Complex networks
Stochastic Block Model

János Török

Department of Theoretical Physics

March 28, 2023

Preferential attachment

- Start with a seed of small network (e.g. clique)
- Attach new nodes to the existing network.
- If attached randomly, random network with exponential degree distribution
- Popular ones have higher chance to get new connections
- New ones attach with probability proportional to existing degree
- This is preferential attachment
- In networks it is called the Barabási-Albert model

Barabási-Albert model

- Probability that a node connects to a node is proportional to the degree of the target node:

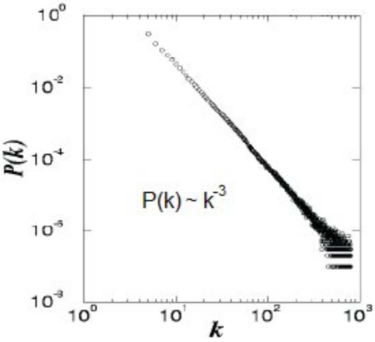
$$\Pi(i) = \frac{k_i}{\sum_j k_j}$$

- Parameter m number of links the new node makes
- Published in 1999
- Extensive impact on science

Barabási-Albert model

- Empirical degree distribution: power law
- Exponent independent of m
- $\gamma = 3$



Barabási-Albert model: Degree distribution calculation

- Number of nodes in time t, $N(t) = t$
- Number of links at time t, $L(t) = mt$
- Average degree at time t, $\langle k \rangle (t) = 2m/N$
- Number of nodes with degree k at time t

$$N(k,t) = Np(k,t) = tp(k,t)$$

- Preferential attachment:

$$\Pi(k) = \frac{k}{\sum_j k_j} = \frac{k}{2mt}$$

- Number of links added to nodes of degree k after the arrival of a new node

$$\underbrace{\frac{k}{2mt}}_{\text{Preferential attachment}} \times \underbrace{tp(k,t)}_{\text{Total number of } k \text{ nodes}} \times \underbrace{m}_{\text{New links}} = \frac{k}{2}p(k,t)$$

Barabási-Albert model: Degree distribution calculation

- Number of links added to nodes of degree k after the arrival of a new node

$$\underbrace{\frac{k}{2mt}}_{\text{Preferential attachment}} \times \underbrace{tp(k,t)}_{\text{Total number of } k \text{ nodes}} \times \underbrace{m}_{\text{New links}} = \frac{k}{2}p(k,t)$$

- Discrete time Master equation

$$(t+1)p(k,t+1) - tp(k,t) = \frac{k-1}{2}p(k-1,t) - \frac{k}{2}p(k,t)$$

Barabási-Albert model: Degree distribution calculation

- Discrete time Master equation

$$(t+1)p(k,t+1) - tp(k,t) = \frac{k-1}{2}p(k-1,t) - \frac{k}{2}p(k,t)$$

- For $k = m$ it is different, the gain term is the newly arriving node:

$$(t+1)p(m,t+1) - tp(m,t) = 1 - \frac{m}{2}p(m,t)$$

Barabási-Albert model: Degree distribution calculation

- We are interested in the steady state

$$\lim_{t \to \infty} p(k,t) = p(k)$$

- Steady state solution of the Master equation:

$$p(k) = \frac{k-1}{2} p(k-1) - \frac{k}{2} p(k)$$
$$p(m) = 1 - \frac{m}{2} p(m)$$

- Recursive relations

$$p(k) = \frac{k-1}{k+2} p(k-1) \quad \text{for } k > m$$
$$p(m) = \frac{2}{m+2} \quad \text{otherwise}$$

Barabási-Albert model: Degree distribution calculation

- Solution

$$p(k) = \frac{2m(m+1)}{k(k+1)(k+2)}$$

- Asymptotically

$$p(k) \sim k^{-3}$$

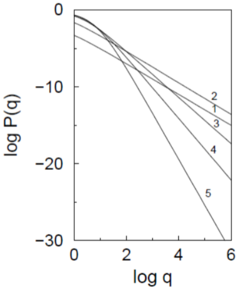
- Independent of m

Initial Attractiveness Model

- Even nodes without connections can be popular
- Often cited example: Citation networks (paper with no citation can be cited)

$$\Pi(k_i) = \frac{A + k_i}{A + \sum_j k_j}$$

- Asymptotically $p(k) \sim k^{-\gamma}$
- $\gamma = 2 + A/m$ tunable exponent

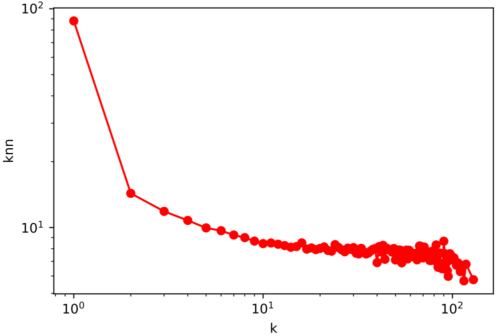


Dorogovtsev, Mendes, Samukhin, Phys. Rev. Lett. 85, 4633 (2000)

Assortativity in Barabási-Albert model

No calculations here :-)

- Disassortative regime $\gamma < 3$, $-m < A < 0$:
- Neutral regime $\gamma = 3$, $A = 0$
- Weak assortative regime $\gamma > 3$, $A > 0$



Clustering in Barabási-Albert model

Calculations :-!

- Definition

$$C = \frac{2N(\Delta)}{k(k-1)}$$

- Probability that nodes i and j are connected: $P(i,j)$
- Probability that nodes i, j, l form a triangle

$$N_l(\Delta) = \sum_{i,j} P(i,j)P(j,l)P(l,i)$$

- We need to calculate $P(i,j)$
- For this we will need the time evolution of the degree of the nodes

Time evolution in Barabási-Albert model

- The time evolution of the degree of the nodes

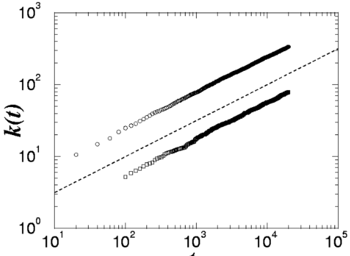
$$\frac{\partial k_i}{\partial t} \propto \Pi(k_i) = m\frac{k_i}{\sum_j k_j}$$

- Time is measured in units of nodes added, so at time t there are $N = t$ number of nodes and $L = mt$ number of links
- So

$$\frac{\partial k_i}{\partial t} \propto \frac{k_i}{2t}$$

- Solution

$$k_i(t) = m\sqrt{t/t_i} \sim t^{1/2}$$

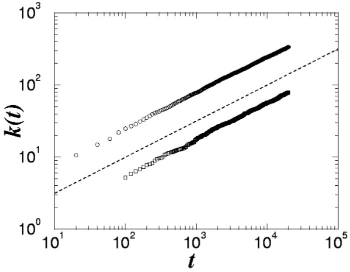


Time evolution in Barabási-Albert model

- The time evolution of the degree of the nodes

$$k_i(t) = m\sqrt{t/t_i} \sim t^{1/2}$$

- Advantage of the first comers!
- Very often one can take $t_i \equiv i$



Clustering in Barabási-Albert model

- Assume that $t_i < t_j$ (i came first)

$$P(i,j) = m\Pi(k_i(t_j)) = \frac{mk_i(t_j)}{\sum_l k_l} = m\frac{k_i(t_j)}{2mt_j}$$

- We know the time evolution of $k_i(t_j)$

$$k_i(t_j) = m\sqrt{t_j/t_i}$$

- From where we get

$$P(i,j) = \frac{m}{2}(t_i t_j)^{-1/2}$$

- Huhh... It is symmetric in i and j!

Clustering in Barabási-Albert model

- Back to the number of triangles:

$$N_l(\Delta) = \sum_{i,j} P(i,j)P(j,l)P(l,i) =$$

$$= \frac{m^3}{8} \sum_{t_i,t_j} (t_i t_j)^{-1/2} (t_j t_l)^{-1/2} (t_l t_i)^{-1/2}$$

$$= \frac{m^3}{8l} \sum_{t_i=1}^{N} \frac{1}{t_i} \sum_{t_j=1}^{N} \frac{1}{t_j}$$

- For large times $N \to \infty$

$$N_l(\Delta) = \frac{m^3}{8l} \log^2 N$$

Clustering in Barabási-Albert model

- So we have the number of triangles:

$$N_l(\Delta) = \frac{m^3}{8l} \log^2 N$$

- We also know that

$$k_l(t) = m\sqrt{N/t_l} \quad \text{so} \quad k_l(k_l - 1) \simeq m^2 N/t_l$$

- Finally the clustering coefficient is

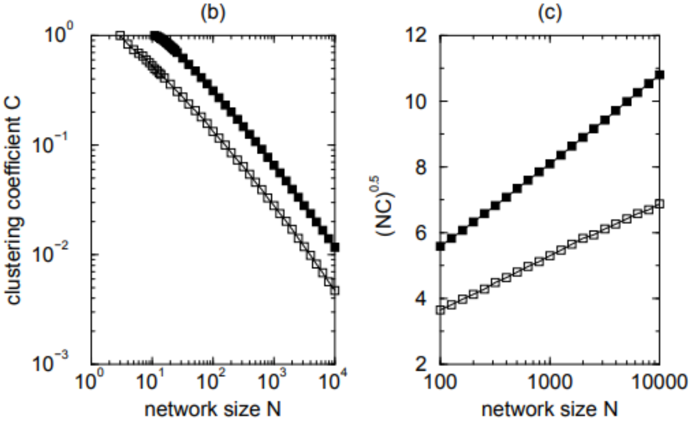
$$C = \frac{2\frac{m^3}{8l}\log^2 N}{k_l(k_l-1)} = \frac{m}{8}\frac{\log^2 N}{N}$$

- For large networks $N \to \infty$ the clustering vanishes $C \to 0$

Clustering in Barabási-Albert model

- The clustering coefficient for BA networks

$$C = \frac{m}{8}\frac{\log^2 N}{N}$$



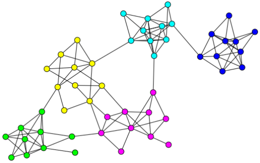
Other models

Linear growth, linear pref. attachment	$\gamma=3$	Barabási and Albert, 1999
Nonlinear preferential attachment $\Pi(k_i)\sim k_i^{\alpha}$	no scaling for $\alpha\neq1$	Krapivsky, Redner, and Leyvraz, 2000
Asymptotically linear pref. attachment $\Pi(k_i)\sim a_{\infty}k_i$ as $k_i\to\infty$	$\gamma\to2$ if $a_{\infty}\to\infty$ $\gamma\to\infty$ if $a_{\infty}\to0$	Krapivsky, Redner, and Leyvraz, 2000
Initial attractiveness $\Pi(k_i)\sim A+k_i$	$\gamma=2$ if $A=0$ $\gamma\to\infty$ if $A\to\infty$	Dorogovtsev, Mendes, and Samukhin, 2000a, 2000b
Accelerating growth $\langle k\rangle\sim t^{\theta}$ constant initial attractiveness	$\gamma=1.5$ if $\theta\to1$ $\gamma\to2$ if $\theta\to0$	Dorogovtsev and Mendes, 2001a
Internal edges with probab. p Rewiring of edges with probab. q	$\gamma=2$ if $q=\frac{1-p+m}{1+2m}$ $\gamma\to\infty$ if $p,q,m\to0$	Albert and Barabási, 2000
c internal edges or removal of c edges	$\gamma\to2$ if $c\to\infty$ $\gamma\to\infty$ if $c\to-1$	Dorogovtsev and Mendes, 2000c
Gradual aging $\Pi(k_i)\sim k_i(t-t_i)^{-\nu}$	$\gamma\to2$ if $\nu\to-\infty$ $\gamma\to\infty$ if $\nu\to1$	Dorogovtsev and Mendes, 2000b
Multiplicative node fitness $\Pi_i\sim\eta_i k_i$	$P(k)\sim\frac{k^{-1-C}}{\ln(k)}$	Bianconi and Barabási, 2001a
Edge inheritance	$P(k_{in})=\frac{d}{k_{in}^{\sqrt{2}}}\ln(ak_{in})$	Dorogovtsev, Mendes, and Samukhin, 2000c
Copying with probab. p	$\gamma=(2-p)/(1-p)$	Kumar *et al.*, 2000a, 2000b
Redirection with probab. r	$\gamma=1+1/r$	Krapivsky and Redner, 2001
Walking with probab. p	$\gamma\simeq2$ for $p>p_c$	Vázquez, 2000
Attaching to edges	$\gamma=3$	Dorogovtsev, Mendes, and Samukhin, 2001a
p directed internal edges $\Pi(k_i,k_i)\propto(k_i^{in}+\lambda)(k_i^{out}+\mu)$	$\gamma_{in}=2+p\lambda$ $\gamma_{out}=1+(1-p)^{-1}+\mu p/(1-p)$	Krapivsky, Rodgers, and Redner, 2001

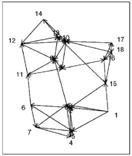
Gasometer Oberhausen

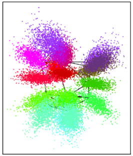
Random graphs/networks

- Generative models
  - randomly generating observable quantities
  - known examples:
    - Erdős-Rényi, or random graph model $\rightarrow$ no structure
    - Watts–Strogatz model $\rightarrow$ small world property
    - Configuration model $\rightarrow$ degree distribution
  - Stochastic Block Models (SBM)
    - will be detailed today
    - community structure
    - hierarchical structure



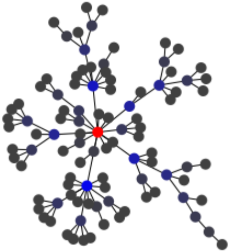
  - Latent Space Models
    - nodes live in a latent space
    - link properties depend on vertex-vertex proximity

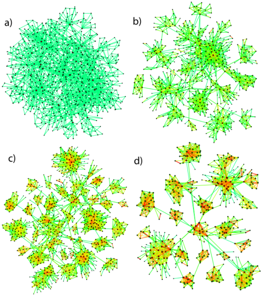




Random graphs/networks

- Growing networks
  - networks change as function of time
  - real life processes can be incorporated (realistic models)
  - stationary state representative of network
  - difficult to tune properties
  - examples:
    - Barabási-Albert model (preferential attachment)
    - Kumpula model (will be detailed today)

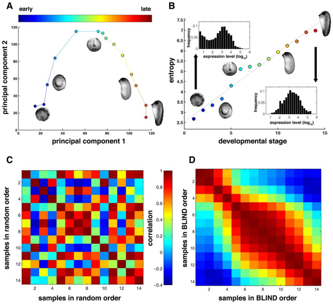


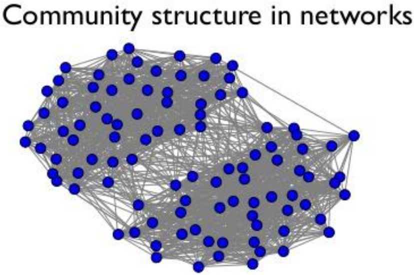


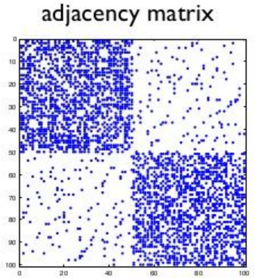
Kumpula, Jussi M., et al. PRL 99 (2007): 228701.

Block models

- Why?
- Adjacency matrix
- Communities
- Block structure

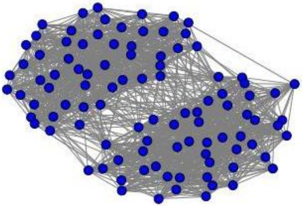


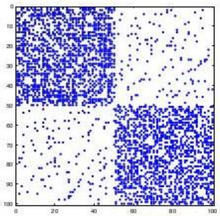




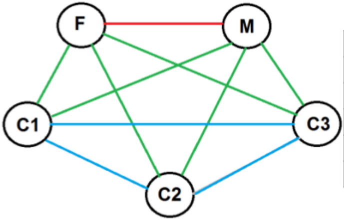
Stochastic Block Models (SBM)

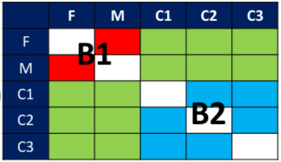
- Community structure





- Multi layer network (nodes are labeled)





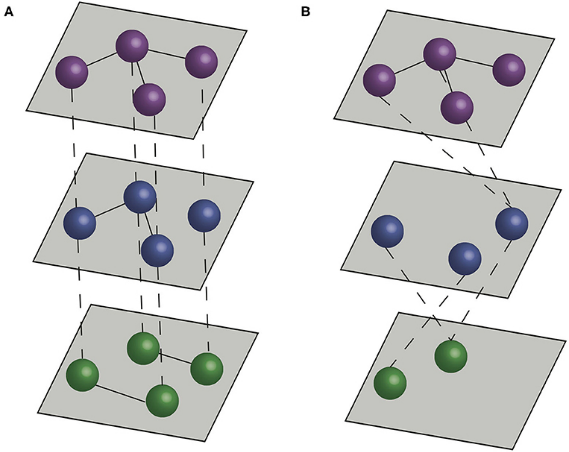
$\Rightarrow$

Block matrix

	B1	B2
B1	P_{11}	P_{12}
B2	P_{21}	P_{22}

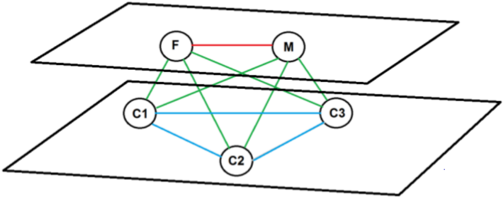
Multilayer, multiplex networks

Multiplex | **Multi layer**



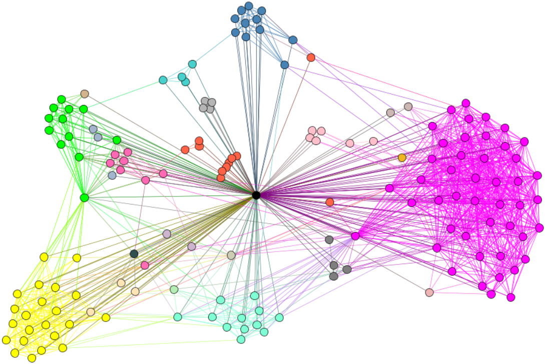
links are colored | nodes are colored

Multi layer representation



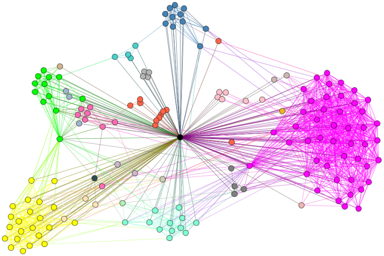
- Different layers (parents, children)
- Intra-layer links (parents – children)
- P_{ij} depends on the layers
- Here P_{ij}=1 special case

Link probability P_{ij}



- Intra-group links with high probability but not 1 (not everybody *knows* each other
- Inter-group links with much lower probability

Link probability P_{ij}



- Groupwise (blockwise) probability (i, j refers to groups)
- P_{ii} intra-group probability **high**
- P_{ij} inter-group probability **low**
- Stochastic equivalence: Probabilities for all links within a block are the same.

Generative models

- Given N nodes
- Define probability distributions for $P(G|\theta)$, where
  - G is a network instance
  - θ set of parameters describing the edge configurations
- Generate:
  - Given θ a network instance G can be generated
- Inference:
  - Given a network G we identify θ that produces it

$$\underbrace{P(G|\theta)}_{\text{model}} <=>[\text{Generation}][\text{Inference}] \underbrace{G=(V,E)}_{\text{data}}$$

Notation

- Number of nodes: N
- Indexes for nodes: u, v
- Adjacency matrix: A_{uv}
- Number of blocks: K
- Indexes for blocks: i, j
- Link probability between groups: P_{ij}

Stochastic Block Models (SBM)

Definition of θ:

- K: number of groups in the model
- z: a N dimensional vector indexing to which group a node belongs to. E.g. $z(i) \in [1, K]$ gives the group index of node i.
- P_{ij}: a $K \times K$ matrix describing the probability that a vertex of group i is connected to a vertex of group j.

Note[1]:

P_{ii}: gives the probability that vertexes of group i are connected.

Note[2]:

Graphs of all groups are Erdős-Rényi random graphs

Note[3]:

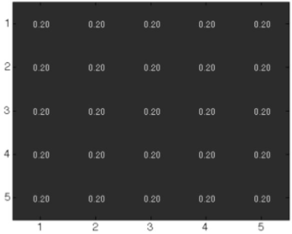
Alternative definition: $\theta = \{K, \mathrm{z}, P_{ij}\} \equiv \{K, \mathrm{s}, P_{ij}\}$, where s is a K dimensional vector with the size of the group as value. Of course

$$\sum_{i=1}^{K} s(i) = N$$

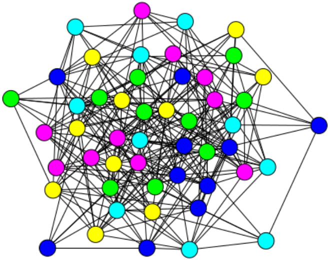
Example: N=50, K=5, s = {10, 10, 10, 10, 10}

Generation

Erdős-Rényi graph



stochastic block matrix

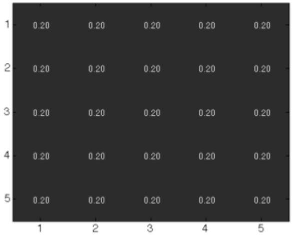


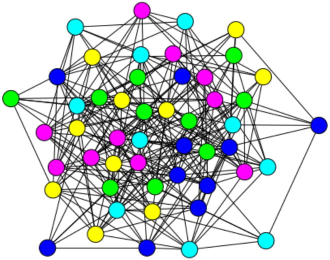
random graph

All examples: Aaron Clauset: *Network Analysis and Modeling, CSCI 5352, Lecture 16*

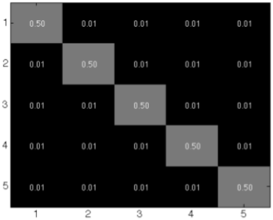
Example: N=50, K=5, s = {10, 10, 10, 10, 10}

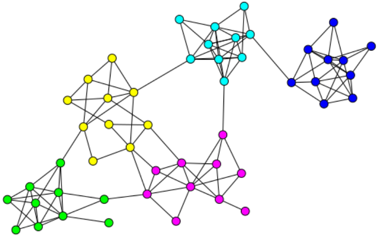
Erdős-Rényi graph





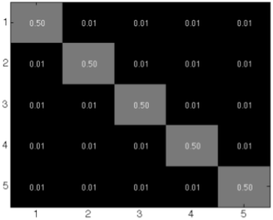
Communities

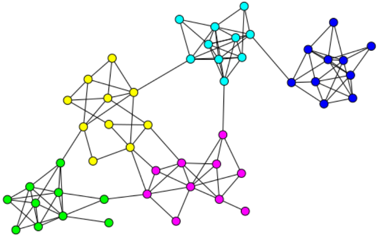




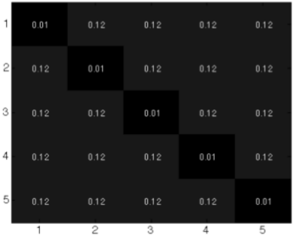
Example: $N=50$, $K=5$, $s = \{10, 10, 10, 10, 10\}$

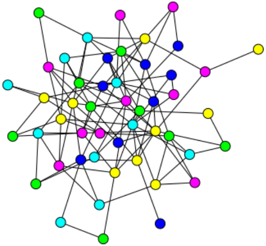
Assortative communities





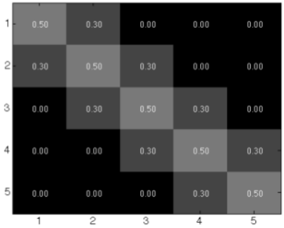
Disassortative communities

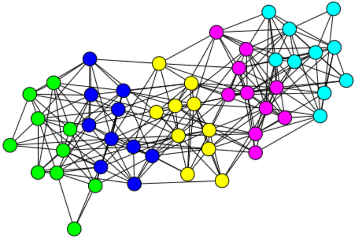




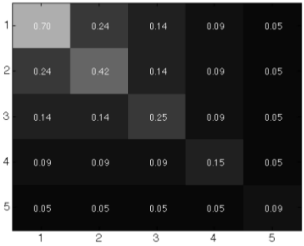
Example: N=50, K=5, $s = \{10, 10, 10, 10, 10\}$

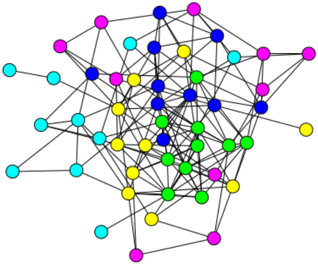
Ordered communities





Core-periphery structure





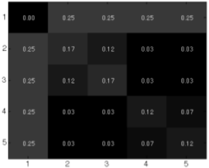
SBM: Degree distribution

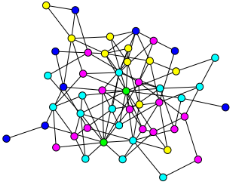
- All groups are ER subgraph with Poisson degree distribution
- Resulting degree distribution is a mixture of Poissonians

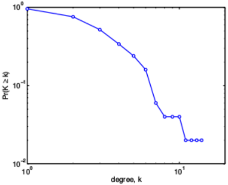
$$E[n|z(n)=j] = \sum_{i=1}^{K} s(i)P_{ij}$$

The expected degree of a node n in group j.

Example for wide distribution



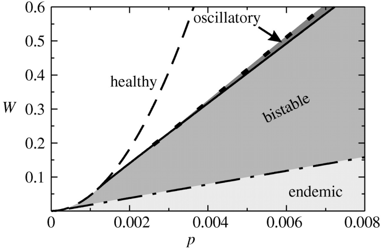




Generation

- Analyze parameter space
- Test for desired quantities, e.g. degree distribution, modularity, assortativity.
- Run parameter scan, and measure quantities
- Draw a phase diagram
- For practical use choose desired parameters
- Nowdays: Estimate it with neural network

e.g. Adaptive coevolutionary networks:



T Gross, B Blasius - Journal of the Royal Society Interface, 2008

SBM: Inference

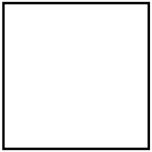
- How to guess θ if we want to model a system with given characteristics?
- To be determined: K, s, P_{ij}. Total:

$$\underbrace{1}_{K} + \underbrace{(K-1)}_{\text{s}} + \underbrace{K(K-1)/2}_{P_{ij},\ i \neq j} + \underbrace{K}_{P_{ii}} = K(K+3)/2$$

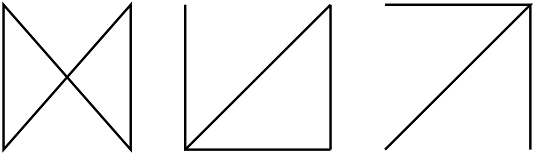
- Brute force will not work
- Maximum likelihood estimation
  - estimate the parameters of a stochastic model such that they maximize the likelihood of obtaining the predefined observations.
  - given the value K the task is to estimate the values of z, and P_{ij}

Maximum likelihood: Example

- We have four nodes and the network is a square



- We want to use the Erdős-Rényi model
- What is p for which we get the square with the maximum likelihood?
- Obviously it is $p=2/3$. But we can get anything like:



SBM: Maximum likelihood

- Likelihood function: Calculate the probability of having an edge between nodes u, v if there was an edge, or the probability of not having an edge if there was none:

$$\mathcal{L}(G|M, z) = \prod_{(u,v)\in E} P[(u,v)|\theta] \prod_{(u,v)\notin E} \{1 - P[(u,v)|\theta]\},$$

  where $P[(u,v)|\theta]$ is the probability of generating an edge between nodes u, v.

- The number of possible links between groups:

$$N_{ij} = \begin{cases} s_i s_j & \text{if } i \neq j \\ s_i(s_i - 1)/2 & \text{if } i = j \end{cases}$$

- Expected number of links between groups is denoted by E_{ij}

SBM: Maximum likelihood

It is obvious that the maximum is when $P_{ij} = E_{ij}/N_{ij}$:

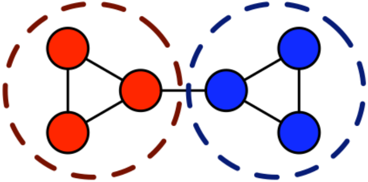
$$\mathcal{L}(G|M,z) = \prod_{(u,v)\in E} P[(u,v)|\theta] \prod_{(u,v)\notin E} \{1 - P[(u,v)|\theta]\}$$
$$= \prod_{i,j} \left(\frac{E_{ij}}{N_{ij}}\right)^{E_{ij}} \left(1 - \frac{E_{ij}}{N_{ij}}\right)^{N_{ij}-E_{ij}}$$

It is customary to calculate the log:

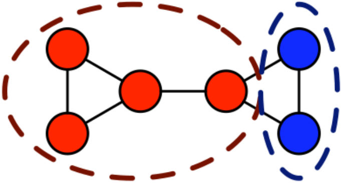
$$\log \mathcal{L}(G|M,z) = \sum_{ij} [E_{ij} \log E_{ij} + (N_{ij} - E_{ij}) \log(N_{ij} - E_{ij}) - N_{ij} \log N_{ij}]$$

- What does $\mathcal{L}$ mean?

SBM: Likelihood example



$$\mathcal{L}_{\text{good}} = 0.043304\ldots$$
$$\ln \mathcal{L}_{\text{good}} = -3.1395\ldots$$



$$\mathcal{L}_{\text{bad}} = 0.000244\ldots$$
$$\ln \mathcal{L}_{\text{bad}} = -8.3178\ldots$$

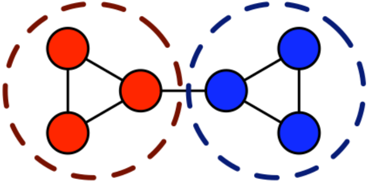
M_{good}	red	blue
red	3/3	1/9
blue	1/9	3/3

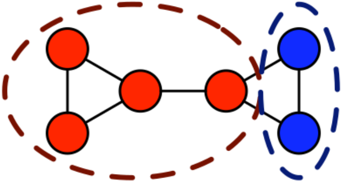
M_{bad}	red	blue
red	4/6	2/8
blue	2/8	1/1

$$\mathcal{L}(G|M,z) = \prod_{i,j} \left(\frac{E_{ij}}{N_{ij}}\right)^{E_{ij}} \left(1 - \frac{E_{ij}}{N_{ij}}\right)^{N_{ij}-E_{ij}}$$

$$= \left(\frac{3}{3}\right)^{3} \underbrace{\left(1 - \frac{3}{3}\right)^{0}}_{=1} \cdot \left(\frac{1}{9}\right)^{1} \left(\frac{8}{9}\right)^{8} \cdot 1^{3} \underbrace{0^{0}}_{=1} = 0.0433\ldots$$

SBM: Likelihood meaning





$\mathcal{L}_{\text{good}} = 0.043304\ldots$
$\ln \mathcal{L}_{\text{good}} = -3.1395\ldots$

$\mathcal{L}_{\text{bad}} = 0.000244\ldots$
$\ln \mathcal{L}_{\text{bad}} = -8.3178\ldots$

M_{good}	red	blue
red	3/3	1/9
blue	1/9	3/3

M_{bad}	red	blue
red	4/6	2/8
blue	2/8	1/1

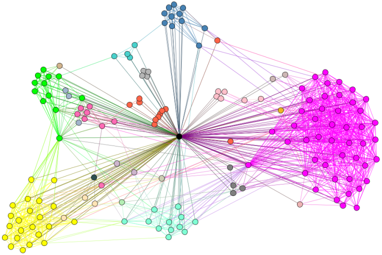
- $\mathcal{L}_{\text{good}} \simeq 177 \cdot \mathcal{L}_{\text{bad}}$: The good partition is 177 times more likely to generate the original data than the bad one.

SBM: Optimizing the likelihood

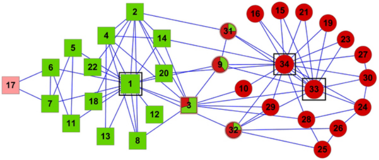
- For given K we may optimize the partition, see below.
- Optimizing K: problem $\rightarrow$ with increasing K the number of fit parameters increase as well $\rightarrow$ better fit
- Limiting case $K = N$, $P_{ij} = A_{ij}$, $\rightarrow$ perfect fit, and $\mathcal{L} = 1$
- Some knowledge is required from the system to estimate K

SBM: Problems

- SBM: Nodes in one block have similar degrees
- Good example: egocentric network

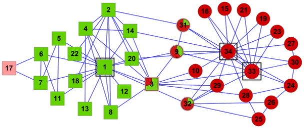


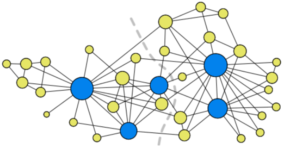
- Bad example: Zachary karate club



SBM: Zachary karate club

- Social partition, vs. SBM partition





- Likelihood values

M_{social}	A (17)	B (17)
A (17)	35/136	11/289
B (17)	11/289	32/136
A (17)	0.2574	0.0381
B (17)	0.0381	0.2353

social division, $\ln \mathcal{L} = -198.50$

M_{SBM}	A (5)	B (29)
A (5)	5/10	54/145
B (29)	54/145	19/406
A (5)	0.5000	0.3724
B (29)	0.3724	0.0468

SBM division, $\ln \mathcal{L} = -179.39$

- SBM is 10^8 times more likely!

Degree corrected SBM: null model

- Logarithm of likelihood, leaving out constant factors:

$$\log \tilde{\mathcal{L}} = \sum_{ij} E_{ij} \log \frac{E ij}{\kappa_i \kappa_j}$$

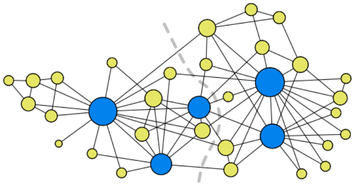
where κ_i is the number of stubs in group i

- Similar to the definition of modularity.
- Null model is not Erdős-Rényi but a network with the expected degree sequence.

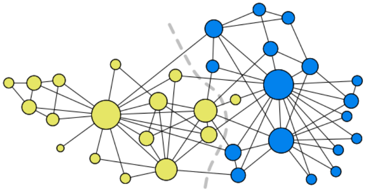
Degree corrected SBM: Results

Zachary karate club

SBM



degree corrected SBM



Degree corrected SBM: Algorithm

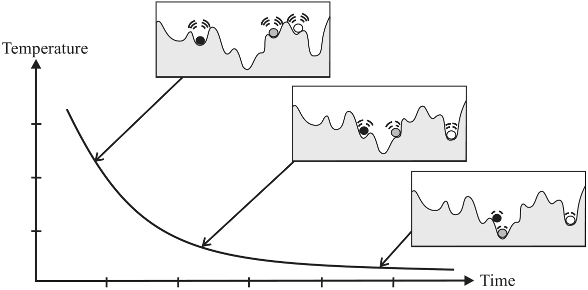
- In principle: Given K, calculate $\tilde{\mathcal{L}}$ for all possible divisions and select the one with the largest value.
- This is impossible $\sim \binom{N}{K}$
- Optimization in a multi dimensional space
- Separate field of research

Optimization

Methods:

- Gradient (greedy):
  - Always decrease the path length
  - Fast, but gets trapped in a local minimum
- Simulated annealing:
  - define elementary step
  - decrease temperature slowly
  - if energy is decreased by move $\rightarrow$ do it
  - allow for increase of energy with probability proportional to

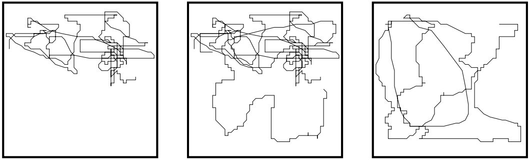
$$P \sim \exp(-\Delta E / T)$$



Simulated annealing for SBM

Elementary step

- Ergodic: able to reach all states, time and ensemble averages are the same



Non ergodic — Ergodic — Ergodic

- e.g. transfer a node from block i to j
- long self averaging times (middle example)
- *clever* choice of elementary step

Other name: Markov Chain Monte Carlo

Simulated annealing for SBM

Elementary step

- Transfer a node u from i to j, (k is a randomly chosen block)

$$p(i \to j|k) = \frac{N_{ik} + \varepsilon}{N_k + \varepsilon K}$$

  where N_{ik} is the number of links between groups i and k and N_k the links in block k. $\varepsilon > 0$ a free parameter. This tests how much u is attached to k

- The transition probability is thus:

$$w(u, i \to j) = \min\left\{ e^{-\beta \Delta \log \tilde{\mathcal{L}}} \frac{\sum_k p_k^u p(i \to j|k)}{\sum_k p_k^u p(j \to i|k)}, 1 \right\}$$

  where $\beta = 1/T$ inverse temperature, p_k^u is the fraction of neighbors of node u belonging to block k.

Simulated annealing for SBM

Elementary step

- Transfer a node u from i to j, (k is a randomly chosen block)

$$p(i \to j|k) = \frac{N_{ik} + \varepsilon}{N_k + \varepsilon K}$$

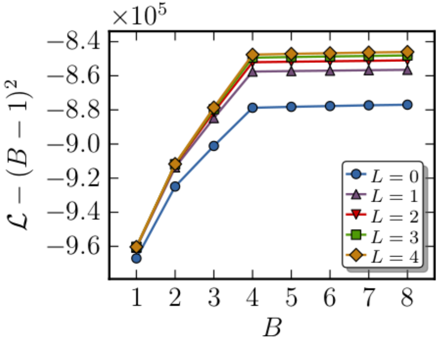
- The transition probability is thus:

$$w(u, i \to j) = \min \left\{ e^{-\beta \Delta \log \tilde{\mathcal{L}}} \frac{\sum_k p_k^u p(i \to j|k)}{\sum_k p_k^u p(j \to i|k)}, 1 \right\}$$

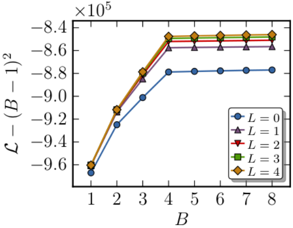
- $\beta = \infty$: greedy algorithm.
- Slowly increase β: simulated annealing
- An efficient C++ implementation of the algorithm described here is freely available as part of the graph-tool Python library at `http://graph-tool.skewed.de` (Peixoto, 2014)

SBM: Optimal selection for K

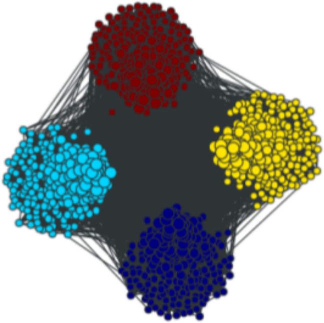
- $\tilde{\mathcal{L}}$ grows with K
- asymptotic increase $\log \tilde{\mathcal{L}} \sim (K-1)^2$
- Use $\log \mathcal{L}^* = \log \tilde{\mathcal{L}} - (K-1)^2$ which is expected to become a constant for large K
- e.g.: simulated data $s = (250, 250, 250, 250)$, $p_k \propto k^{-1.1}$, for $k \in [k_{\min}, k_{\max}]$
- Graph $B \equiv K$



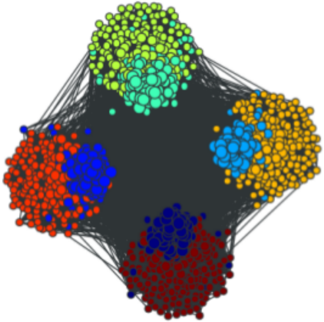
SBM: Optimal selection for K



L controls the precision of the likelihood function



$B = 4, L = \{0, 1, 2, 3, 4\}$



$B = 8, L = 0$

SBM: Summary

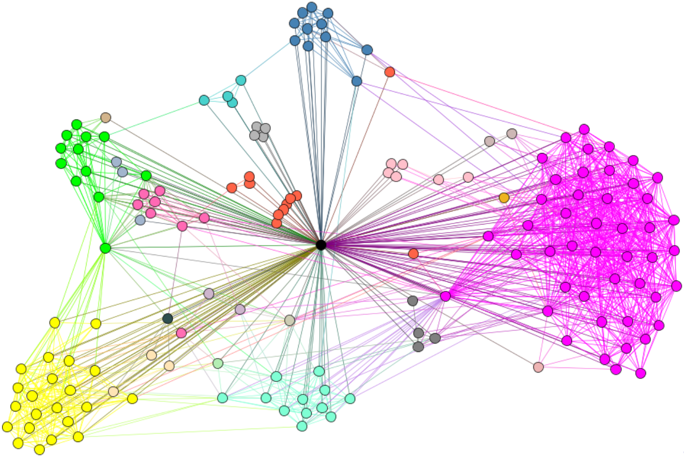
- Very flexible, generative method to model
- Communities, but also arbitrary mixing patterns, including, for example, bipartite, and core-periphery structures;
- Able to separate noise from structure;
- No resolution limit
- Generalization to directed, weighted networks possible.
- Structure detection is converted to parameter inference
- Increasingly efficient algorithms
- Can be used to detect communities

SBM: Suggested reading

- B. Karrer and M. E. J. Newman, Degree-corrected block modeling, Physical Review E **83**, 016107 (2011)
- T.P. Peixoto, Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models, Physical Review E **89** (1), 012804 (2014)
- T.P. Peixoto, Hierarchical Block Structures and High-Resolution Model Selection in Large Networks, Physical Review X **4**, 011047 (2014)

Growing networks

- Simulate real life
- Use minimal elements
- Do not incorporate effect what one wants to recover
- Example: simulate social network (modular)



Growth models

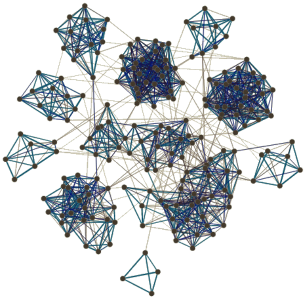
- Barabási-Albert model: Simple growth mechanism, preferential attachment, model for Internet
- More complicated systems?
- Two version of a simple model for social networks

Social networks

- Human relation
- Very complicated dynamics
- Not really a growth model, more a dynamics steady state
- Observations:
  - Weighted network
  - Large clustering coefficient (friend of friends usually know each other)
  - Not scale free
  - Small world
  - Granovetter: Strength of the weak ties

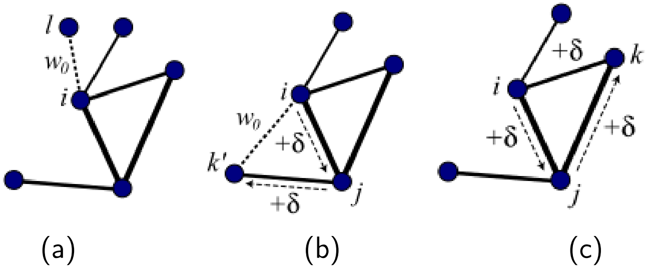
Granovetter: Strength of the weak ties

- Human groups are strongly connected
- There are weak connections connecting the groups
- These weak connections mean sproadic meeting
- Important for information flow
- Example: Find a job



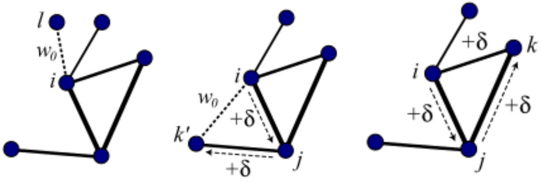
Kumpula model

- N nodes (originally unconnected)
- (a) Randomly meet someone (low probability) global attachment
- (b) Two friends of someone get to know each other, cyclic closure
- (c) An already present triangle gets strengthened



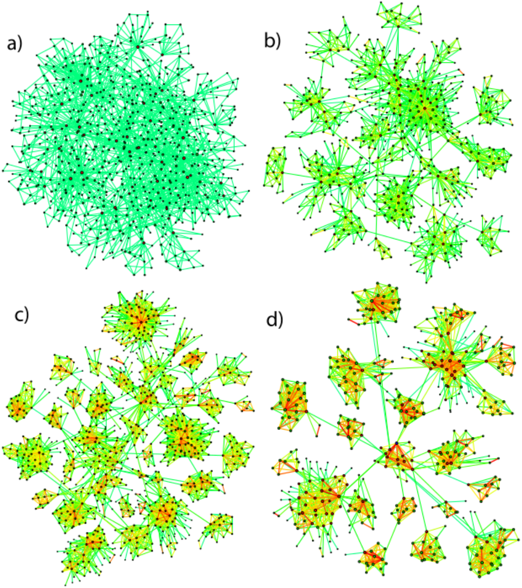
Kumpula model

- N nodes (originally unconnected)
- (a) (with prob. p_r) random link to an unconnected node. Link weight w_0
- (with prob. p_d) i selects friend j with prob. proportional to the link weight. j selects friend k similarly. Both links are strengthened by δ. Two cases:
  - (b) There is no link between i and k: create a link with p_Δ with weight w_0
  - (c) There is a link between i and k: strengthen by δ
- (d) (with prob. p_d) clear the links of a node (enforce steady state, there are more realistic versions)



(a) (b) (c)

Kumpula model: results ($\delta = 0,\ 0.1,\ 0.5,\ 1.0$)



Kumpula model: results

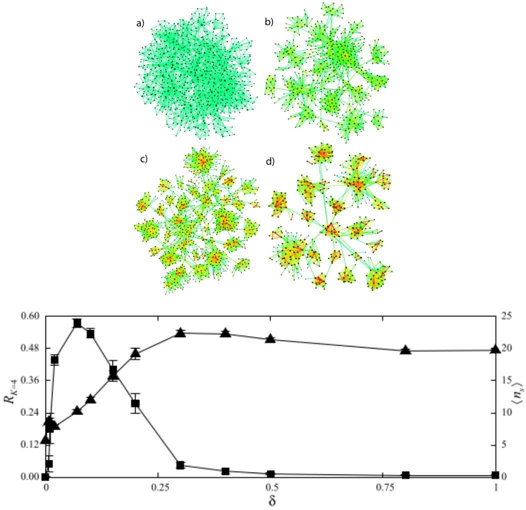


FIG. 3: $R_{k=4}$ (□) and $\langle n_s \rangle$ (△) as a function of δ. Results are averaged over 10 realizations of $N = 5 \times 10^4$ networks. Error bars are measured standard deviations.

Kumpula model: results

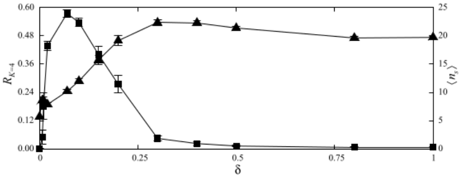
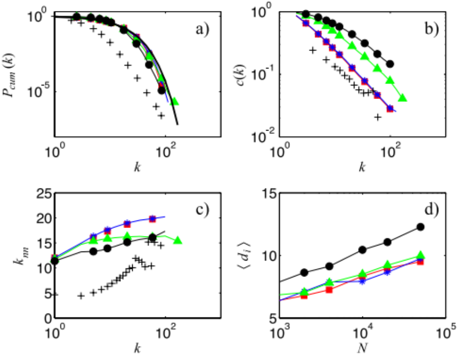


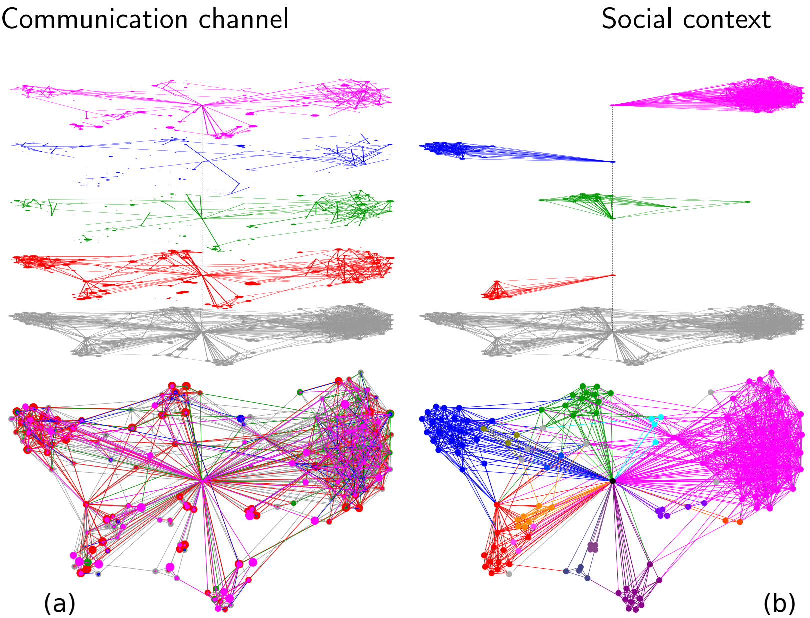
FIG. 3: $R_{k=4}$ (□) and $\langle n_s \rangle$ (△) as a function of δ. Results are averaged over 10 realizations of $N = 5 \times 10^4$ networks. Error bars are measured standard deviations.



Kumpula model: results

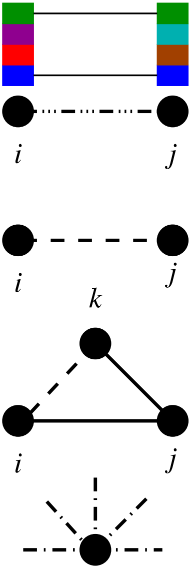
- Very simple assumptions
- Emergence of community structure (depending on parameters)
- Good to test effects of elementary processes on global structure
- Not apt for recovering well defined structures

Multiplex networks: Social networks

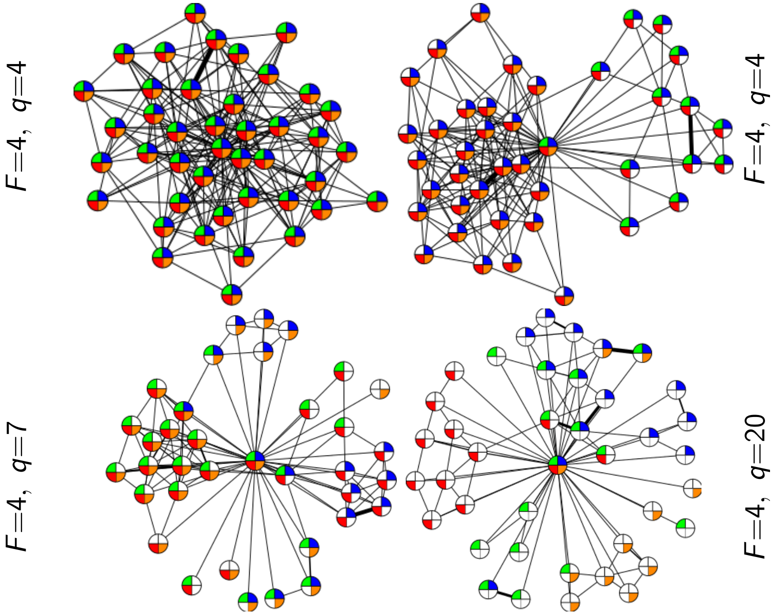


Multiplex model of social networks

- Peaple have F social features with q values each
- Ego first selects feature (s)he wants to do some social action
- (S)he can do it only with people with matching the specific feature
- Random connection, rare
- Triangles: common
  - Link selection proportional to weight
  - Link establishment with some probability and strengthening participating links
  - Link aging



Multiplex social model: egocentric networks



Multilayer social model: Phase diagram