Complex Network, Fractal and Kronecker Graph

Yang Lu

Preface

Nature and nature 's law hid in night

God said , let Newton be!

And all was light .

—Alexander Pope

Question 1

- Can we research Climate Prediction by tracing the water drop?
- Can we research Brain Mechanism by studying the neuron?
- Can we research Secret of Lives by studying microbe?
- Why?

Answer

- Of course NO!!!
- Why
  1. Whole is not only the accumulation of Parts
  2. Miracles emerge when scale up
     - Brain
  3. Philosophical Perspective: quantitative changes vs. qualitative changes

Question 2

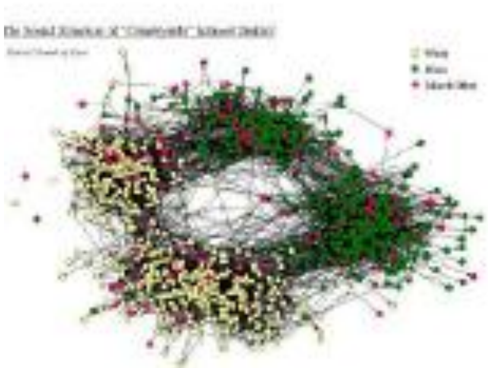
- Why shouldn’t we start up to study the Macro situations?
  - Limitation of knowledge
  - Limitation of Horizon of Sight
  - Limitation of Ambition
- Actually top Scientists are on the way!!

Ant? Hawk?

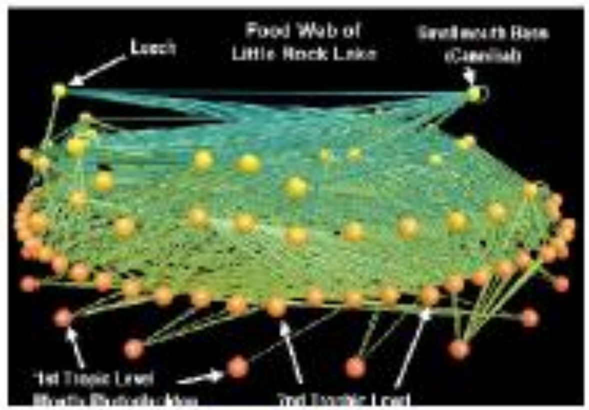
How to start up ?

- Complex Network theory
- Fractal
- Application: Kronecker Graph

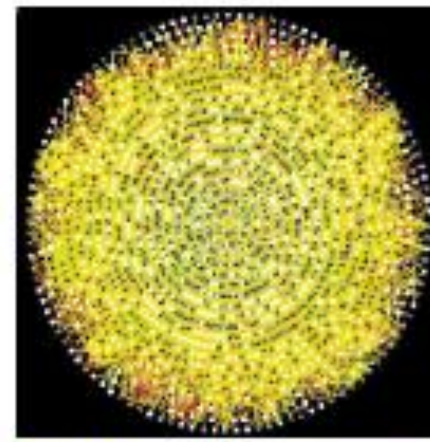
Networks Everywhere!



Friendship Network
[Moody '01]



Food Web
[Martinez '91]



Protein Interactions
[genomebiology.com]

Graphs are everywhere!

Application Widespread!

- Social Network
  - Friends Recommendation
  - Potential Customers to market-to
- Epidemic Immunization
  - Control the spread of virus
- Etc.

Statistical properties of networks

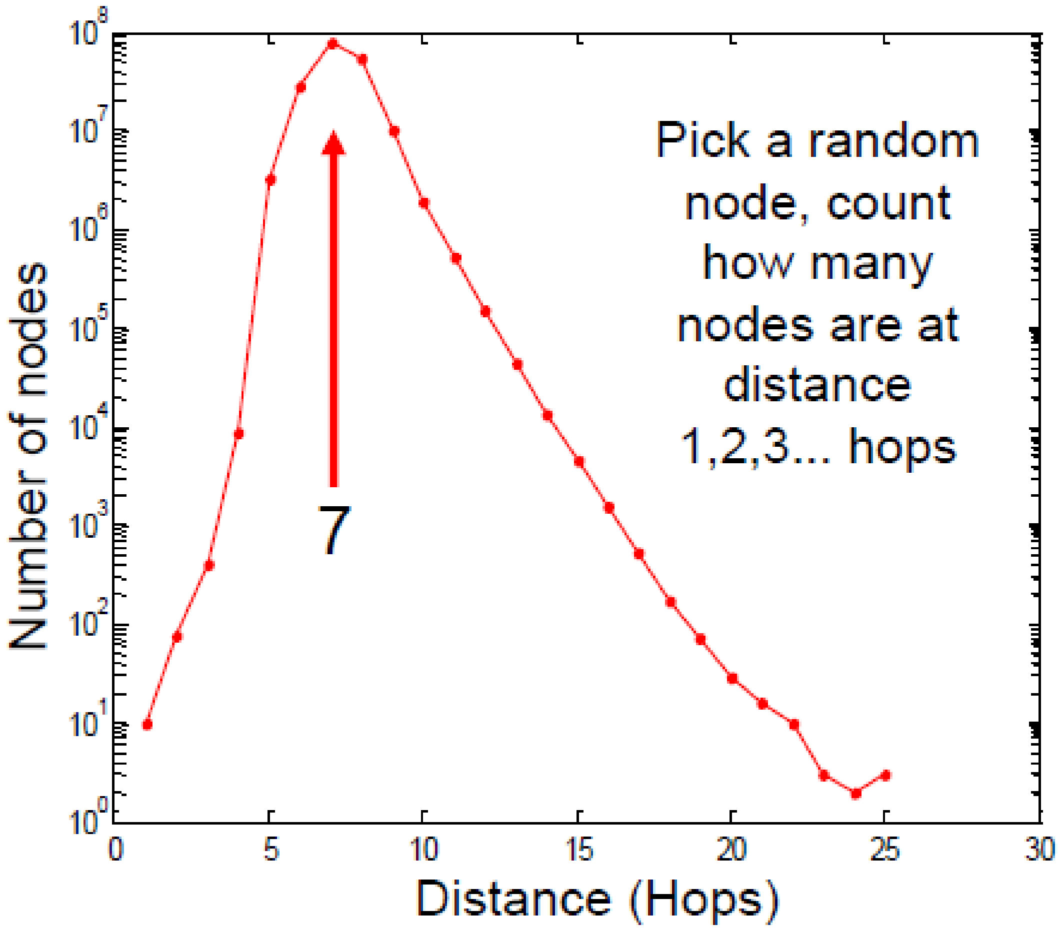
- Features that are common to networks of different types:
  - Properties of static networks:
    - Small-world effect
    - Transitivity or clustering
    - Degree distributions (scale free networks)
    - Network resilience
    - Community structure
    - Subgraphs or motifs
  - Temporal properties:
    - Densification
    - Shrinking diameter

Small-world effect (1)

- Six degrees of separation (Milgram 60s)
  - Random people in Nebraska were asked to send letters to Stockbrokes in Boston
  - Only 25% letters reached the goal
  - But they reached it in about 6 steps
- Measuring path lengths:
  - Diameter (longest shortest path): $max\ d_{ij}$
  - Effective diameter: distance at which 90% of all connected pairs of nodes can be reached
  - Mean geodesic (shortest) distance l $\ell = \frac{1}{\frac{1}{2}n(n+1)} \sum_{i \geq j} d_{ij}$

Small-world effect (2)

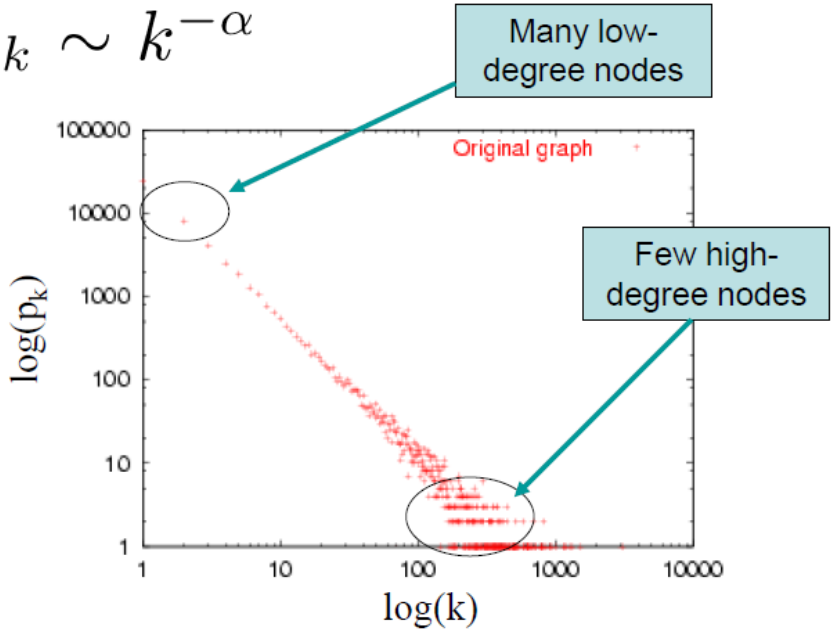
- Distribution of shortest path lengths



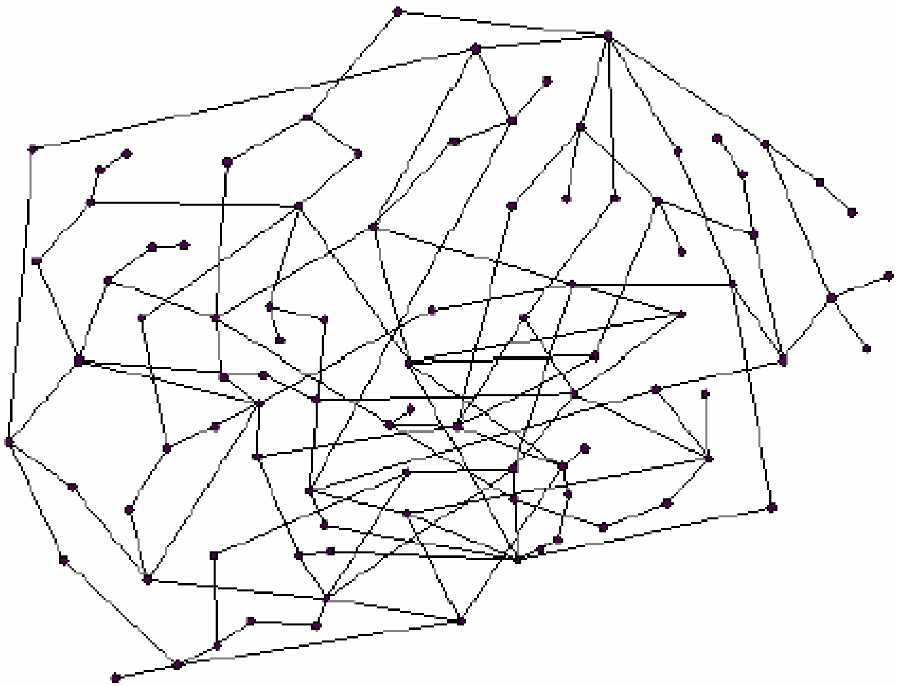
Degree distributions

- Erdos-Renyi Random Network
  - The probability of Edge E_{ij} exists is constant p
  - Poisson Distribution
- Scale-free Network
  - $p_k \sim k^{-\alpha}$
  - Power-law Distribution

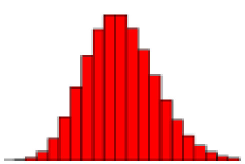
$$p_k \sim k^{-\alpha}$$



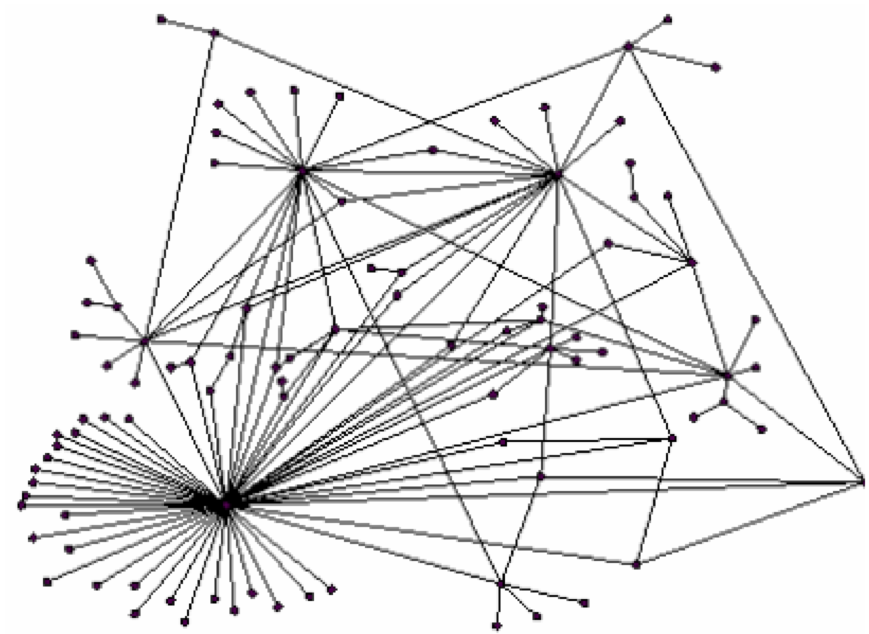
Poisson vs. Scale-free network



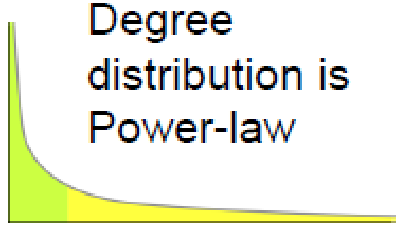
Poisson network
(Erdos-Renyi random graph)



Degree distribution is Poisson



Scale-free (power-law) network



Degree distribution is Power-law

Temporal Graph Patterns

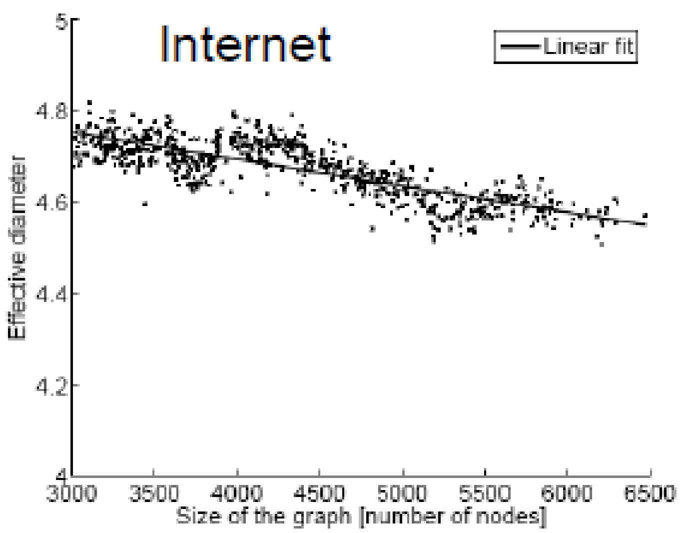
- Conventional Wisdom:
  - Constant average degree: the number of edges grows linearly with the number of nodes
  - Slowly growing diameter: as the network grows the distances between nodes grow
- Actually:
  - Densification Power Law: networks are becoming denser over time
  - Shrinking Diameter: diameter is decreasing as the network grows

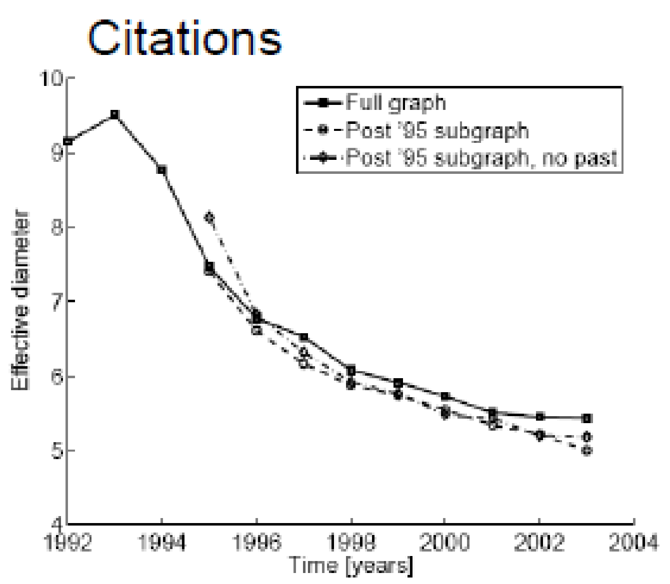
Densification

- What is the relation between the number of nodes and the number of edges in a network?
- Densification Power Law
  - $E(t)$ means edges at time t
  - $N(t)$ means edges at time t
- Suppose $N(t+1) = 2 * N(t)$, then $E(t+1) > 2 * E(t)$
- But still obey Power-Law
- $E(t) \propto N(t)^{\alpha}, 1 \leq \alpha \leq 2$

Shrinking diameters

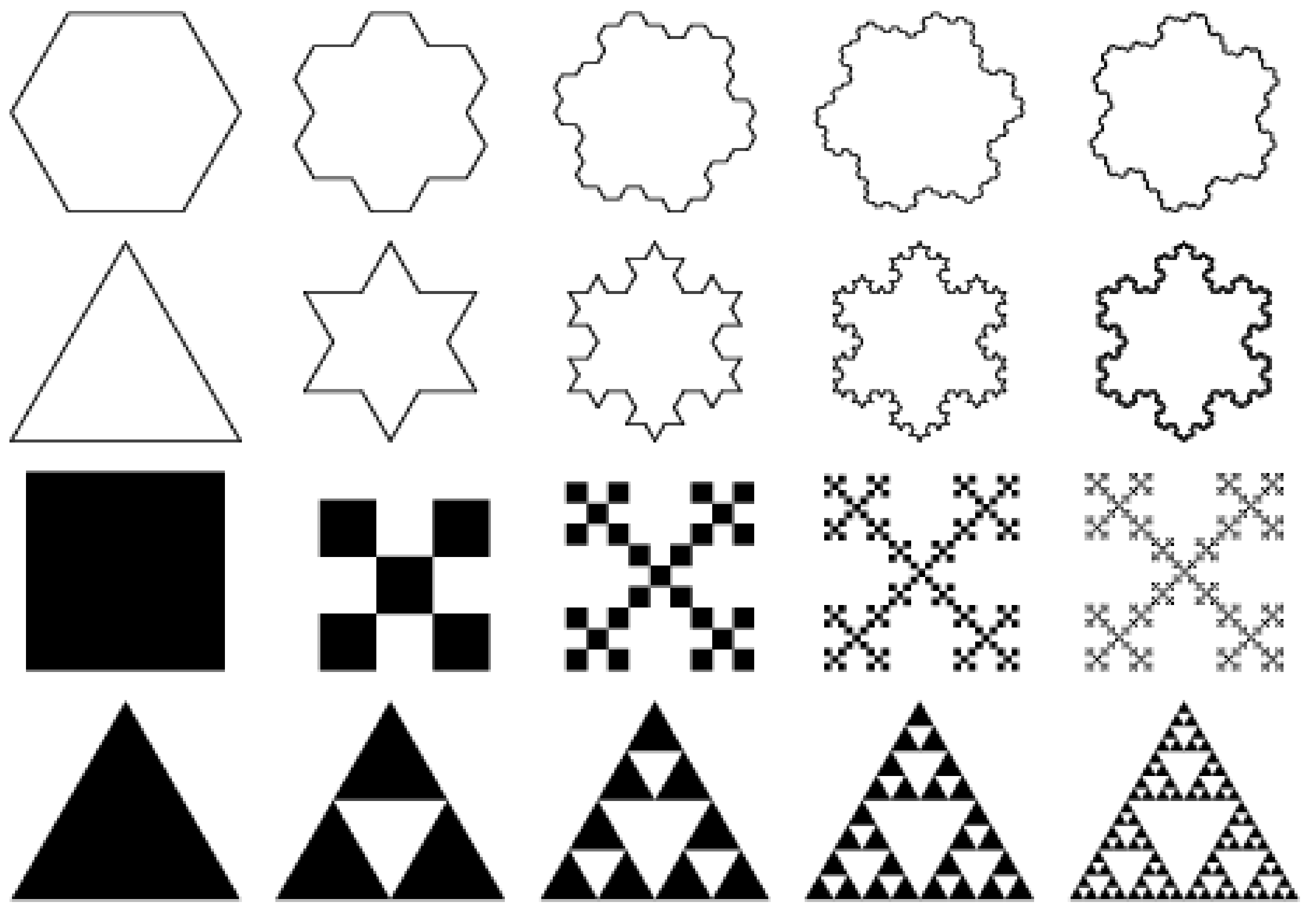
- Diameter Shrinks/Stabilizes over time!
  - as the network grows the distances between nodes slowly decrease





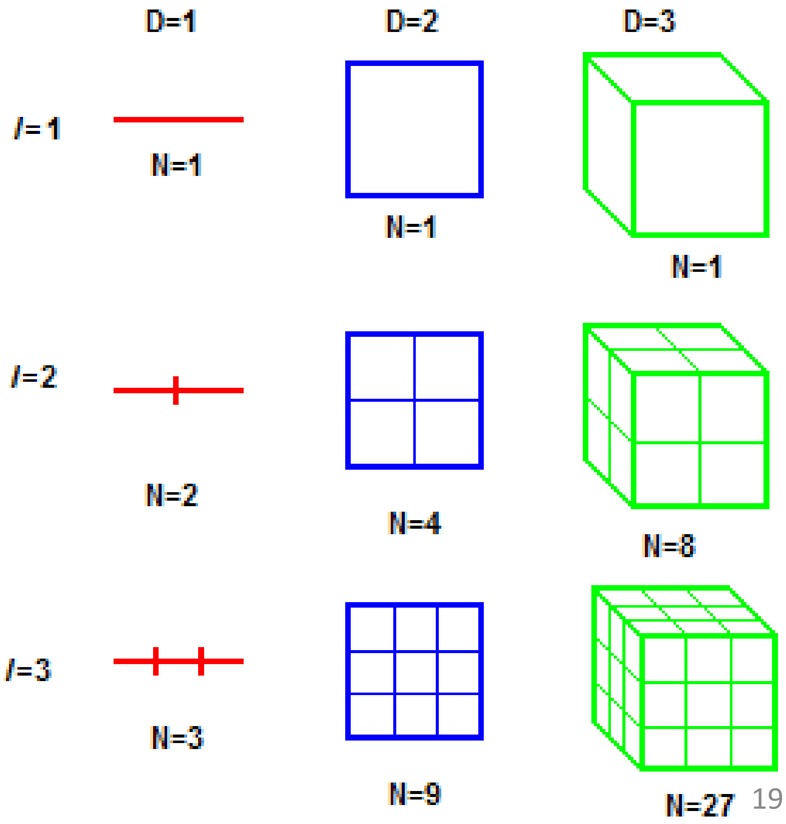
What is Fractal?

- Self-Similar



Fractal Dimension

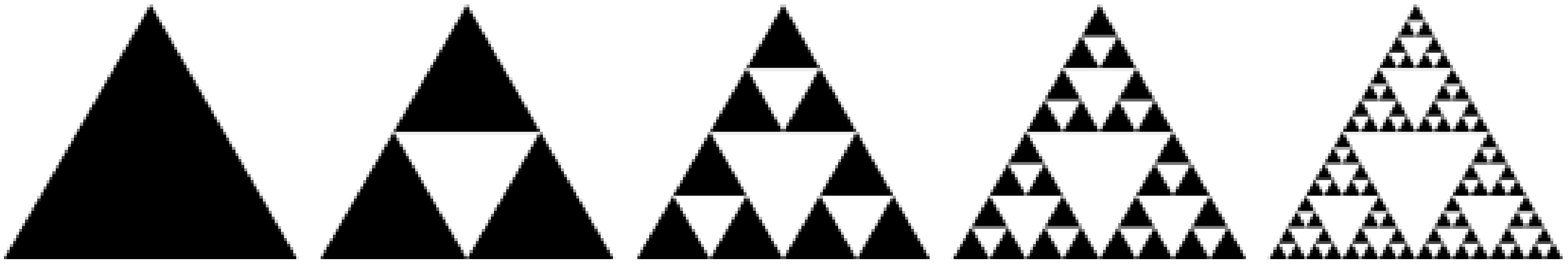
- Scaling Rule Equation: $N \propto \epsilon^D$:
  - N is number of new sticks
  - ϵ is the scaling factor
  - D is the dimension
- Thus: $D = \frac{\log N}{\log \epsilon}$



Example

Q:What is the dimensionality of Sierpinski triangle?

A: $^{\log 3}/_{\log 2}$



Network vs. Fractal?

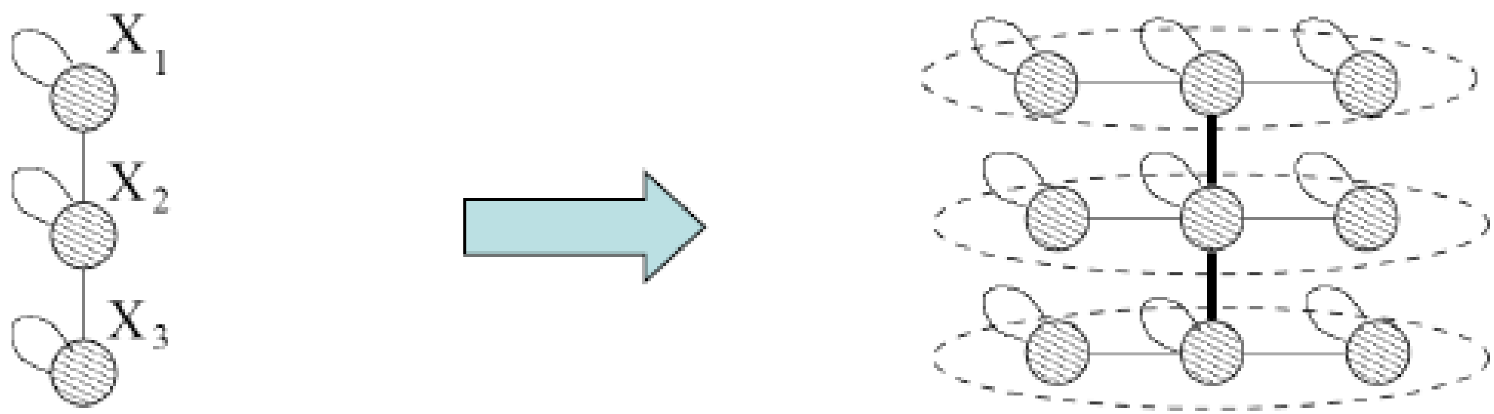
- What's the relationship between Reality Network and Fractal ?
- Graph Generation Model
  - Kronecker graphs

Kronecker graphs

- Want to have a model that can generate a realistic graph:
  - Static Patterns
    - Power Law Degree Distribution
    - Small Diameter
    - Power Law Eigenvalue and Eigenvector Distribution
  - Temporal Patterns
    - Densification Power Law
    - Shrinking/Constant Diameter

Idea: Recursive graph generation

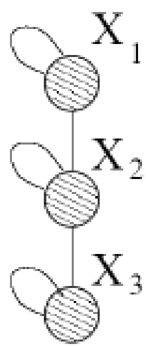
- Intuition: self-similarity leads to power-laws
- Try to mimic recursive graph/community growth
- There are many obvious (but wrong) ways:
- Kronecker Product is a way of generating self-similar matrices

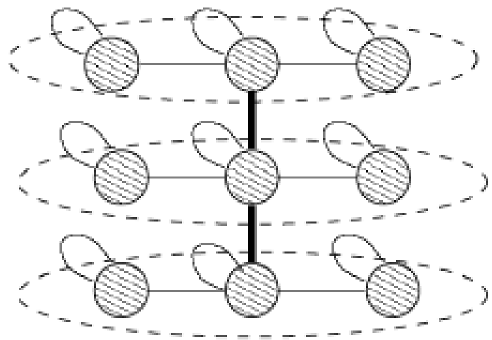


Initial graph

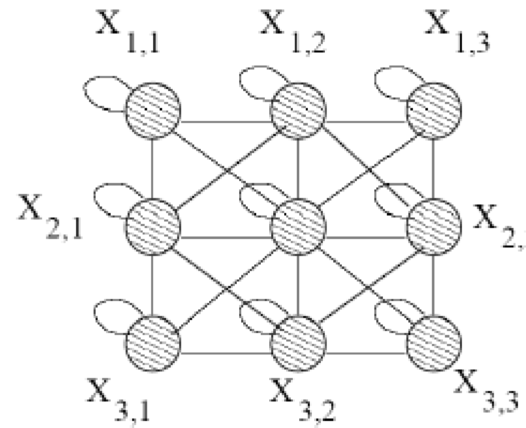
Recursive expansion

Kronecker product: Graph





Intermediate stage



1	1	0
1	1	1
0	1	1

(3x3)

G_1

Adjacency matrix

G_1	G_1	0
G_1	G_1	G_1
0	G_1	G_1

(9x9)

$$G_2 = G_1 \otimes G_1$$

Adjacency matrix

Kronecker product: Definition

- The Kronecker product of matrices A and B is given by
- We define a Kronecker product of two graphs as a Kronecker product of their adjacency matrices

$$\underset{N\,x\,M}{\mathbf{C} = \mathbf{A}} \otimes \underset{K\,x\,L}{\mathbf{B}} \doteq \begin{pmatrix} a_{1,1}\mathbf{B} & a_{1,2}\mathbf{B} & \dots & a_{1,m}\mathbf{B} \\ a_{2,1}\mathbf{B} & a_{2,2}\mathbf{B} & \dots & a_{2,m}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}\mathbf{B} & a_{n,2}\mathbf{B} & \dots & a_{n,m}\mathbf{B} \end{pmatrix}$$

$N{*}K\; x\; M{*}L$

Kronecker Graph

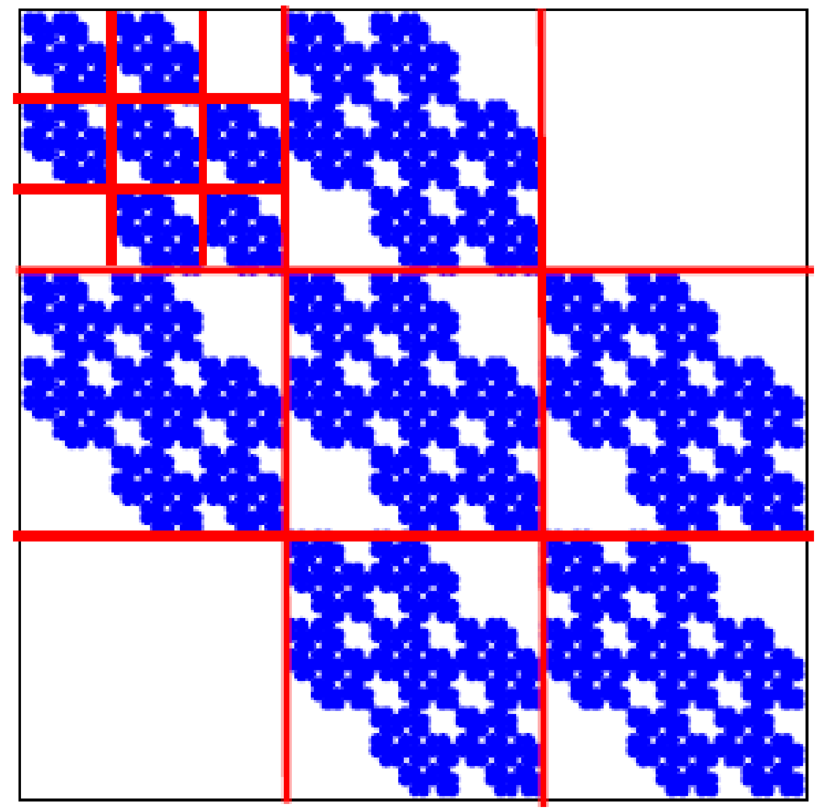
- We create the self-similar graphs recursively
  - Start with a initiator graph G_1 on N_1 nodes and E_1 edges
  - The recursion will then product larger graphs $G_2, G_3, \ldots, G_k$ on N_k nodes
- We obtain a growing sequence of graphs by iterating the Kronecker product

$$G_k = \underbrace{G_1 \otimes G_1 \otimes \ldots G_1}_{k\ times}$$

Kronecker product: Graph

1	1	0
1	1	1
0	1	1

G_1



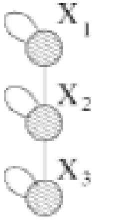
G_4 adjacency matrix

Question

- Given the real graph G
- How do we choose the parameters to match all of these at once?

Model estimation: approach

- Maximum likelihood estimation
  - Given real graph G
  - Estimate Kronecker initiator graph Θ (e.g., $\begin{bmatrix}1&1&0\\1&1&1\\0&1&1\end{bmatrix}$) which



$$\arg\max_{\Theta} P(G \mid \Theta)$$

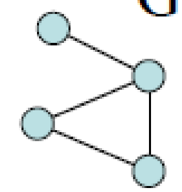
- We need to (efficiently) calculate

$$P(G \mid \Theta)$$

- And maximize over Θ (e.g., using gradient descent)

Fitting Kronecker graphs

G



- Given a graph G and Kronecker matrix Θ we calculate probability that Θ generated G $P(G|\Theta)$

0.5	0.2
0.1	0.3

Θ

0.25	0.10	0.10	0.04
0.05	0.15	0.02	0.06
0.05	0.02	0.15	0.06
0.01	0.03	0.03	0.09

Θ_k

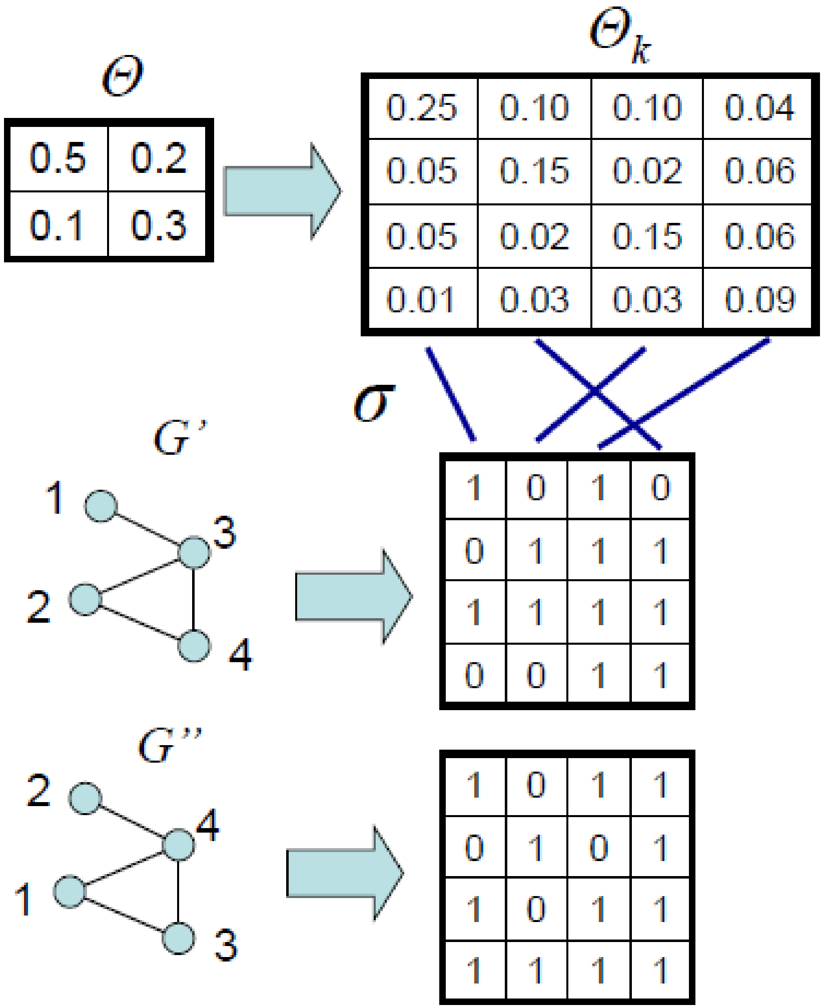
1	0	1	1
0	1	0	1
1	0	1	1
1	1	1	1

G

$P(G|\Theta)$

$$P(G \mid \Theta) = \prod_{(u,v)\in G} \Theta_k[u,v] \prod_{(u,v)\notin G} (1-\Theta_k[u,v])$$

Challenge 1: Node correspondence



$P(G'|\Theta) = P(G''|\Theta)$

- Nodes are unlabeled
- Graphs G' and G'' should have the same probability $P(G'|\Theta) = P(G''|\Theta)$
- One needs to consider all node correspondences σ

$$P(G \mid \Theta) = \sum_{\sigma} P(G \mid \Theta, \sigma) P(\sigma)$$

- All correspondences are a priori equally likely
- There are $O(N!)$ correspondences

Challenge 2: calculating P(G|Θ,σ)

- Assume we solved the correspondence problem
- Calculating

$$P(G \mid \Theta) = \prod_{(u,v)\in G} \Theta_k[\sigma_u, \sigma_v] \prod_{(u,v)\notin G} (1 - \Theta_k[\sigma_u, \sigma_v])$$

σ… node labeling

- Takes $O(N^2)$ time
- Infeasible for large graphs ($N \sim 10^5$)

0.25	0.10	0.10	0.04
0.05	0.15	0.02	0.06
0.05	0.02	0.15	0.06
0.01	0.03	0.03	0.09

Θ_{kc}

σ

1	0	1	1
0	1	0	1
1	0	1	1
0	0	1	1

G

$P(G|\Theta, \sigma)$

Model estimation: solution

- Naïvely estimating the Kronecker initiator takes $O(N!N^2)$ time:
  - $N!$ for graph isomorphism
    - Metropolis sampling: $N! \rightarrow$ *(big) const*
  - N^2 for traversing the graph adjacency matrix
    - Properties of Kronecker product and sparsity ($E << N^2$): $N^2 \rightarrow E$
- We can estimate the parameters of Kronecker graph in linear time $O(E)$

Solution 1: Node correspondence

- Log-likelihood

$$l(\Theta) = \log \sum_{\sigma} P(G|\Theta, \sigma) P(\sigma)$$

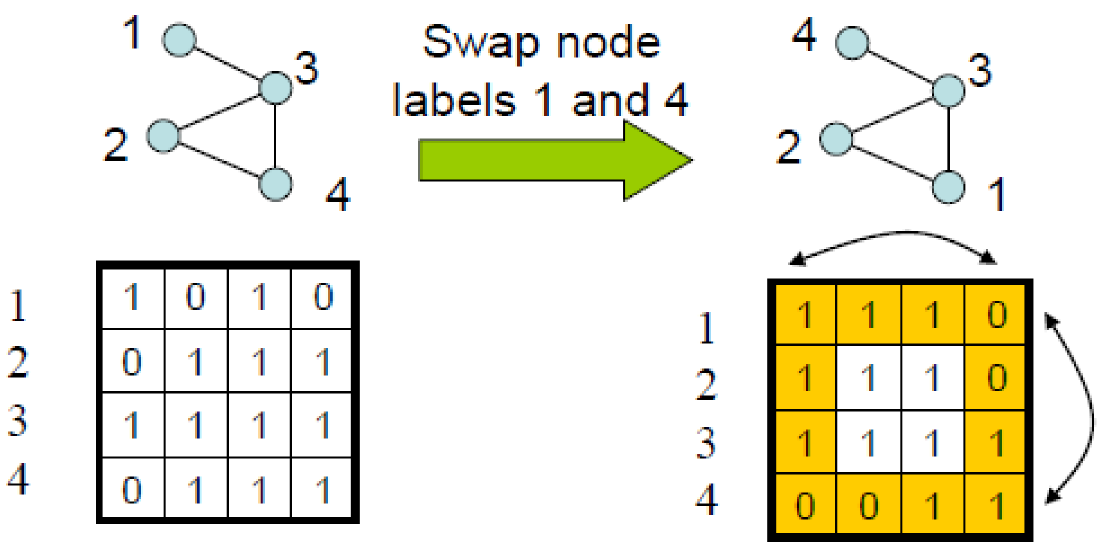
- Gradient of log-likelihood

$$\frac{\partial}{\partial \Theta} l(\Theta) = \sum_{\sigma} \frac{\partial \log P(G|\sigma, \Theta)}{\partial \Theta} \boxed{P(\sigma|G, \Theta)}$$

- Sample the permutations from $P(\sigma|G,\Theta)$ and average the gradients

Sampling node correspondences

- Metropolis sampling:
  - Start with a random permutation
  - Do local moves on the permutation
  - Accept the new permutation
    - If new permutation is better (gives higher likelihood)
    - If new is worse accept with probability proportional to the ratio of likelihoods



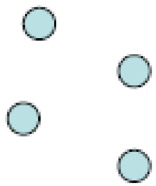
Can compute efficiently: Only need to account for changes in 2 rows / columns

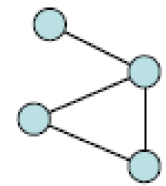
Solution 2: Calculating $P(G|\Theta,\sigma)$

- Calculating naively $P(G|\Theta,\sigma)$ takes $O(N^2)$
- Idea:
  - First calculate likelihood of empty graph, a graph with 0 edges
  - Correct the likelihood for edges that we observe in the graph
- By exploiting the structure of Kronecker product we obtain closed form for likelihood of an empty graph

Solution 2: Calculating P(G|Θ,σ)

- We approximate the likelihood:





$$l(\Theta) \approx \underbrace{l_e(\Theta) +}_{\text{Empty graph}} \sum_{(u,v)\in G} \underbrace{-\log(1-\Theta_k[\sigma_u,\sigma_v])}_{\text{No-edge likelihood}} + \underbrace{\log(\Theta_k[\sigma_u,\sigma_v])}_{\text{Edge likelihood}}$$

- The sum goes only over the edges
- Evaluating $P(G|\Theta,\sigma)$ takes $O(E)$ time
- Real graphs are sparse, $E << N^2$

Q & A