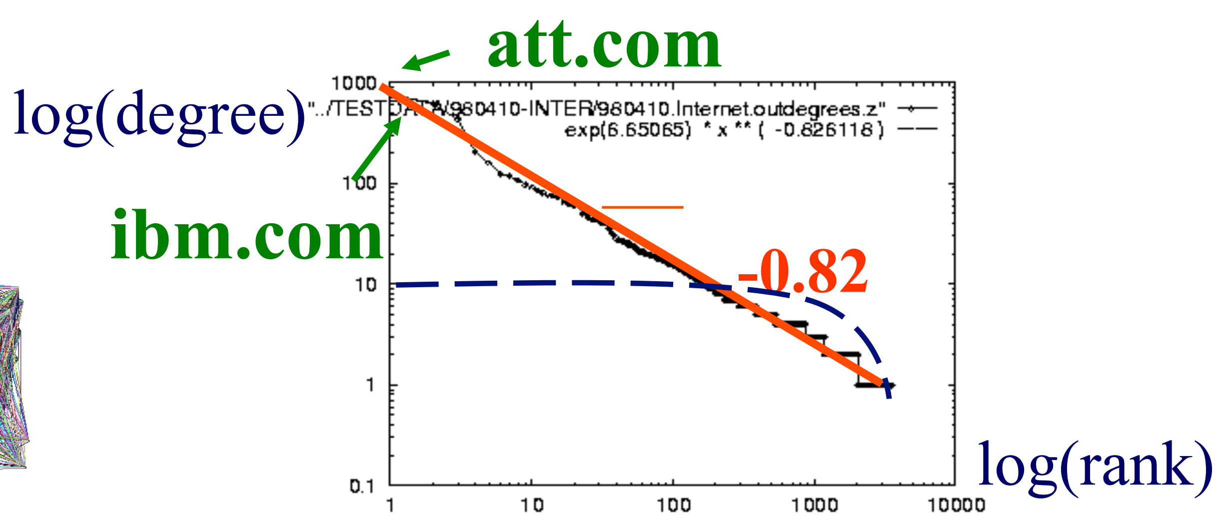



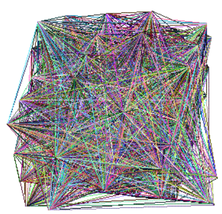
# Solution# S.1

- Q: So what?
- A1: # of two-step-away pairs:

= friends of friends (F.O.F.)

**internet domains**



**Gaussian trap**

# Solution# S.1

= friends of friends (F.O.F.)

- Q: So what?
- A1: # of two-step-away pairs: O(d_max ^2) ~ 10M^2

**internet domains**

⇩

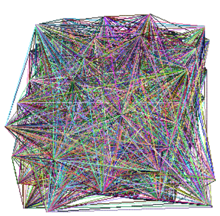
~0.8PB ->
a data center(!)

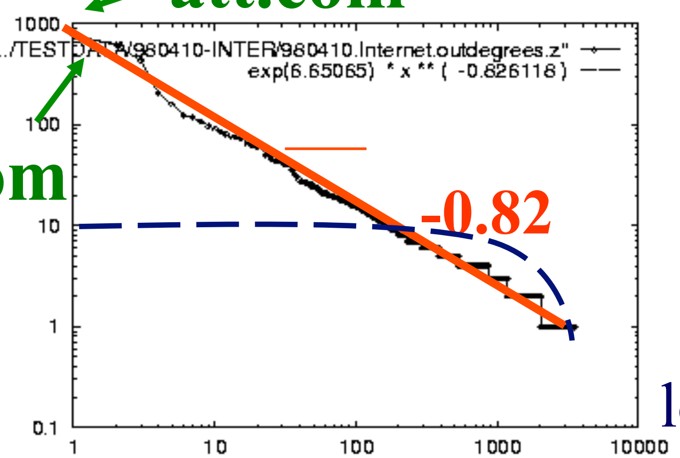
att.com

log(degree)

ibm.com

-0.82






DCO @ CMU

**Gaussian trap**

# Solution# S.1

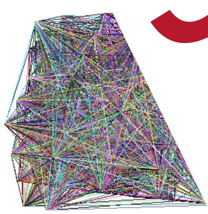
- Q: So what?
- A1: # of two-step-aw... ?) ~ 10M^2

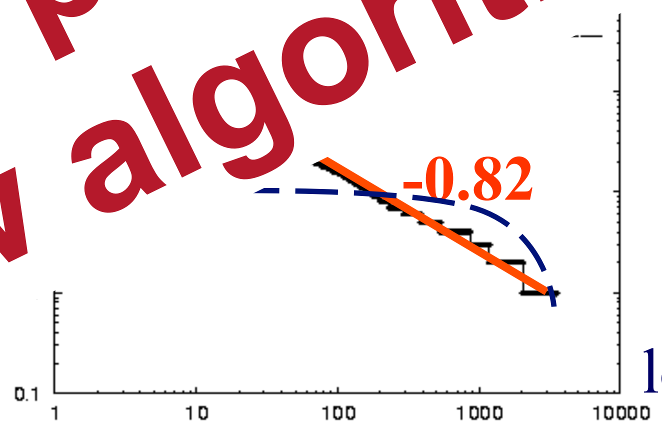
**inte...**

⇩

~0.8PB ->
a data center(!)

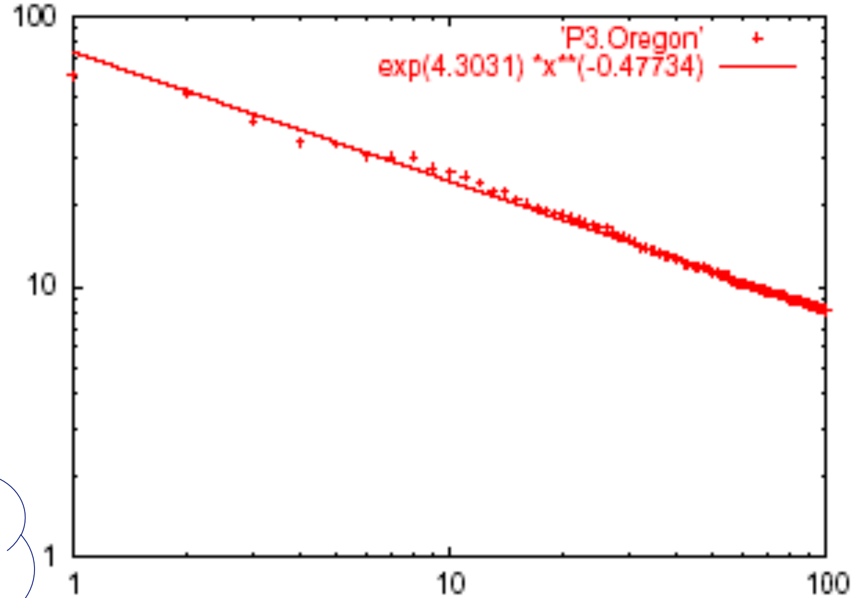
**Such patterns ->**
**New algorithms**






# Solution# S.2: Eigen Exponent *E*

Eigenvalue




Rank of decreasing eigenvalue

Exponent = slope

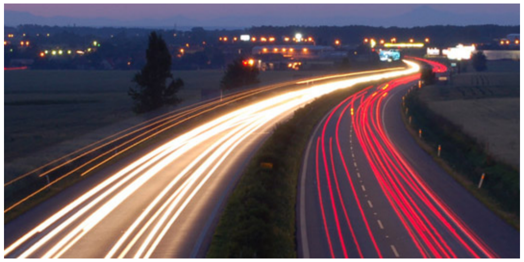
$E = -0.48$

May 2001

$A\,x = \lambda\,x$

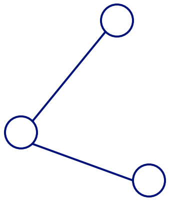
- A2: power law in the eigenvalues of the adjacency matrix

# Roadmap



- Introduction – Motivation
- Problem#1: Patterns in graphs
  - Static graphs
    - degree, diameter, eigen,
    - Triangles
  - Time evolving graphs
- Problem#2: Tools

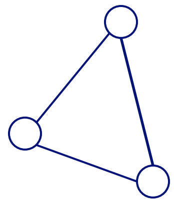
# Solution# S.3: Triangle 'Laws'



- Real social networks have a lot of triangles

Carnegie Mellon

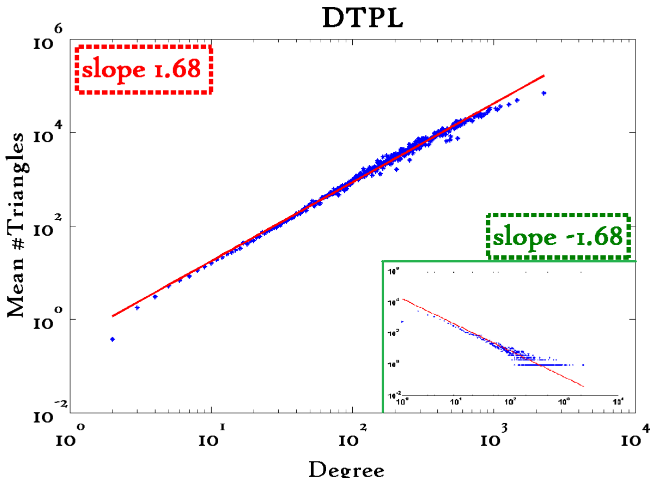
# Solution# S.3: Triangle 'Laws'



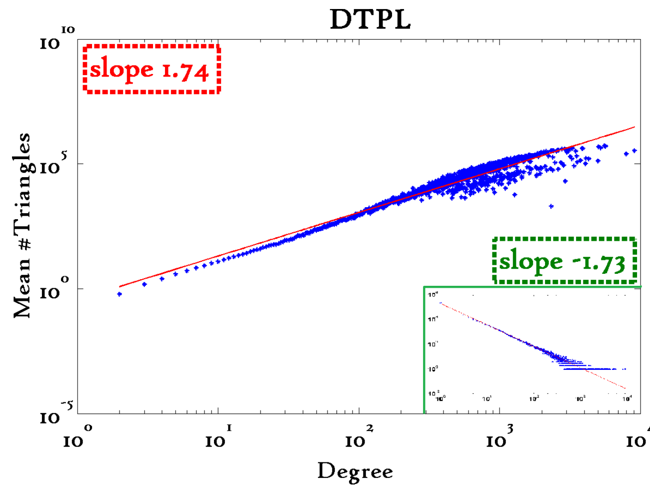
- Real social networks have a lot of triangles
  - Friends of friends are friends
- Any patterns?
  - 2x the friends, 2x the triangles ?

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

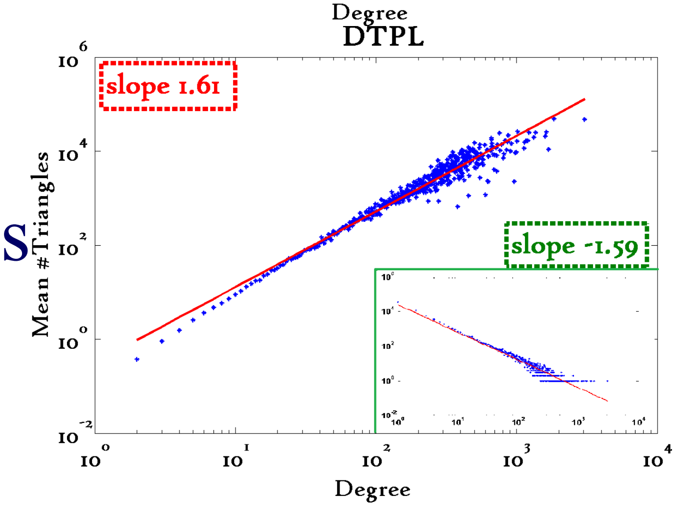
Reuters

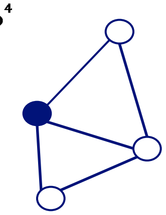



SN




Epinions



X-axis: degree

Y-axis: mean # triangles

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

details

# Triangle Law: Computations
## [Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos) – $O(d_{max}^2)$
Q: Can we do that quickly?
A:

details

# Triangle Law: Computations

## [Tsourakakis ICDM 2008]

But: triangles are expensive to compute

(3-way join; several approx. algos) – $O(d_{max}{}^2)$

Q: Can we do that quickly?

A: Yes!

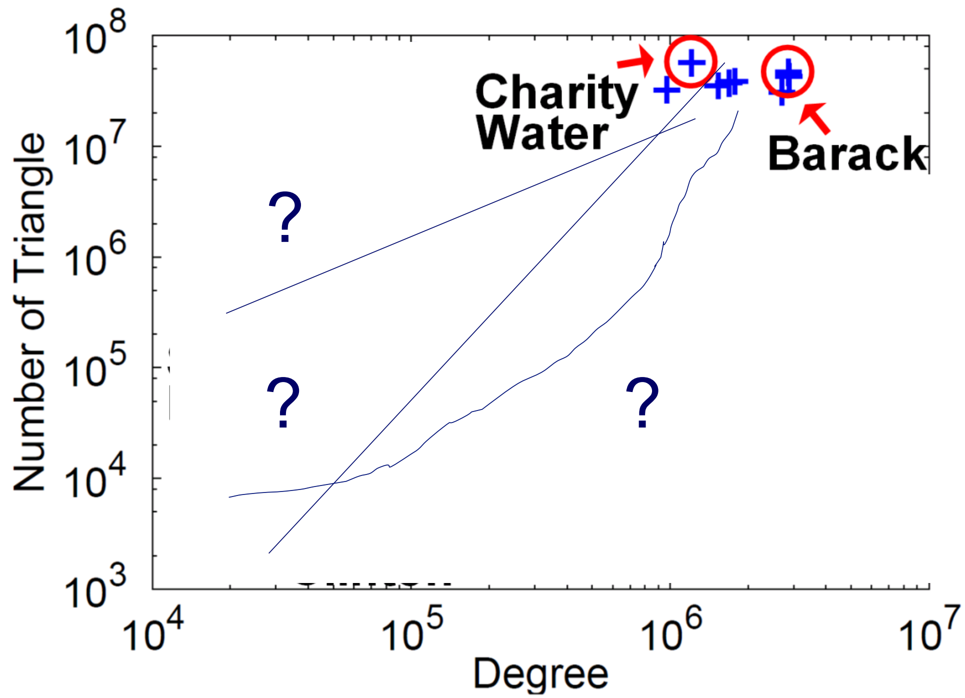
**#triangles = 1/6 Sum ( $\lambda_i^3$ )**

(and, because of skewness (S2) ,

we only need the top few eigenvalues! - O(E)

$A x = \lambda x$

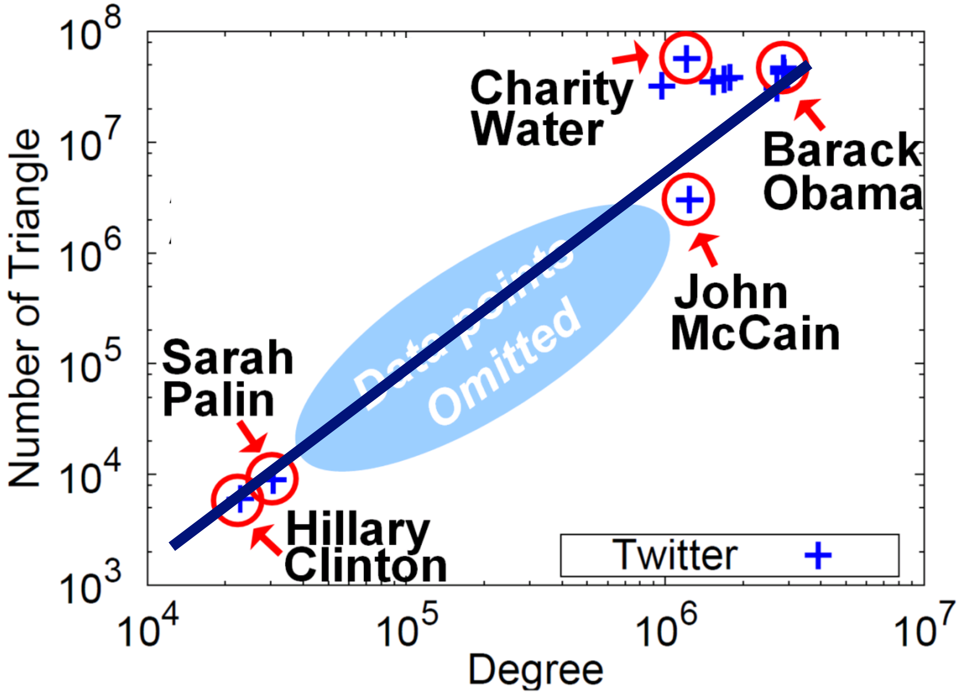
# Triangle counting for large graphs?


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

# Triangle counting for large graphs?



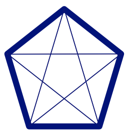


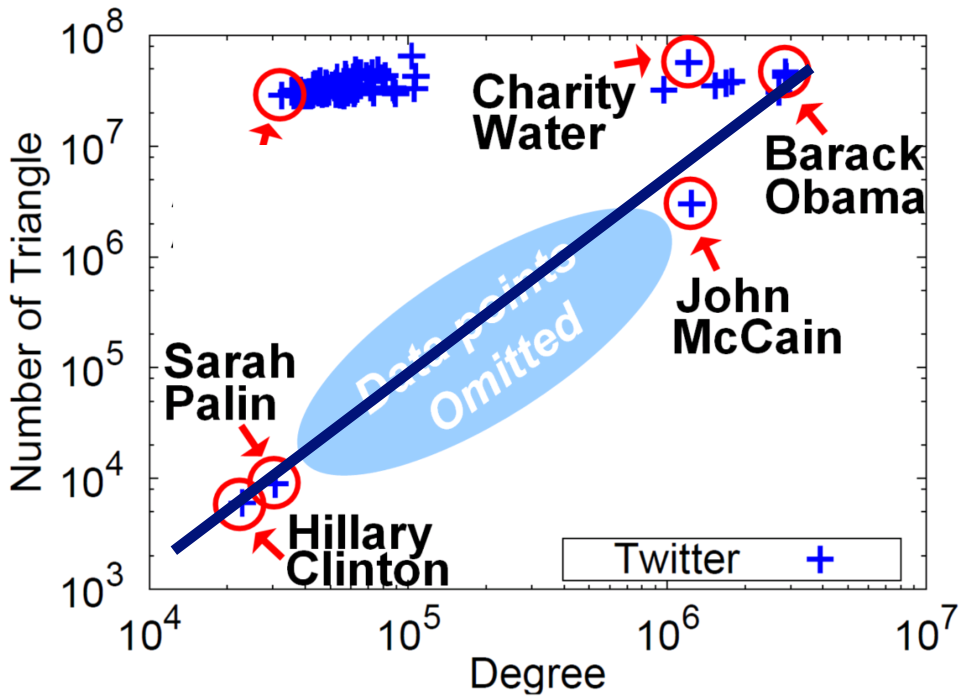
Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

Carnegie Mellon

# Triangle counting for large graphs?



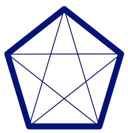


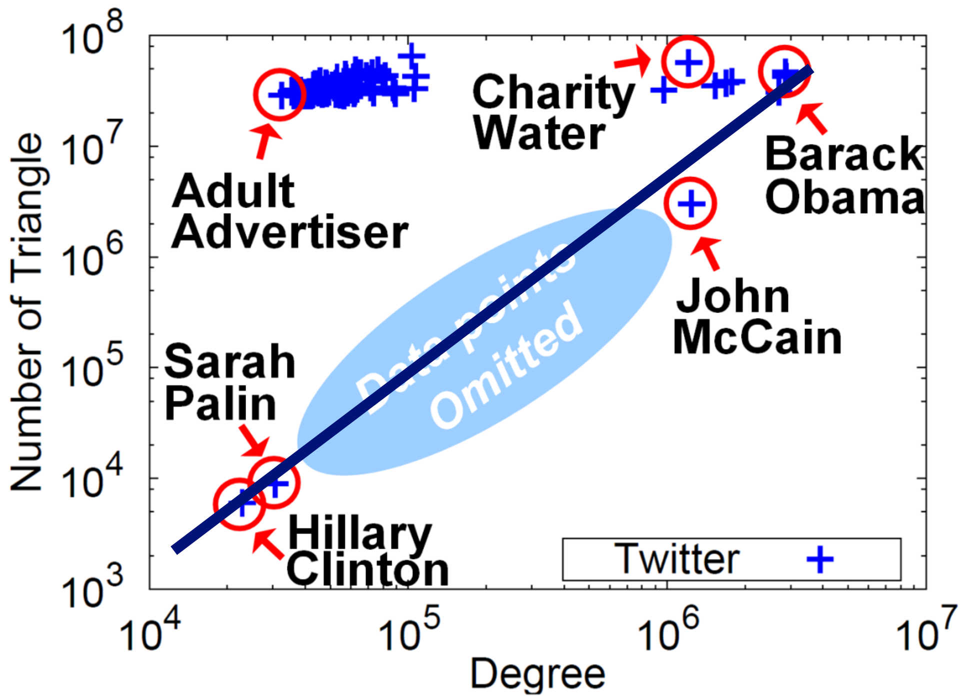


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

# Triangle counting for large graphs?



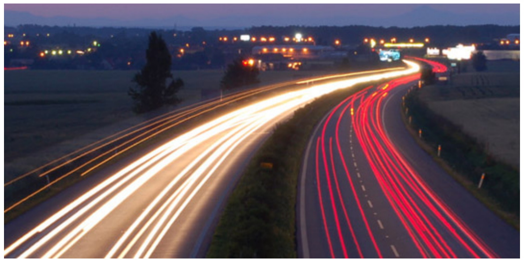




Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

# Roadmap

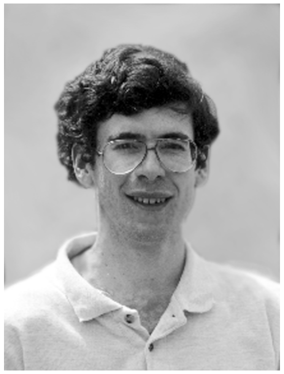


- A case for cross-disciplinarity
- Introduction – Motivation
- Part#1: Patterns in graphs
  - Static graphs
  - ➡ Time evolving graphs
- Part#2: Cascade analysis
- Conclusions

# Problem: Time evolution

- with Jure Leskovec (CMU -> Stanford)

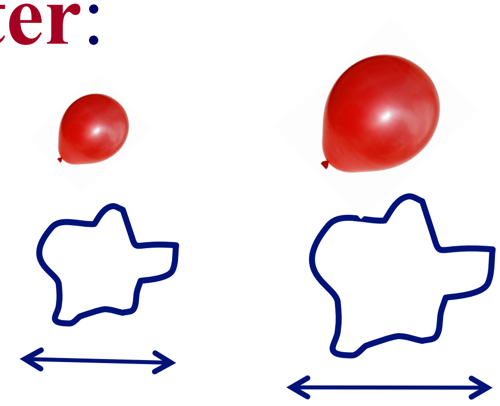
- and Jon Kleinberg (Cornell – sabb. @ CMU)



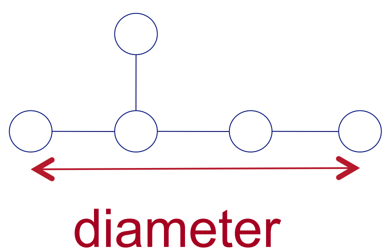
Jure Leskovec, Jon Kleinberg and Christos Faloutsos: *Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations*, KDD 2005

# T.1 Evolution of the Diameter

- Prior work on Power Law graphs hints at **slowly growing diameter**:
  - [diameter ~ O( $N^{1/3}$)]
  - diameter ~ O(log N)
  - diameter ~ O(log log N)

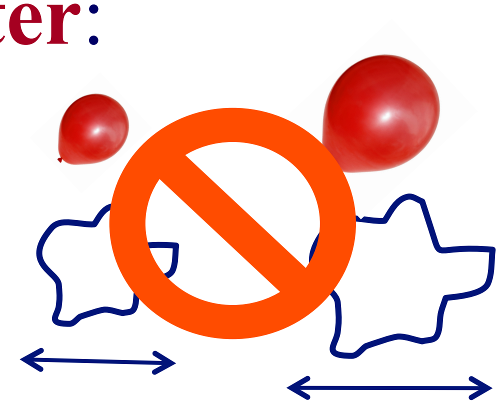


- What is happening in real data?



# T.1 Evolution of the Diameter

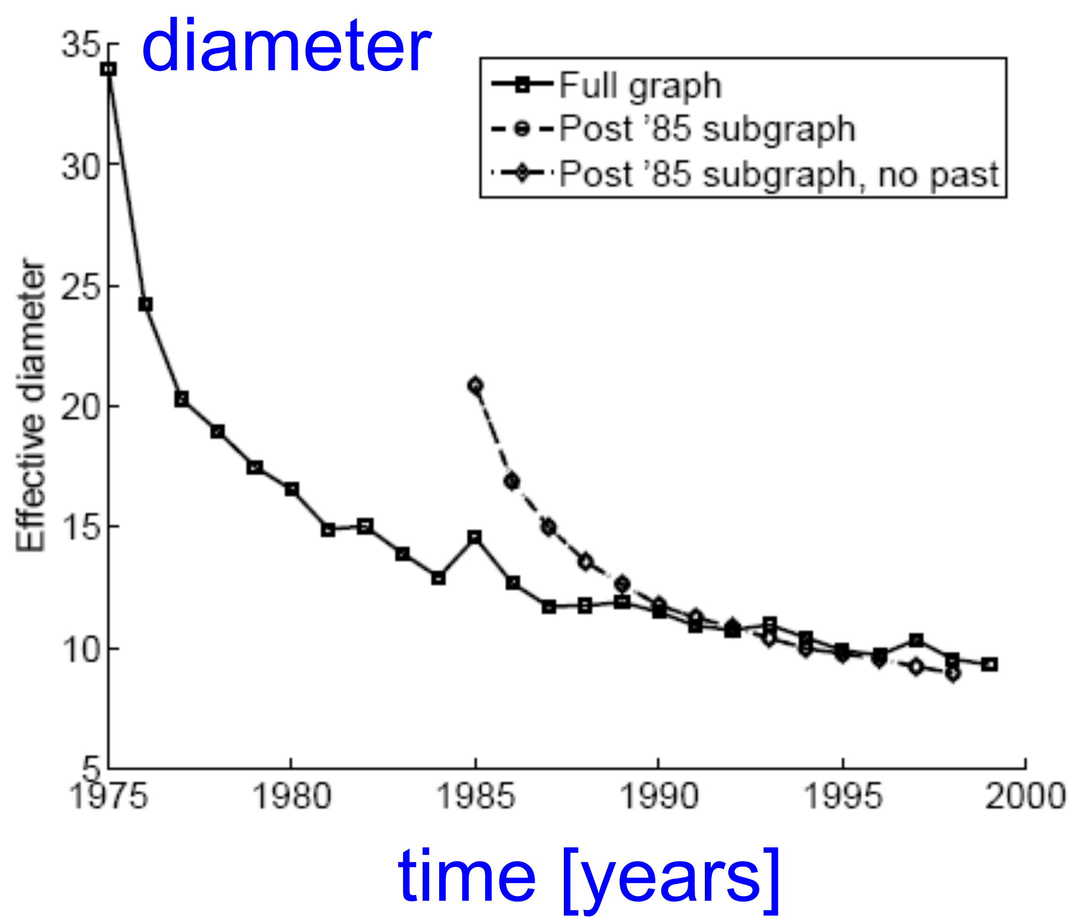
- Prior work on Power Law graphs hints at **slowly growing diameter**:
  - [diameter ~ $O(N^{1/3})$]
  - diameter ~ O(log N)
  - diameter ~ O(log log N)



- What is happening in real data?
- Diameter **shrinks** over time

# T.1 Diameter – "Patents"

- Patent citation network
- 25 years of data
- @1999
  - – 2.9 M nodes
  - – 16.5 M edges



# T.2 Temporal Evolution of the Graphs

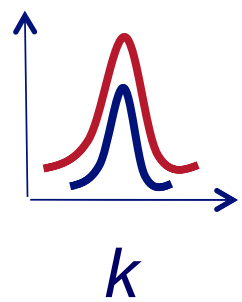
- N(t) … nodes at time t
- E(t) … edges at time t
- Suppose that

  $N(t+1) = 2 * N(t)$

- Q: what is your guess for

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

Say, *k* friends on average



# T.2 Temporal Evolution of the Graphs

**Gaussian trap**

- N(t) … nodes at time t
- E(t) … edges at time t
- Suppose that

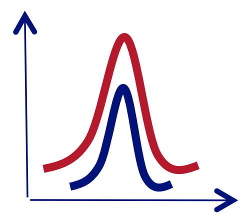
  $N(t+1) = 2 * N(t)$

- Q: what is your guess for

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

- A: over-doubled! ~ 3x
  - But obeying the ``Densification Power Law''

Say, *k* friends on average



# T.2 Temporal Evolution of the Graphs

**Gaussian trap**

- N(t) … nodes at time t
- E(t) … edges at time t
- Suppose that

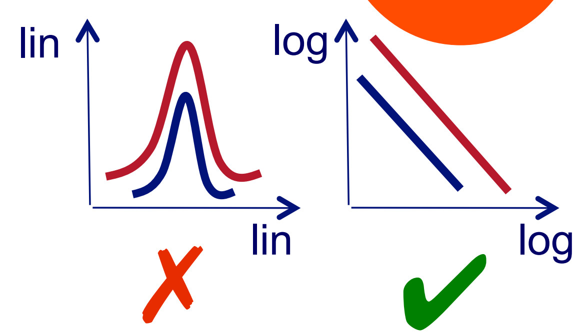
  N(t+1) = 2 * N(t)

- Q: what is your guess for

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

- A: over-doubled! ~ 3x
  - But obeying the ``Densification Power Law''

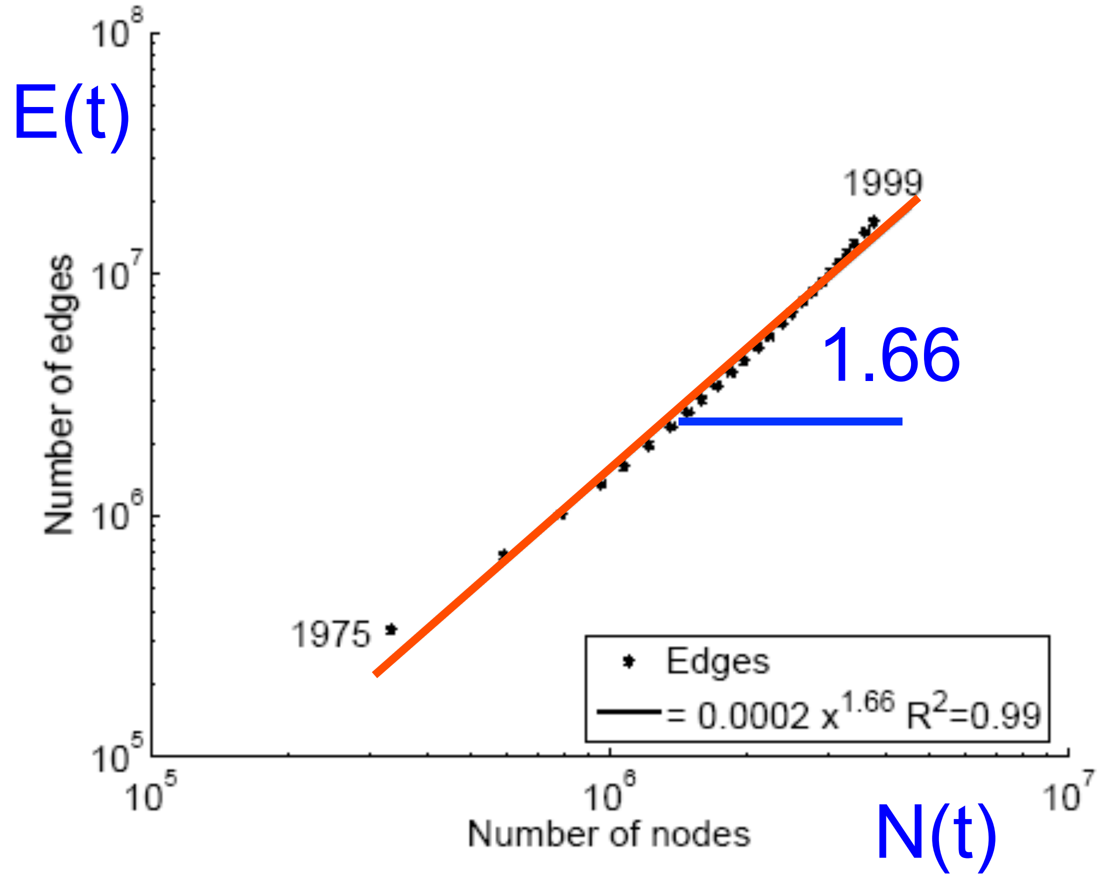
Say, *k* friends on average



Carnegie Mellon

# T.2 Densification – Patent Citations

- Citations among patents granted
- @1999
  - 2.9 M nodes
  - 16.5 M edges
- Each year is a datapoint



# 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]<br>**L02.** Triangle Power Law (TPL) [Tsourakakis `08]<br>**L03.** Eigenvalue Power Law (EPL) [Siganos et al. `03]<br>**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]<br>**L06.** Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05]<br>**L07.** Constant size 2nd and 3rd connected components [McGlohon et al. `08]<br>**L08.** Principal Eigenvalue Power Law ($\lambda_1$PL) [Akoglu et al. `08]<br>**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

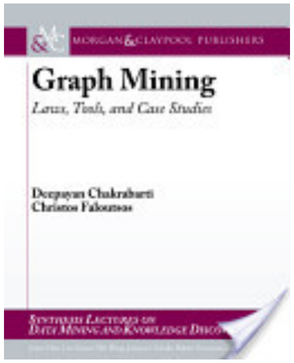
| | Unweighted | Weighted |
|---|---|---|
| Static | ✔ L01. Power-law degree distribution [Faloutsos et al. `99, Kleinberg et al. `99, Chakrabarti et al. `04, Newman `04]<br>✔ L02. Triangle Power Law (TPL) [Tsourakakis `08]<br>✔ L03. Eigenvalue Power Law (EPL) [Siganos et al. `03]<br>L04. Community structure [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]<br>✔ L06. Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05]<br>L07. Constant size 2nd and 3rd connected components [McGlohon et al. `08]<br>L08. Principal Eigenvalue Power Law ($\lambda_1$PL) [Akoglu et al. `08]<br>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]<br>L02. Triangle Power Law (TPL) [Tsourakakis `08]<br>L03. Eigenvalue Power Law (EPL) [Siganos et al. `03]<br>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]<br>L06. Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05]<br>L07. Constant size 2nd and 3rd connected components [McGlohon et al. `08]<br>L08. Principal Eigenvalue Power Law ($\lambda_1$PL) [Akoglu et al. `08]<br>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] |

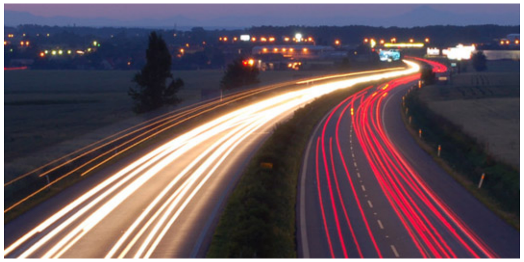
- Mary McGlohon, Leman Akoglu, Christos Faloutsos. *Statistical Properties of Social Networks.* in "Social Network Data Analytics" (Ed.: Charu Aggarwal)




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

SKIP

# Roadmap



- A case for cross-disciplinarity
- Introduction – Motivation
- Part#1: Patterns in graphs
  - …
  - ➡ Why so many power-laws?
- Part#2: Cascade analysis
- Conclusions

SKIP

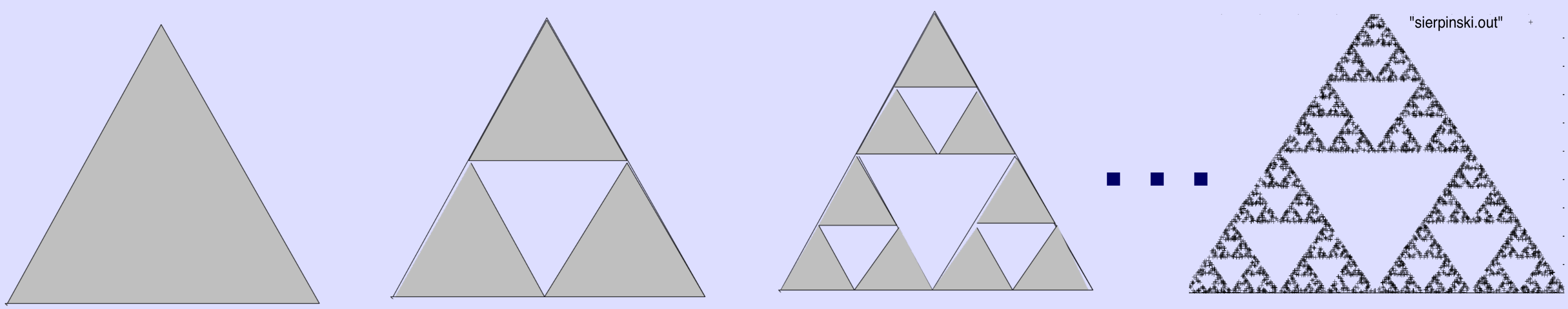
# Why so many P.L.?

- Possible answer: self-similarity / fractals

SKIP

# 20'' intro to fractals

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

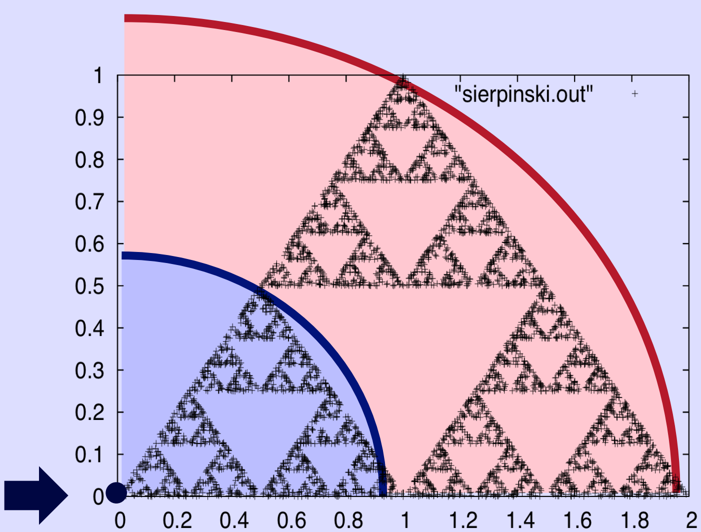


SKIP

# 20'' intro to fractals

Self-similarity -> no char. scale
-> power laws, eg:
2x the radius,
3x the #neighbors nn(r)

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



SKIP

# 20'' intro to fractals

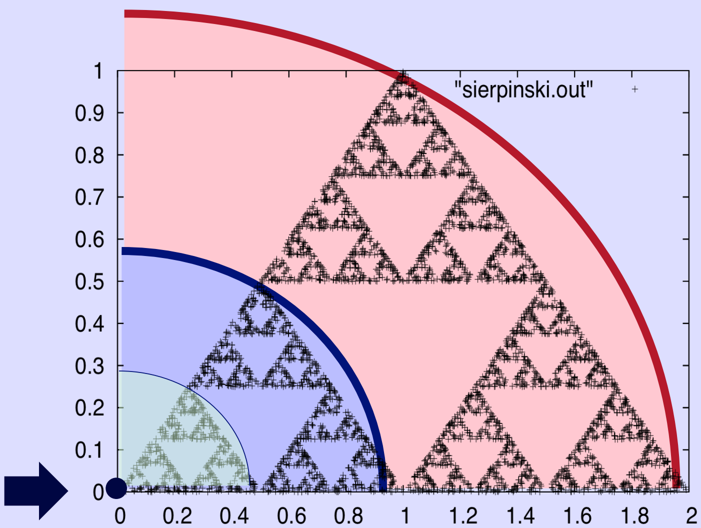
Self-similarity -> no char. scale

-> power laws, eg:

2x the radius,

3x the #neighbors nn(r)

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



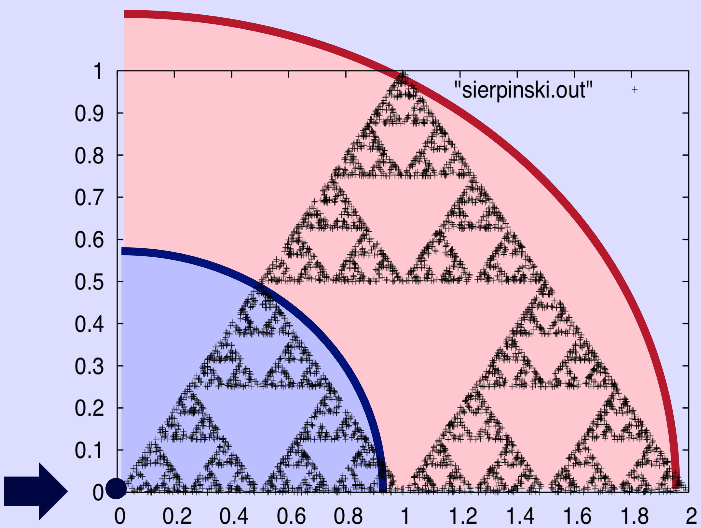
SKIP

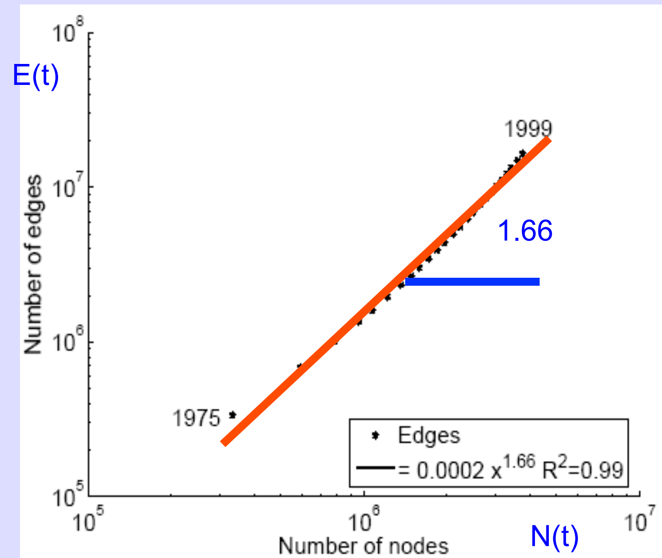
# 20'' intro to fractals

Self-similarity -> no char. scale
-> power laws, eg:
2x the radius,
3x the #neighbors

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

Reminder:
Densification P.L.
(2x nodes, ~3x edges)



SKIP

# 20'' intro to fractals

Self-similarity -> no char. scale

-> power laws, eg:

2x the radius,

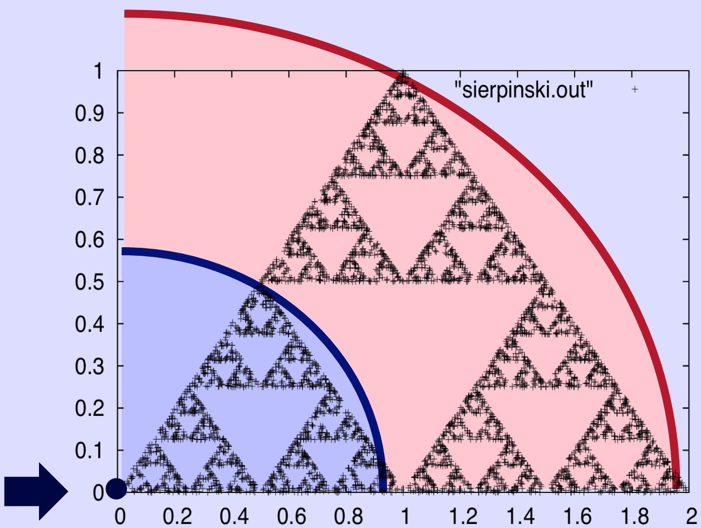
3x the #neighbors

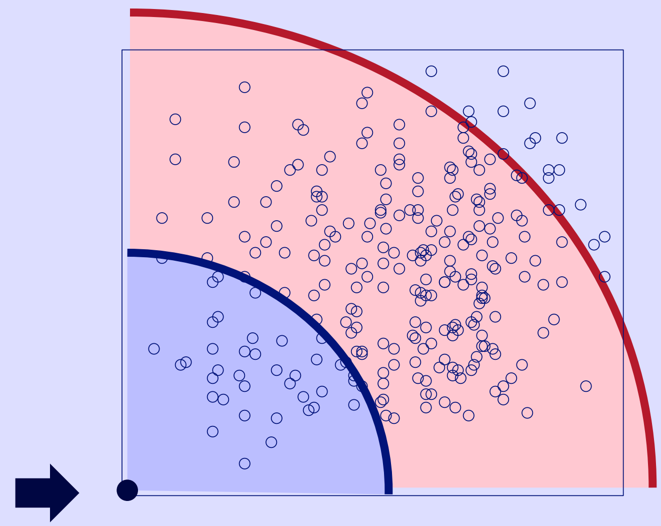
$nn = C\, r^{\log 3/\log 2}$

2x the radius,

4x neighbors

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



SKIP

# 20'' intro to fractals

Self-similarity -> no char. scale

-> power laws, eg:

2x the radius,

3x the #neighbors

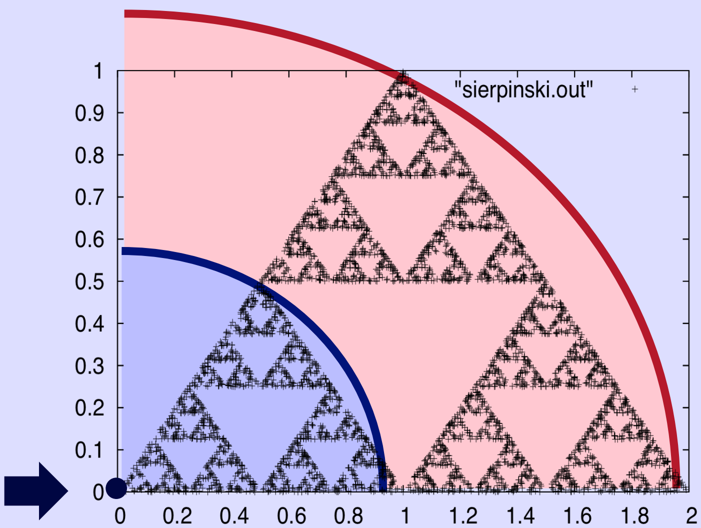
$nn = C\ r^{\log 3 / \log 2}$ =1.58

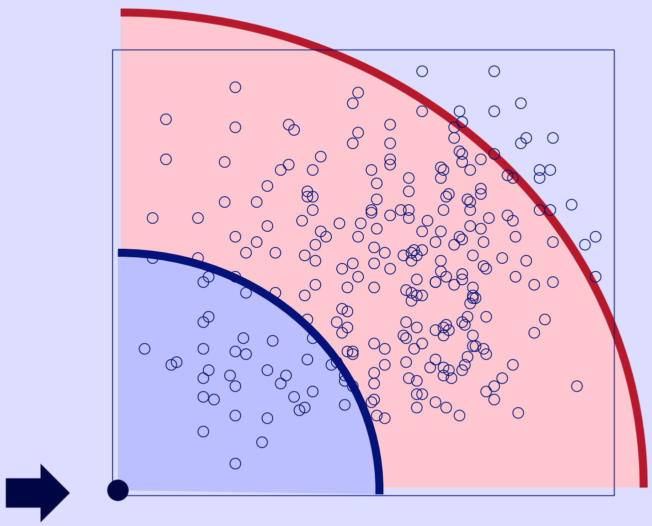
2x the radius,

4x neighbors

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

Fractal dim.



SKIP

# 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.

SKIP

# 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'

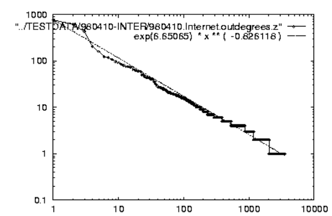
*R-MAT: A Recursive Model for Graph Mining*,
by D. Chakrabarti, Y. Zhan and C. Faloutsos,
SDM 2004, Orlando, Florida, USA

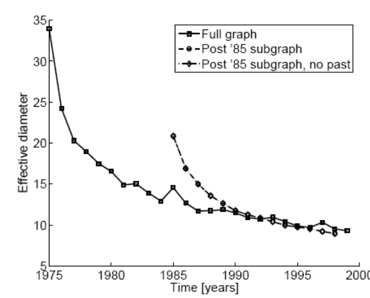
*Realistic, Mathematically Tractable Graph Generation and Evolution, Using Kronecker Multiplication,*
by J. Leskovec, D. Chakrabarti, J. Kleinberg,
and C. Faloutsos, in PKDD 2005, Porto, Portugal

SKIP

# 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

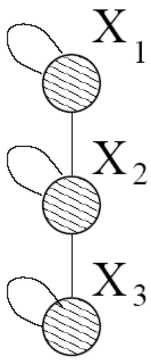


Carnegie Mellon

SKIP

# Kronecker Graphs



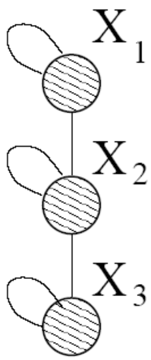

| | | |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 1 |

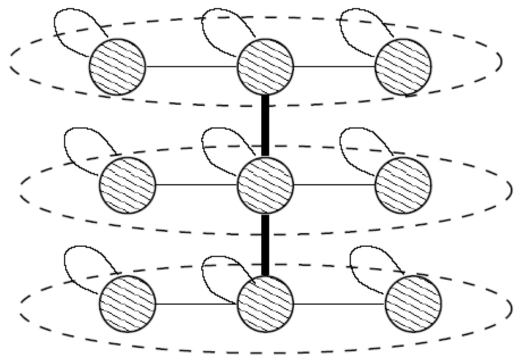
$G_1$

Adjacency matrix

SKIP

# Kronecker Graphs



Intermediate stage

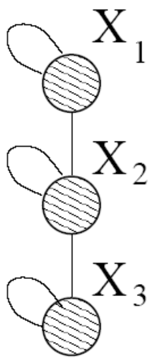
| | | |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 1 |

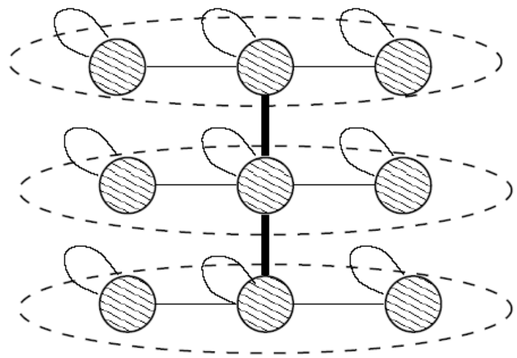
$G_1$

Adjacency matrix

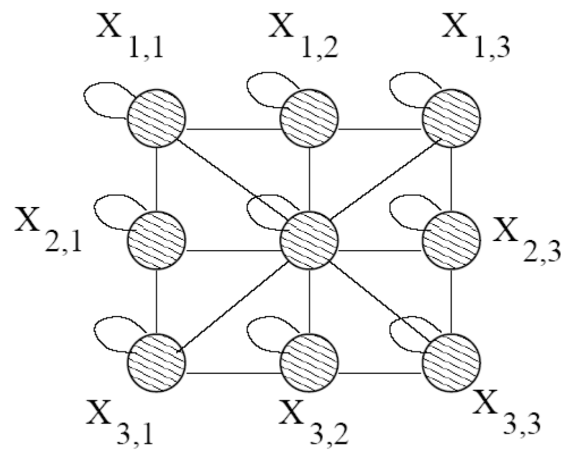
SKIP

# 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 | $G_1$ | $G_1$ |

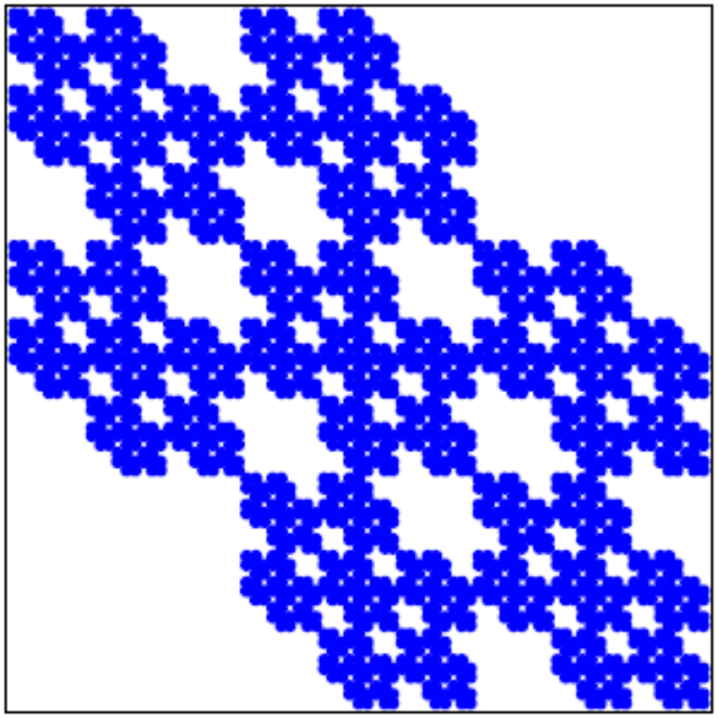
$$G_2 = G_1 \otimes G_1$$

Adjacency matrix

SKIP

# Kronecker Graphs

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

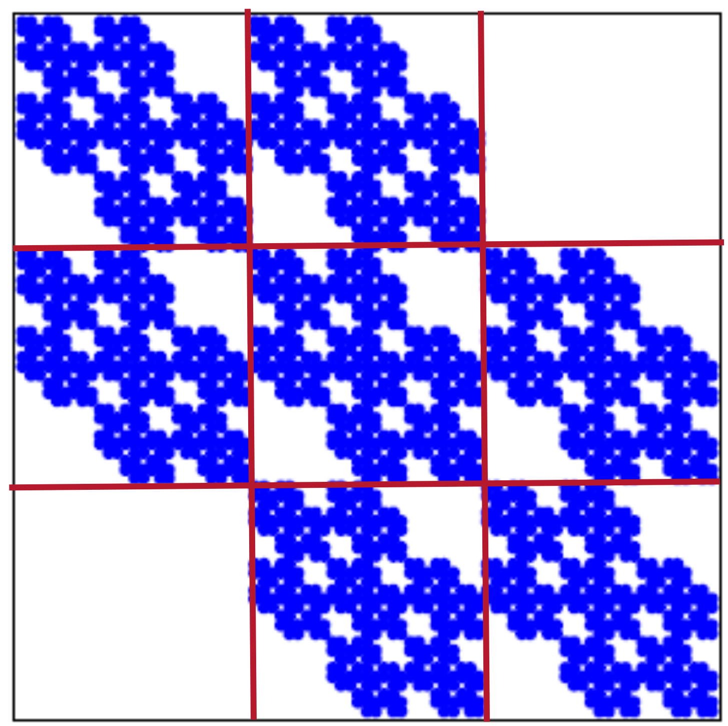


$G_4$ adjacency matrix

SKIP

# Kronecker Graphs

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

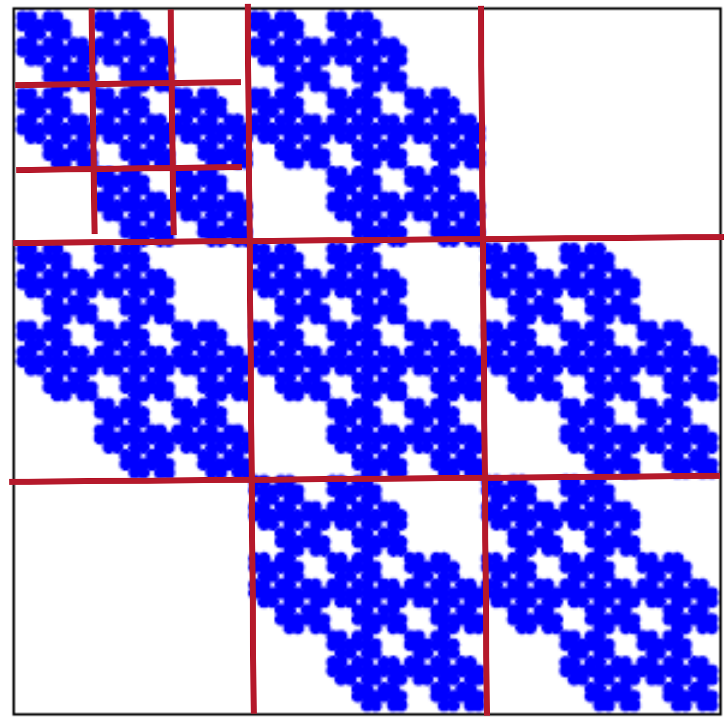


$G_4$ adjacency matrix

SKIP

# Kronecker Graphs

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



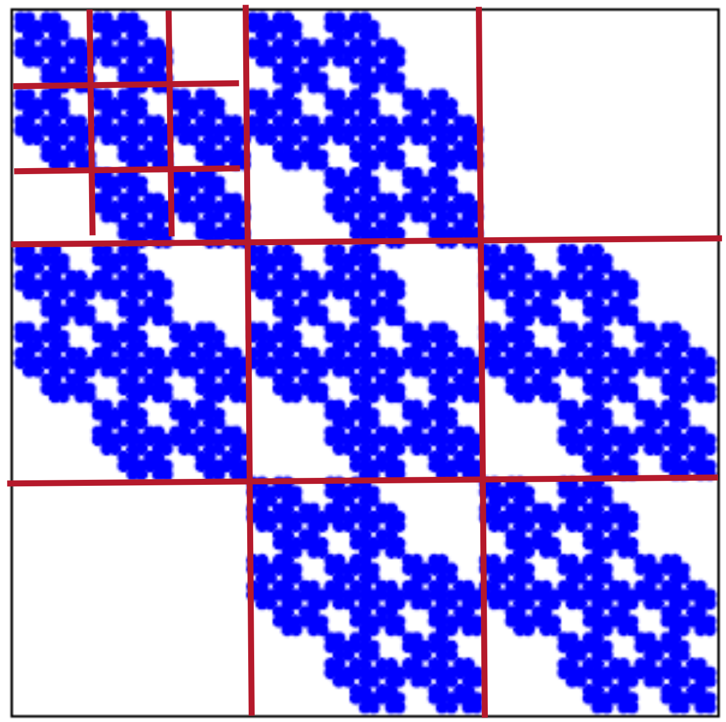
$G_4$ adjacency matrix

SKIP

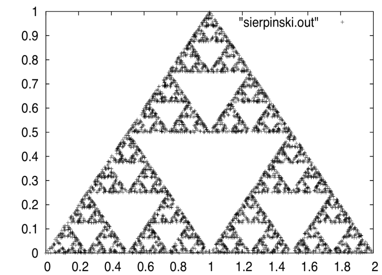
# Kronecker Graphs

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

Holes within holes; Communities within communities



$G_4$ adjacency matrix



# Properties:

- We can PROVE that
  - Degree distribution is multinomial ~ power law
  - Diameter: constant (new)
  - Eigenvalue distribution: multinomial
  - First eigenvector: multinomial

SKIP

# 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

SKIP

# 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, …

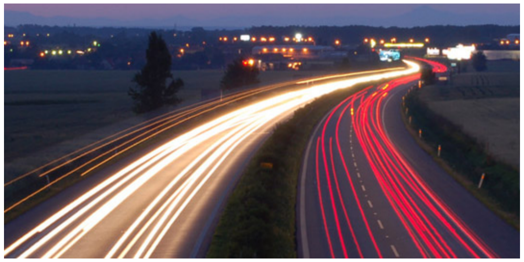
*To iterate is human, to recurse is devine*

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

# Summary of Part#1

- *many* patterns in real graphs
  - Small & shrinking diameters
  - Power-laws everywhere
  - Gaussian trap
- Self-similarity (RMAT/Kronecker): good model

# Roadmap



- A case for cross-disciplinarity
- Introduction – Motivation
- Part#1: Patterns in graphs
- ➡ Part#2: Cascade analysis
- Conclusions

# Comic relief:

- What would a barefooted man get if he steps on an electric wire?

http://energyquest.ca.gov/games/jokes/george.html

# Comic relief:

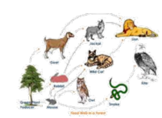
- What would a barefooted man get if he steps on an electric wire?
  (Answer) A pair of *shocks*

http://energyquest.ca.gov/games/jokes/george.html

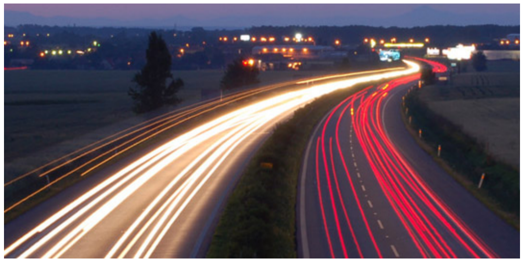
# Part 2: Cascades & Immunization

# Why do we care?

- Information Diffusion
- Viral Marketing
- Epidemiology and Public Health
- Cyber Security
- Human mobility
- Games and Virtual Worlds
- Ecology
- ........



# Roadmap



- A case for cross-disciplinarity
- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: Cascade analysis
  - (Fractional) Immunization
  - Epidemic thresholds
- Conclusions

*Fractional Immunization of Networks*

B. Aditya Prakash, Lada Adamic, Theodore Iwashyna (M.D.), Hanghang Tong, Christos Faloutsos

SDM 2013, Austin, TX

# Whom to immunize?

- Dynamical Processes over networks

- Each circle is a hospital
- ~3,000 hospitals
- More than 30,000 patients transferred

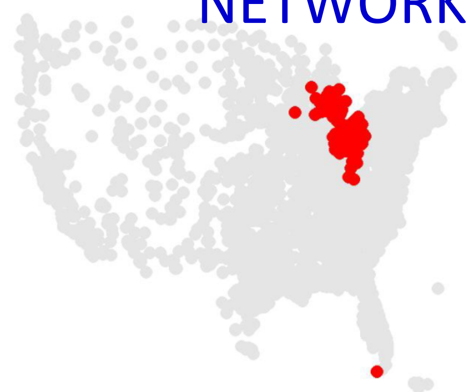
[US-MEDICARE NETWORK 2005]

**Problem**: Given *k* units of disinfectant, whom to immunize?

# Whom to immunize?

~6x fewer!

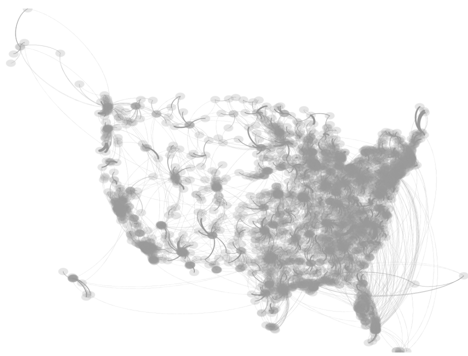
[US-MEDICARE NETWORK 2005]



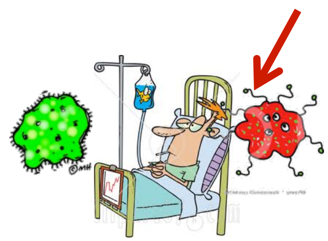
CURRENT PRACTICE

OUR METHOD

# Fractional Asymmetric Immunization



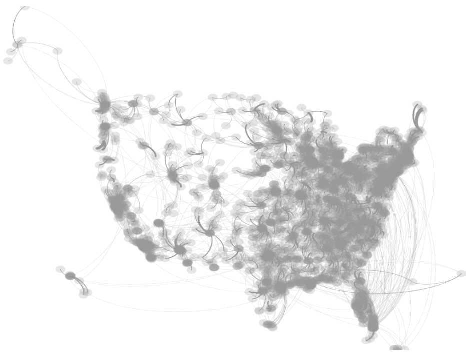
Drug-resistant Bacteria (like XDR-TB)

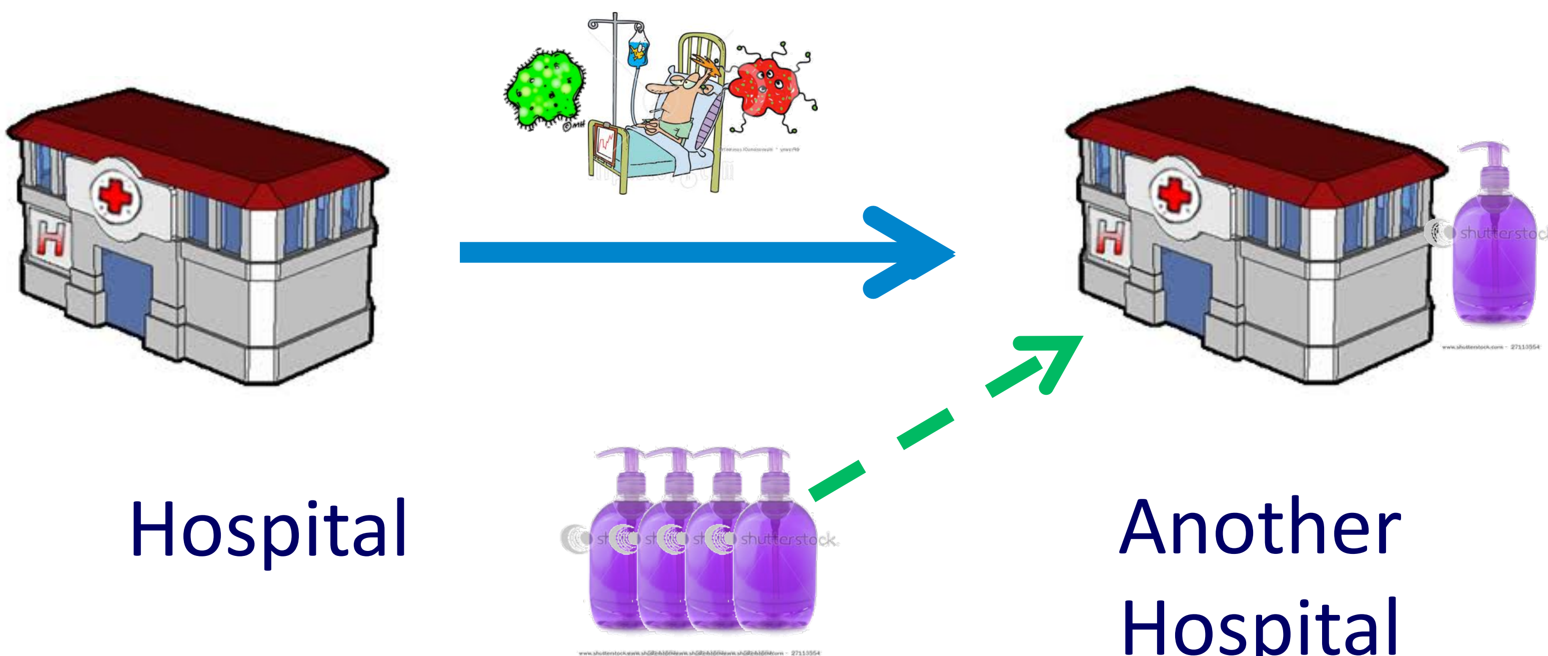


Hospital

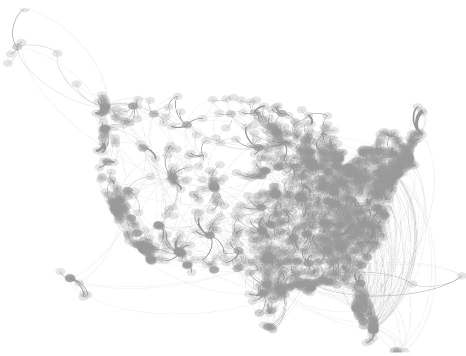
Another Hospital

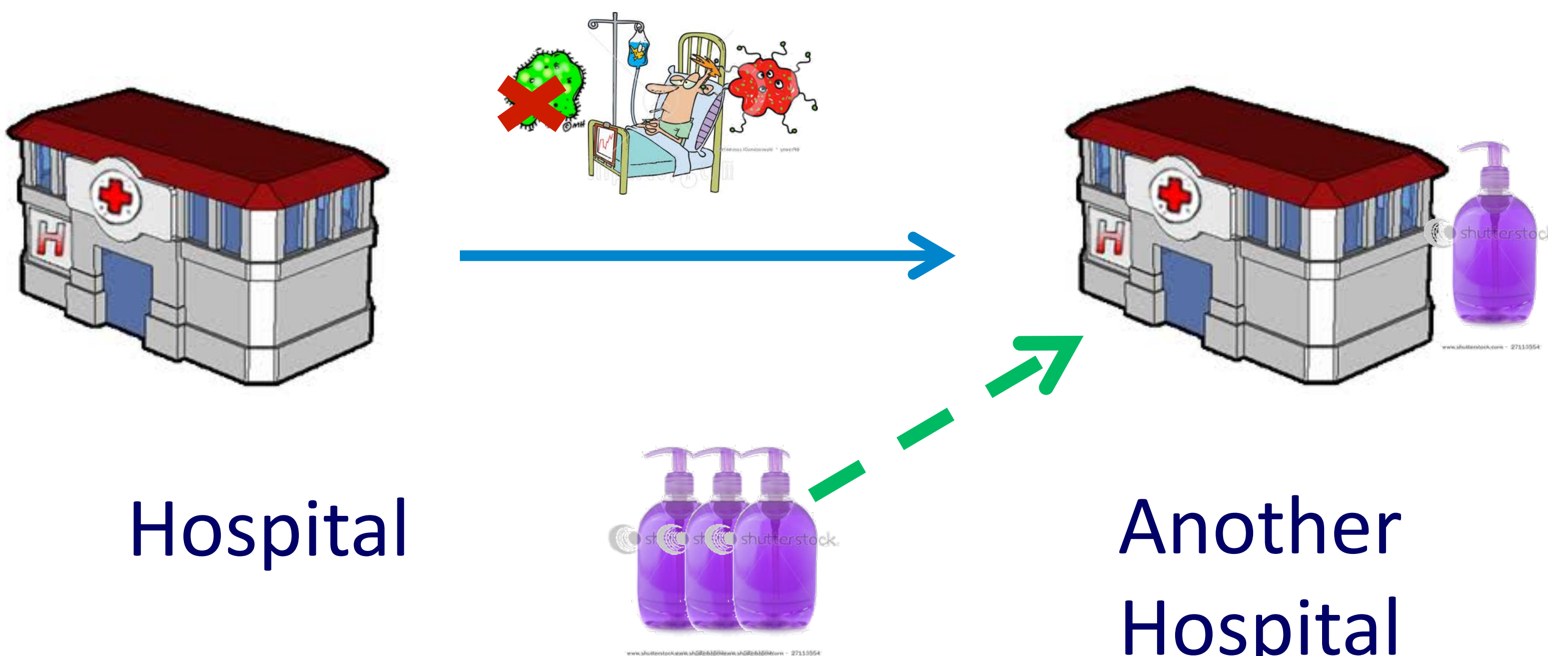
# Fractional Asymmetric Immunization



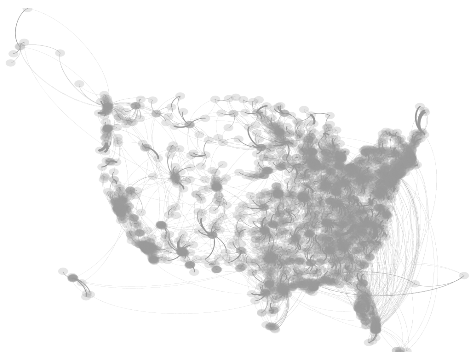


# Fractional Asymmetric Immunization

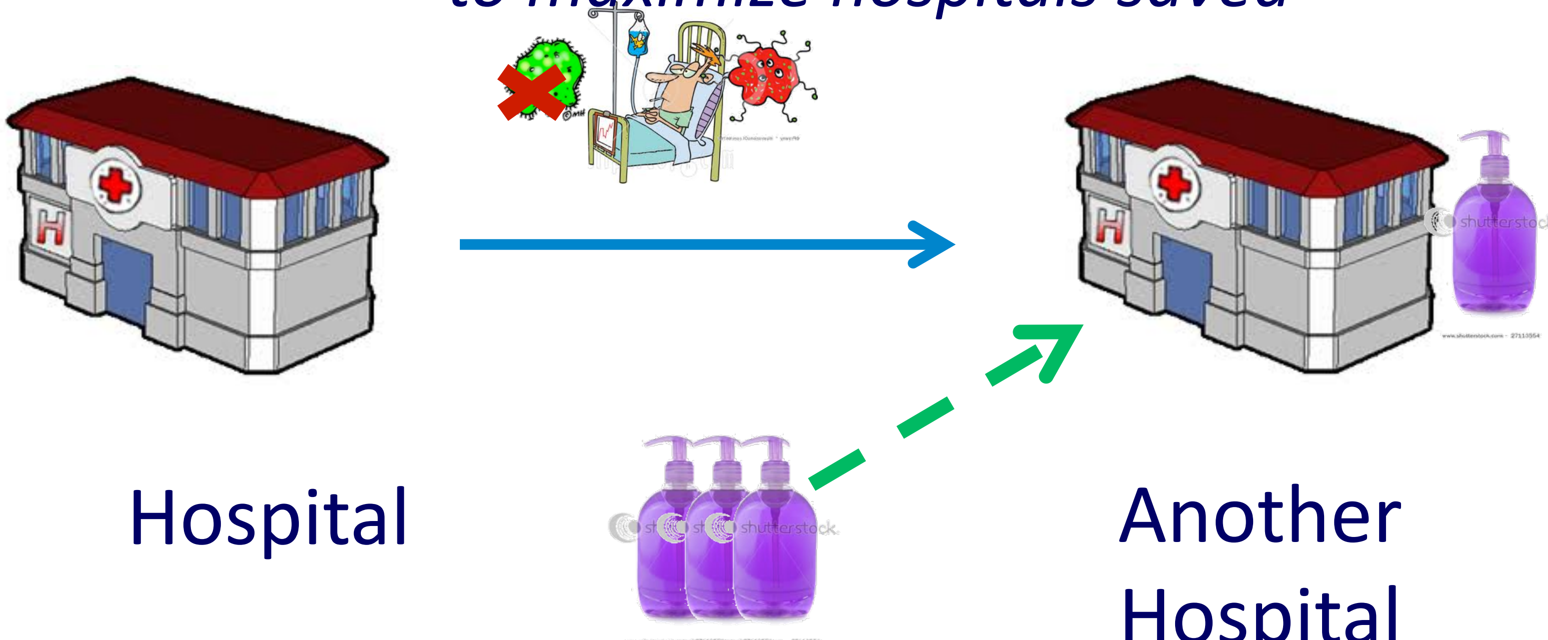




# Fractional Asymmetric Immunization



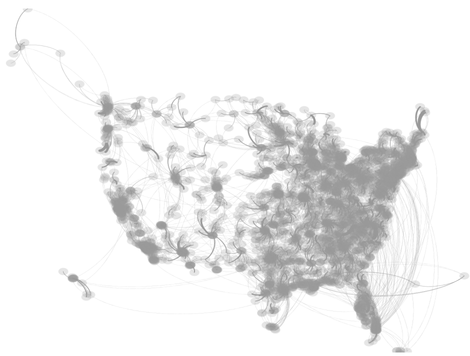
***Problem**:*
*Given k units of disinfectant,*
*distribute them*
*to maximize hospitals saved*



Hospital

Another Hospital

# Fractional Asymmetric Immunization

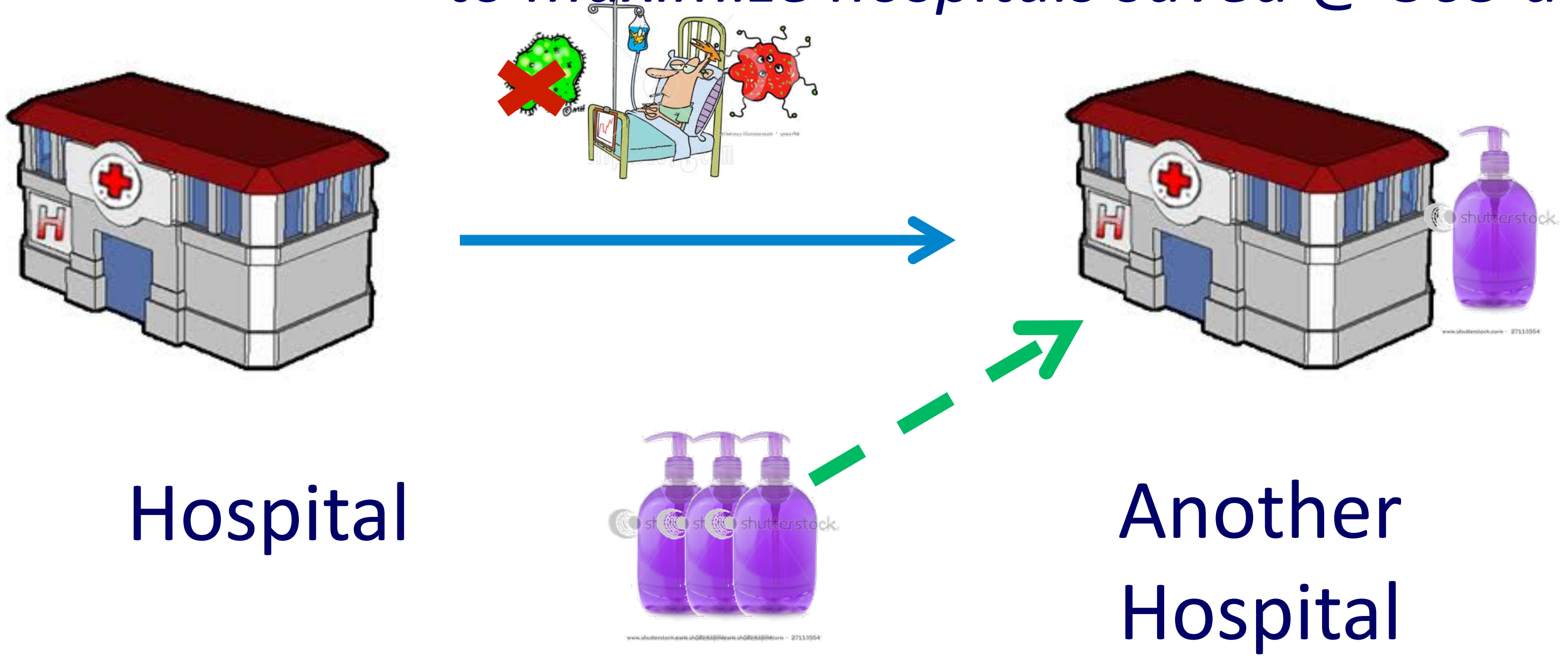


***Problem**:*

*Given k units of disinfectant,*

*distribute them*

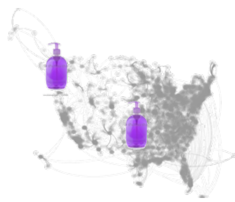
*to maximize hospitals saved @ 365 days*



# Straightforward solution:

Simulation:

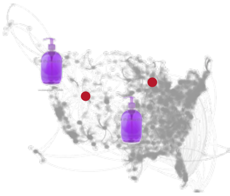
1. Distribute resources
2. ‘infect’ a few nodes
3. Simulate evolution of spreading
   – (10x, take avg)
4. Tweak, and repeat step 1



# Straightforward solution:

Simulation:

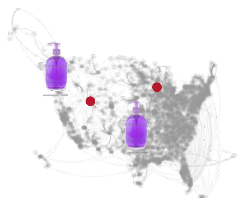
1. Distribute resources
2. ‘infect’ a few nodes
3. Simulate evolution of spreading
   – (10x, take avg)
4. Tweak, and repeat step 1



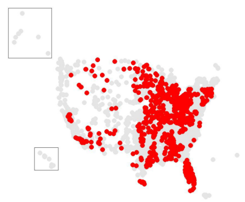
# Straightforward solution:

Simulation:

1. Distribute resources



2. ‘infect’ a few nodes
3. Simulate evolution of spreading
   – (10x, take avg)

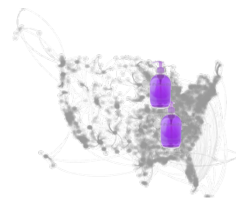


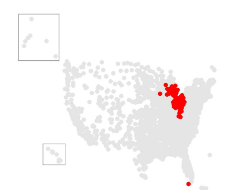
4. Tweak, and repeat step 1

# Straightforward solution:

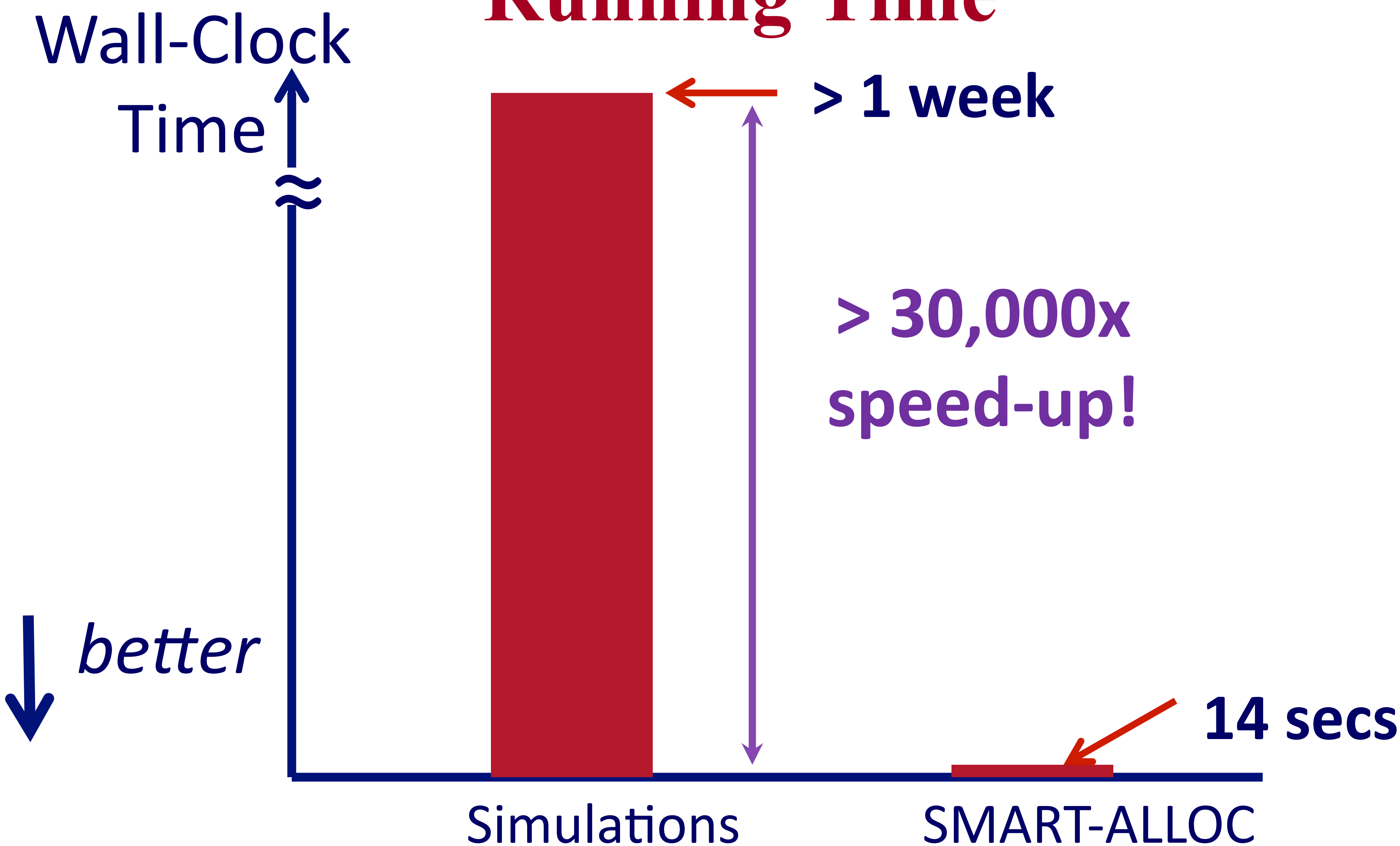
Simulation:

1. Distribute resources
2. 'infect' a few nodes
3. Simulate evolution of spreading
   – (10x, take avg)
4. Tweak, and repeat step 1

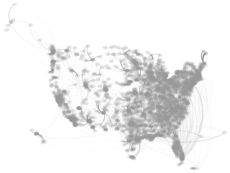




# Running Time

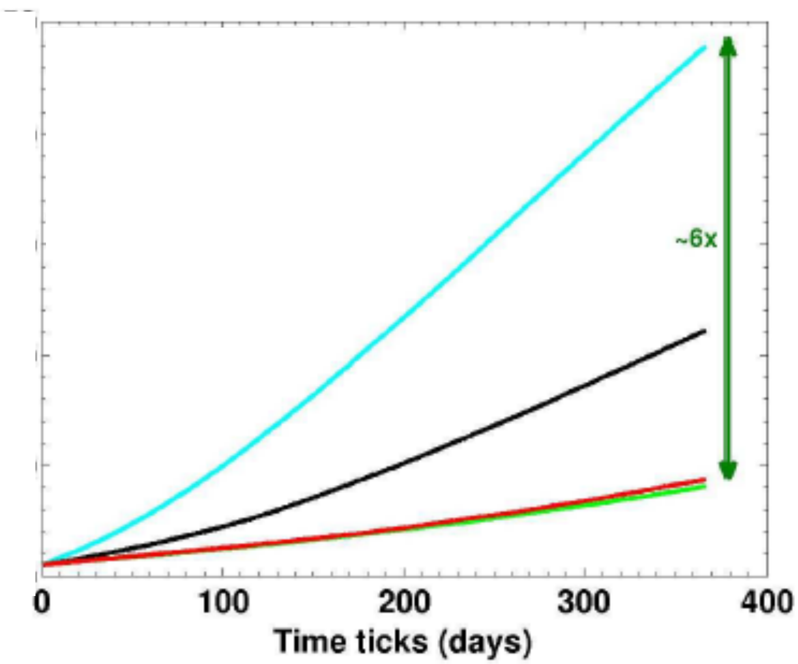


# Experiments



uniform

# infected




better

SMART-ALLOC

# epochs

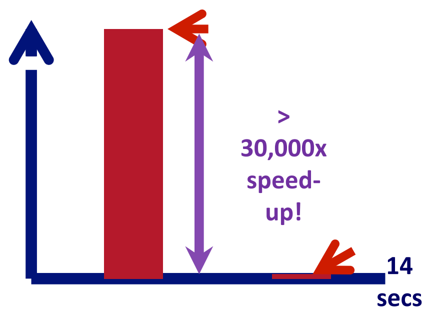
K = 120

# What is the 'silver bullet'?

A: Try to decrease connectivity of graph

Q: how to measure connectivity?

- Avg degree? Max degree?
- Std degree / avg degree ?
- Diameter?
- Modularity?
- 'Conductance' (~min cut size)?
- Some combination of above?



# What is the 'silver bullet'?

A: Try to decrease connectivity of graph

Q: how to measure connectivity?

A: first **eigenvalue** of adjacency matrix

Avg degree
Max degree
Diameter
Modularity
'Conductance'

Q1: why??

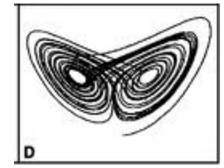
(Q2: dfn & intuition of eigenvalue ? )

# Why eigenvalue?

A1: ‘G2’ theorem and ‘**eigen-drop**’:

- For (almost) **any** type of virus
- For **any** network
- -> no epidemic, if small-enough first eigenvalue ($\lambda_1$) of *adjacency* matrix

*Threshold Conditions for Arbitrary Cascade Models on Arbitrary Networks*, B. Aditya Prakash, Deepayan Chakrabarti, Michalis Faloutsos, Nicholas Valler, Christos Faloutsos, ICDM 2011, Vancouver, Canada



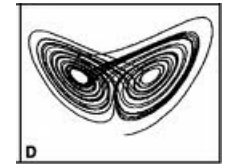
# Why eigenvalue?

A1: ‘G2’ theorem and ‘**eigen-drop**’:

- For (almost) **any** type of virus
- For **any** network
- -> no epidemic, if small-enough first eigenvalue ($\lambda_1$) of *adjacency* matrix

- Heuristic: for immunization, try to min $\lambda_1$
- The smaller $\lambda_1$, the closer to extinction.

# G2 theorem




*Threshold Conditions for Arbitrary Cascade Models on Arbitrary Networks*

B. Aditya Prakash, Deepayan Chakrabarti, Michalis Faloutsos, Nicholas Valler, Christos Faloutsos

IEEE ICDM 2011, Vancouver

extended version, in arxiv

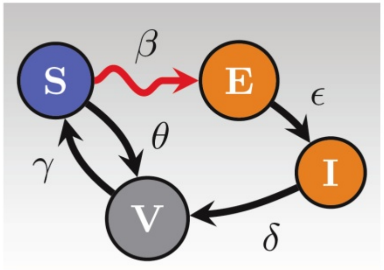
http://arxiv.org/abs/1004.0060

~10 pages proof

Carnegie Mellon

# Our thresholds for some models

- *s = effective strength*
- *s < 1 : below threshold*

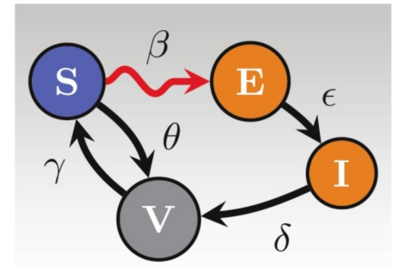



| Models | Effective Strength (s) | Threshold (tipping point) |
|---|---|---|
| SIS, SIR, SIRS, SEIR | $s = \lambda \left(\frac{\beta}{\delta}\right)$ | $s = 1$ |
| SIV, SEIV | $s = \lambda . \left(\frac{\beta\gamma}{\delta(\gamma+\theta)}\right)$ | |
| $SI_1I_2V_1V_2$ (H.I.V.) | $s = \lambda . \left(\frac{\beta_1 v_2 + \beta_2 \varepsilon}{v_2(\varepsilon + v_1)}\right)$ | |

Carnegie Mellon

# Our thresholds for some models

- *$s$ = effective strength*
- *$s < 1$ : below threshold*



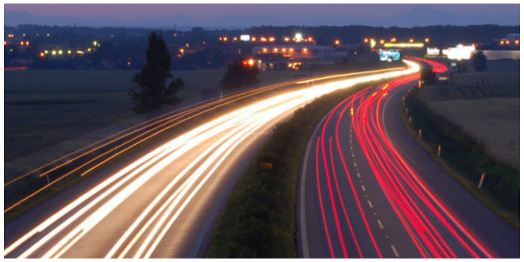

No immunity

Temp. immunity

w/ incubation

| | e Strength | Threshold (tipping point) |
|---|---|---|
| SIS, SIR, SIRS, SEIR | $s = \lambda \left(\frac{\beta}{\delta}\right)$ | $s = 1$ |
| SIV, SEIV | $s = \lambda \,.\left(\frac{\beta\gamma}{\delta(\gamma+\theta)}\right)$ | |
| $SI_1I_2V_1V_2$ (**H.I.V.**) | $s = \lambda \,.\left(\frac{\beta_1 v_2 + \beta_2 \varepsilon}{v_2(\varepsilon + v_1)}\right)$ | |

# Roadmap



- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: Cascade analysis
  - (Fractional) Immunization
  - intuition behind $\lambda_1$
- Conclusions

# Intuition for λ

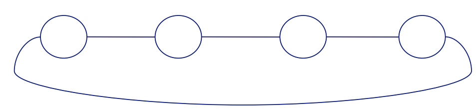
## "Official" definitions:

- *Let **A** be the adjacency matrix. Then λ is the root with the largest magnitude of the characteristic polynomial of **A** [det(**A** – λ**I**)].*
- Also: $\mathbf{A}\ \mathbf{x} = \lambda\ \mathbf{x}$

Neither gives much intuition!

## "Un-official" Intuition

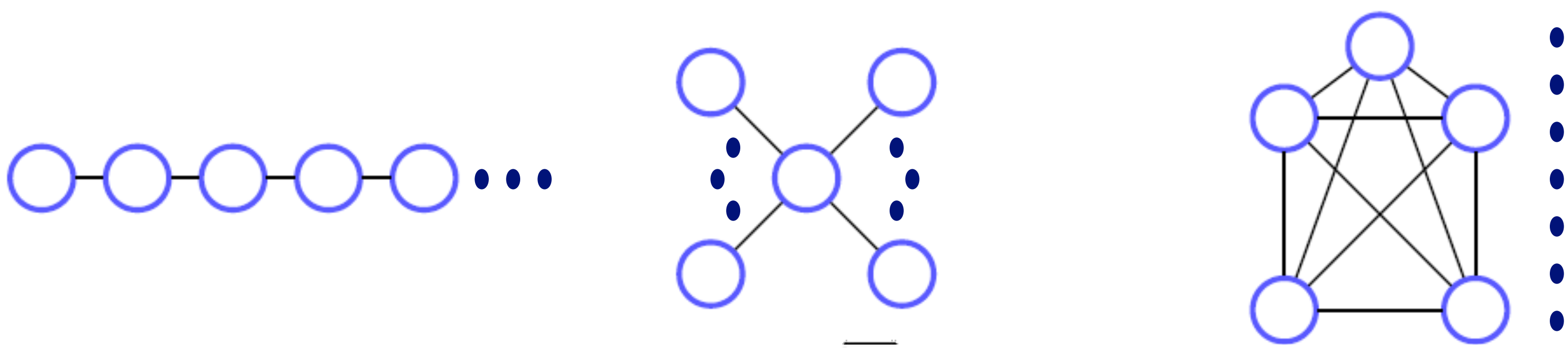
- For 'homogeneous' graphs, *λ* == *degree*



- *λ* ~ avg degree
  - done right, for skewed degree distributions

# Largest Eigenvalue (λ)

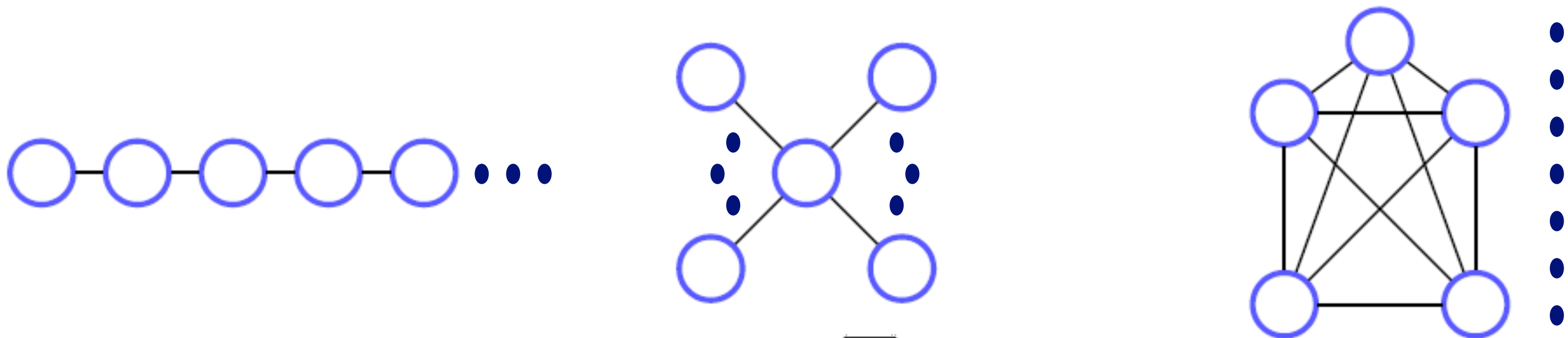
better connectivity ⟶ higher $\lambda$




*N* = 1000 nodes

# Largest Eigenvalue (λ)

better connectivity ⟶ higher $\lambda$



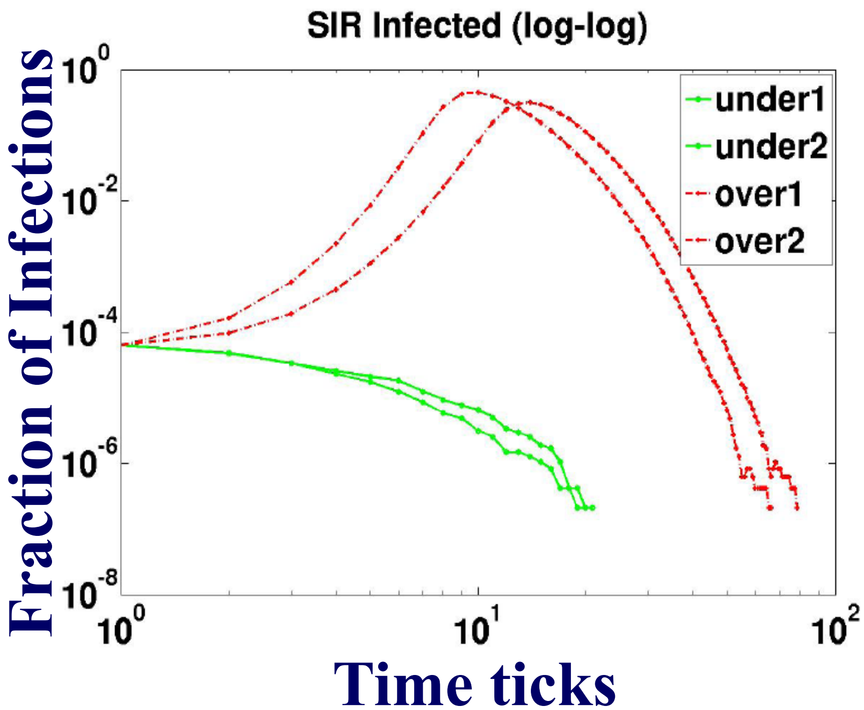
$\lambda \approx 2$ | $\lambda = \sqrt{N}$ | $\lambda = N\text{-}1$

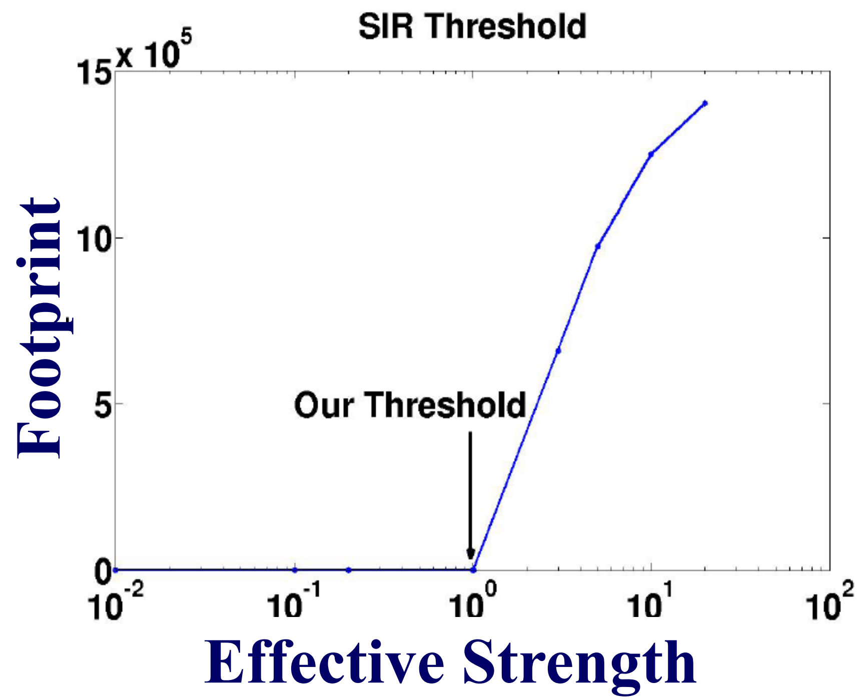
(a)Chain | (b)Star | (c)Clique

$\lambda \approx 2$ | $\lambda = 31.67$ | $\lambda = 999$

$N$ = 1000 nodes

# Examples: Simulations – SIR (mumps)



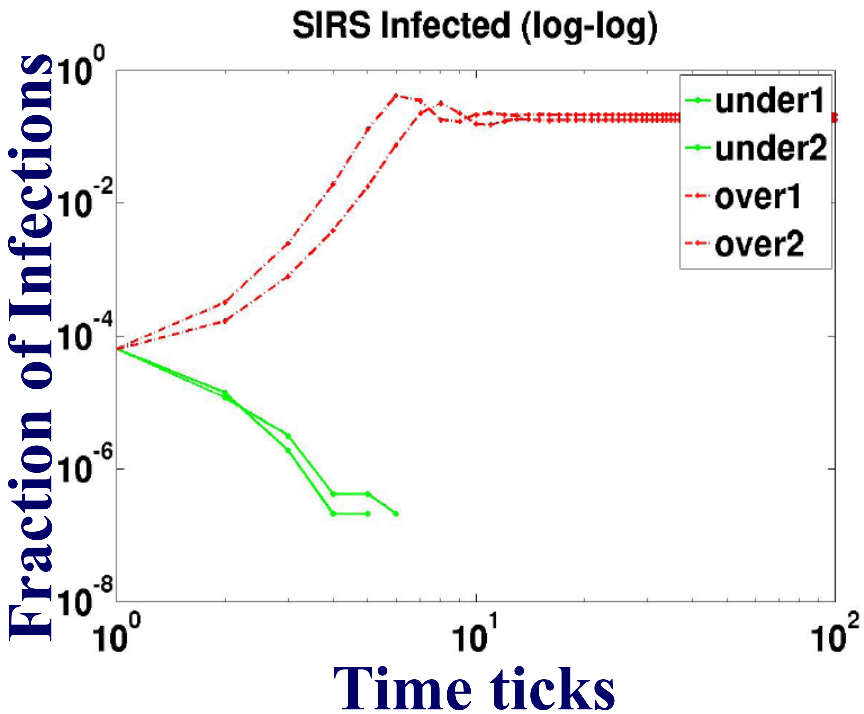



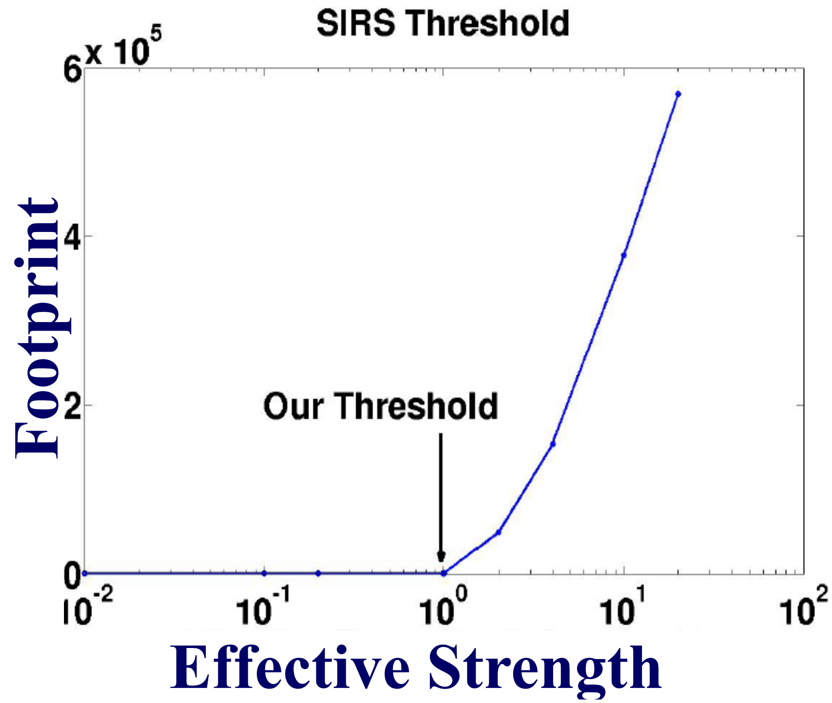

**(a) Infection profile** **(b) "Take-off" plot**

PORTLAND graph: *synthetic population,*

*31 million links, 6 million nodes*

# Examples: Simulations – SIRS (pertusis)




**(a) Infection profile** **(b) "Take-off" plot**

PORTLAND graph: *synthetic population,*

*31 million links, 6 million nodes*

# Immunization - conclusion

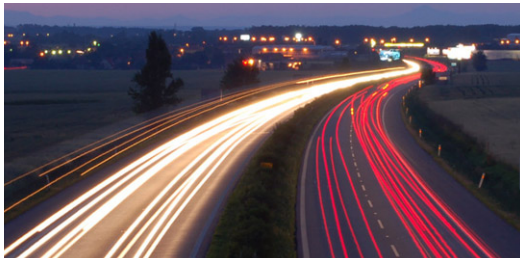
In (**almost any**) immunization setting,

- Allocate resources, such that to
- **Minimize** $\boldsymbol{\lambda_1}$
- (*regardless* of virus specifics)

- Conversely, in a market penetration setting
  - Allocate resources to
  - Maximize $\boldsymbol{\lambda_1}$

Carnegie Mellon

# Roadmap



- Introduction – Motivation
- Part#1: Patterns in graphs
- Part#2: Cascade analysis
  - (Fractional) Immunization
  - Epidemic thresholds
- ➡ Acks & Conclusions

# 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

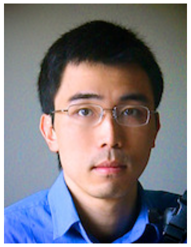
www.cs.cmu.edu/~pegasus

Results on large graphs: with Pegasus + hadoop + M45

Apache license

Code, papers, manual, video

Prof. U Kang

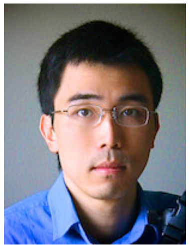


Prof. Polo Chau

# Cast

Akoglu, Leman

Beutel, Alex



Chau, Polo

Kang, U

Koutra, Danai

McGlohon, Mary

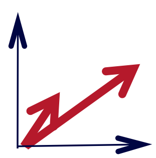
Prakash, Aditya

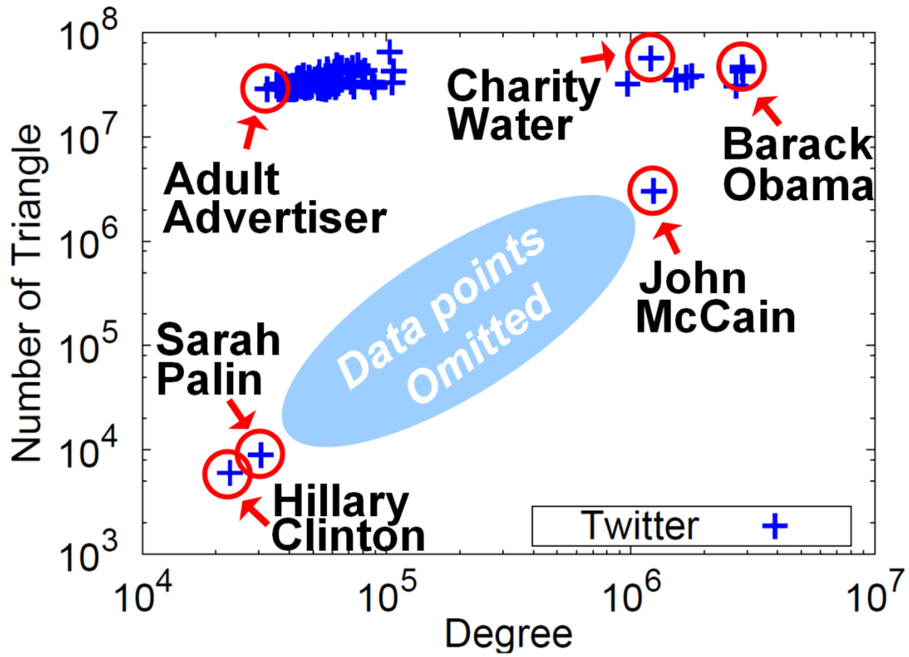
Papalexakis, Vagelis

Tong, Hanghang

# CONCLUSION#1 – Big data

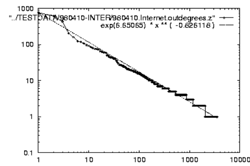
- **Large** datasets reveal patterns/outliers that are invisible otherwise

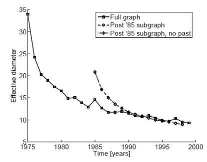


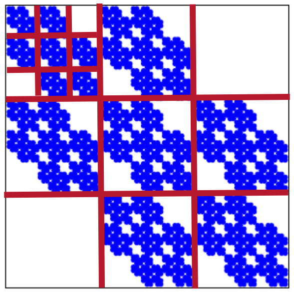


# CONCLUSION#2 – self-similarity

- powerful tool / viewpoint
  - Power laws; shrinking diameters
  - **Gaussian trap** (eg., F.O.F.)
  - RMAT – `graph500` generator

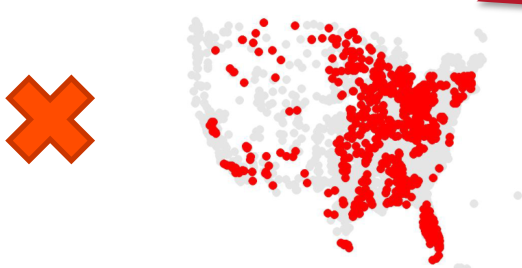


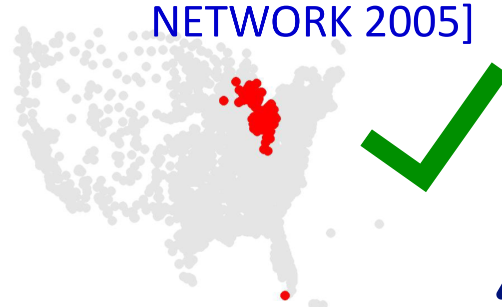



# CONCLUSION#3 – eigen-drop

- Cascades & immunization: G2 theorem & **eigenvalue**

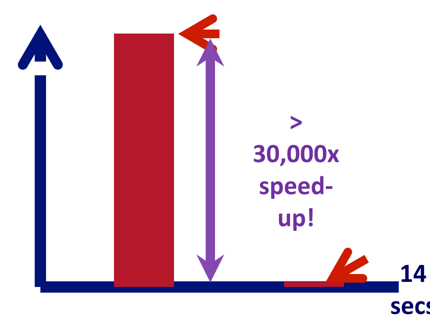



[US-MEDICARE NETWORK 2005]



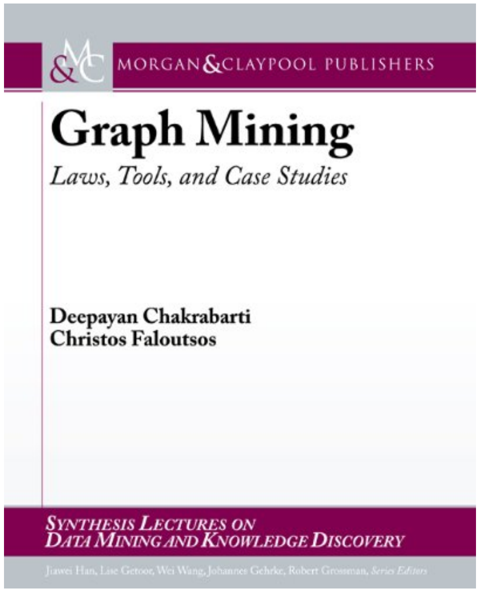
CURRENT PRACTICE

OUR METHOD



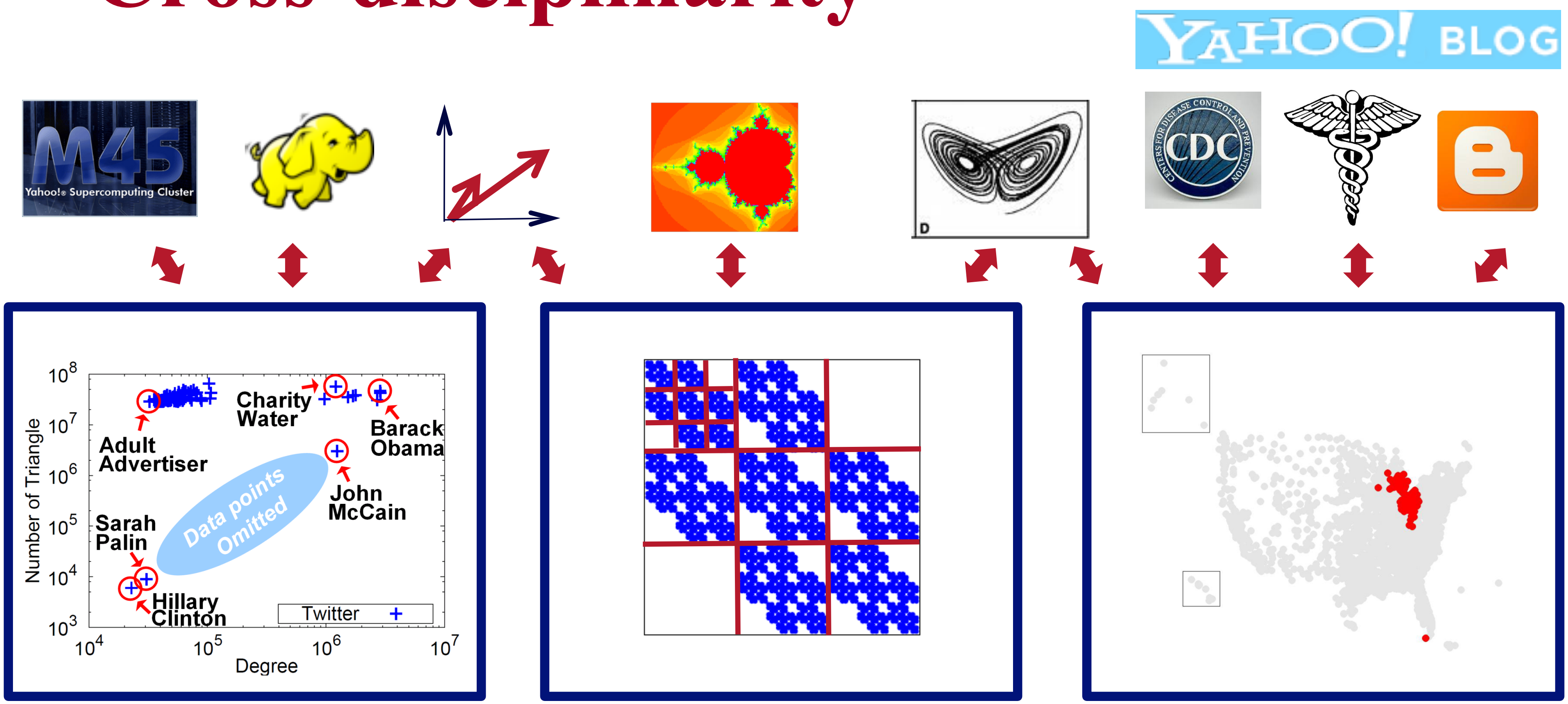
# References


- D. Chakrabarti, C. Faloutsos: *Graph Mining – Laws, Tools and Case Studies*, Morgan Claypool 2012
- http://www.morganclaypool.com/doi/abs/10.2200/S00449ED1V01Y201209DMK006

# TAKE HOME MESSAGE:

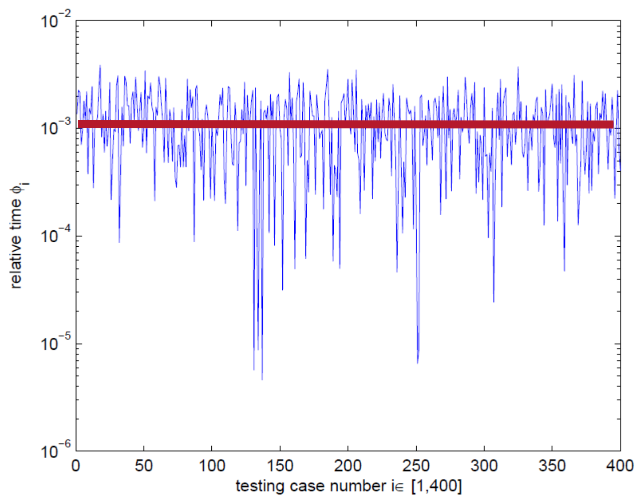
## Cross-disciplinarity



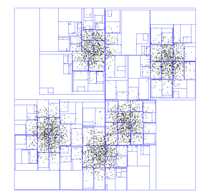
# Already started paying off for power grids

- Same accuracy, **100x – 100K x faster**

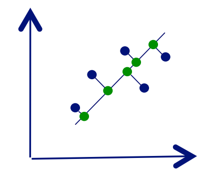
**1000 x**




Kd-tree



SVD

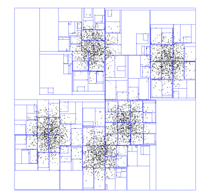


**[1] Yang Weng, Christos Faloutsos, Marija D. Ili´c, and Rohit Negi, Speed up of Data-Driven State Estimation Using Low-Complexity Indexing Method, IEEE PES-General Meeting, (accepted), 2014**

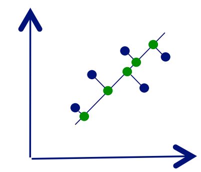
# THANK YOU!

- Same accuracy, **100x – 100K x faster**

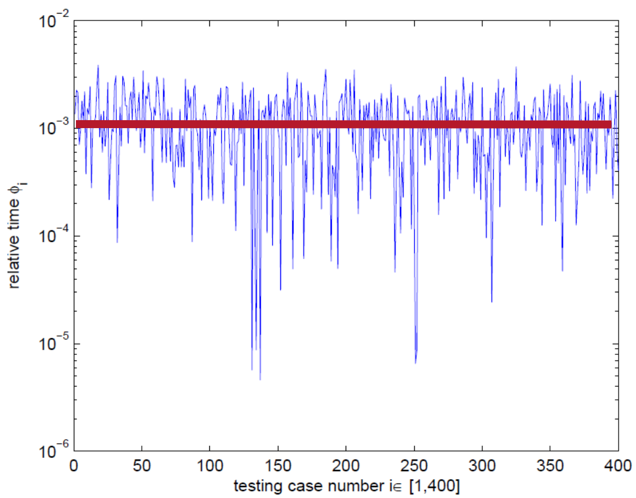
Kd-tree



SVD



**1000 x**




**[1] Yang Weng, Christos Faloutsos, Marija D. Ili´c, and Rohit Negi, Speed up of Data-Driven State Estimation Using Low-Complexity Indexing Method, IEEE PES-General Meeting, (accepted), 2014**