Aalto University
School of Science
and Technology

CS-E5745
Mathematical Methods for Network Science

Mikko Kivelä

Department of Computer Science
Aalto University, School of Science
mikko.kivela@aalto.fi

December 22, 2021

Generating functions and their use in networks

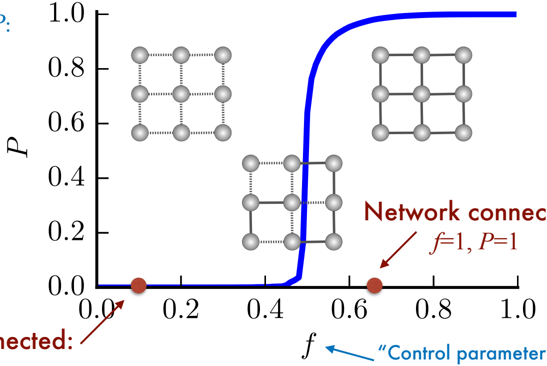
- Learning goals this week:
  - Learn the concept of probability generating functions (PGF's) and their basic properties
  - Recognise what kind of problems can be solved with PGF's and be able to solve them
  - Learn how to solve a Galton-Watson process using PGF's and how to apply that to networks
- We will be following the Section 13 in Newman: *Networks, An Introduction*

Components and excess degree

- Problem: Find the component size distribution of a (sparse) network produced by a configuration model
  - Assumptions: network is infinitely large, there are almost no loops
- Equivalent problem: start a BFS process from random node in a tree
  - Branching factor is given by the *excess degree* distribution $q(k)$
- Reminder: We already did this in the basic course (8 next slides)

Percolation theory

- Change something in the network (add/remove links, increase transmission probability, etc) and the component structure changes



“Order parameter” P: fraction of nodes in the largest connected component (LCC)

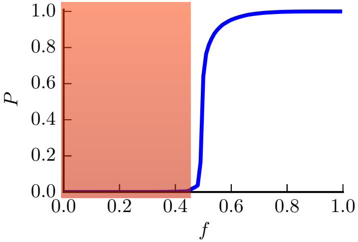
Network connected:
f=1, P=1

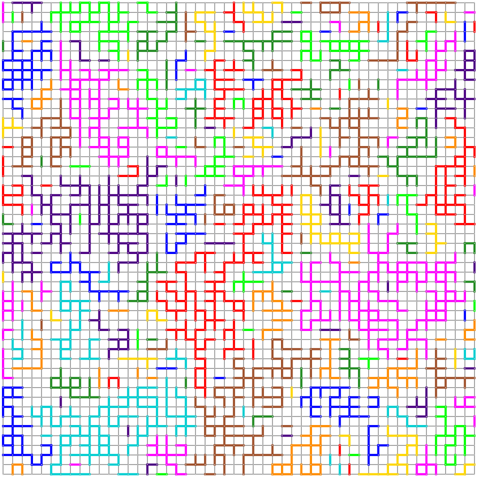
Network disconnected:
f=0, P≈0

“Control parameter” f: fraction of active links

Disconnected phase

- Largest component relatively small
- Other components small

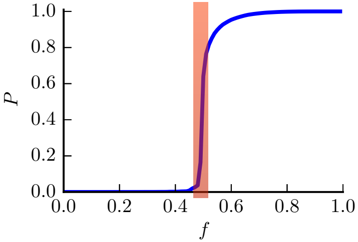


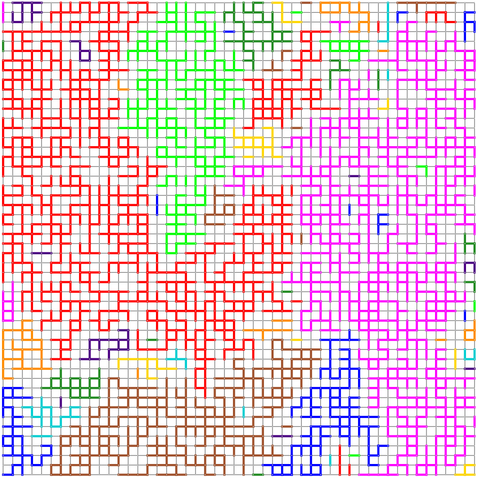


f=0.4

Phase transition

- The largest component becomes the “giant component”
- Other components from very large to small

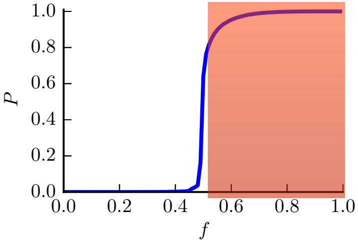


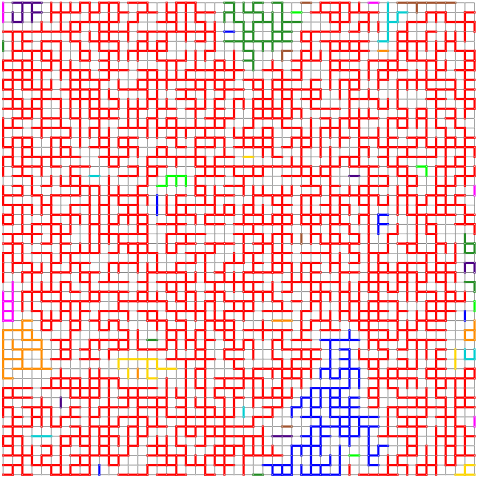


f=0.5

Connected phase

- The giant component size same scale as network size
- Other components small

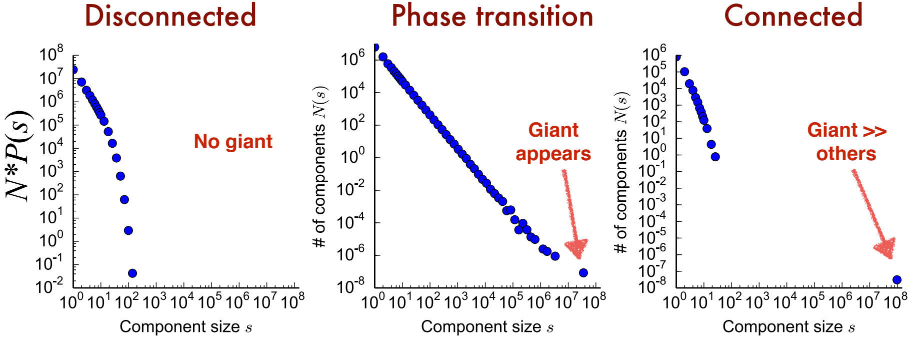




f=0.55

Component size distributions

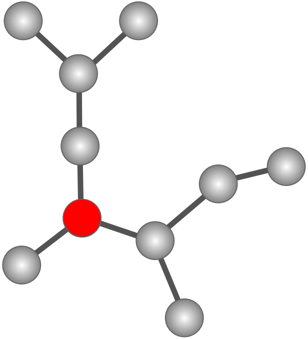
(square grid with $N=10^4*10^4$ nodes)



- The size distribution of other components at the phase transition point follows a power law!
  - "Critical point" in the theory of critical phenomena

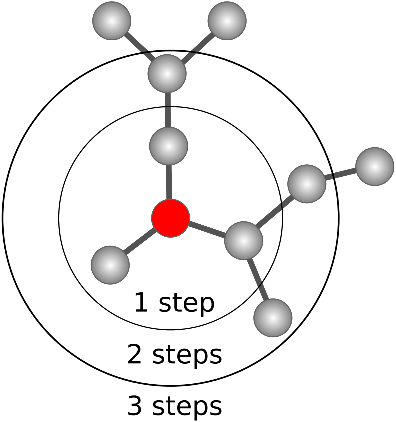
How to estimate the transition point?

- Idea: start from a random node, find how many nodes you can reach
- Before transition: you can always reach only a small number of nodes
- After transition: possibility of reaching very large number of nodes

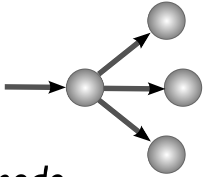


Branching processes

- Sparse large random networks have (almost) no loops
- Breadth first search is a "branching process":
  - A node has q "children"
- At step t, n_t nodes
  - $n_{t+1} = \langle q \rangle n_t$
  - Exponential growth ($\langle q \rangle > 1$) or decay ($\langle q \rangle < 1$)



Excess degree



- The excess degree q: *follow a link to a node, how many links does it have, not including the link that was followed?*
  - Remember the friendship paradox: following a link leads to high degree nodes: $\langle k_{nn} \rangle = \langle k^2 \rangle / \langle k \rangle$
- Expected excess degree: $\langle q \rangle = \langle k^2 \rangle / \langle k \rangle - 1$

expected number of neighbours

not including the link that was followed

Components and excess degree

- Problem: start a BFS process from random node in a tree
  - Branching factor is given by the *excess degree* distribution
- There are k_1 neighbors where k_1 is drawn from $p(k)$. If $k_1 > 0$:
  - There are $k_2 = \sum_{i=1}^{k_1} k_{1,i}$ second neighbors where each $k_{1,i}$ (number of second neighbors the first neighbor i has) is drawn from $q(k)$. If $k_2 > 0$:
  - There are $k_3 = \sum_{i=1}^{k_2} k_{2,i}$ third neighbors where each $k_{2,i}$ is drawn from $q(k)$. If $k_3 > 0$:
  - ...
- What is the distribution of k_2, k_3, ... ?
  - This is a variation of the *Galton-Watson process*
  - We can write the above equations using *random variables* K_d, and solve them using *probability generating functions*

Probability generating functions

- Let X be a random variable with non-negative integers as outcomes, and probability distribution $P(X = k) = p(k)$:

$$g(z) = p(0) + p(1)z + p(2)z^2 \cdots = \sum_{k=0}^{\infty} p(k)z^k \tag{1}$$

- Example: $p(1) = 0.5$ and $p(2) = 0.5$, then PGF is $g(z) = 0.5z + 0.5z^2$
- Example: Poisson distribution $p(k) = e^{-c}\frac{c^k}{k!}$ gives $g(z) = \sum_{k=0}^{\infty} e^{c}\frac{c^k}{k!}z^k = e^{c(z-1)}$

Probability generating function properties (1/4)

- $p(k)$ can be extracted through derivation

$$p(k) = \left[\frac{1}{k!} \frac{d^k}{dz^k} g(z) \right]_{z=0} \tag{2}$$

- Example: for $g(z) = 0.5z + 0.5z^2$, we get $p(2) = \left[\frac{1}{2!} \frac{d^2}{dz^2} g(z) \right]_{z=0} = \left[\frac{1}{2!} 1 \right]_{z=0} = 0.5$
- Example: for $g(z) = e^{c(z-1)}$, we get $p(2) = \left[\frac{1}{2!} \frac{d^2}{dz^2} g(z) \right]_{z=0} = \left[\frac{1}{2} c^2 e^{c(z-1)} \right]_{z=0} = \frac{1}{2} c^2 e^{-c}$

Probability generating function properties (2/4)

- Moments can also be calculated through derivation

$$\langle X^m \rangle = \left[\overbrace{z\frac{d}{dz} \ldots z\frac{d}{dz}}^{m} g(z) \right]_{z=1} = \left[(z\frac{d}{dz})^m g(z) \right]_{z=1} \quad (3)$$

- Works also for the “zeroth” moment: $g(1) = 1$

Probability generating function properties (3/4)

- Sums of independent random variables X_1 and X_2 become products of GFs

$$g_{X_1+X_2}(z) = g_{X_1}(z) * g_{X_2}(z) \tag{4}$$

- If the X_i *i.i.d.* then the sum $S = \sum_{i=1}^{N} X_i$ becomes a power of the GF

$$g_S(z) = [g_{X_i}(z)]^N \tag{5}$$

- Constant c is just a random variable that always has the same result

$$g_{X_1+c}(z) = g_{X_1}(z) * z^c \tag{6}$$

Probability generating function properties (4/4)

- If N is also a random variable in $S = \sum_{i=1}^{N} X_i$, then the sum becomes a combination

$$g_S(z) = g_N(g_{X_i}(z)) \tag{7}$$

- This is the case in the Galton-Watson process!

Generating functions for degrees

- We use the notation from Newman:
  - For the degree distribution $p(k)$:

$$g_0(z) = \sum_{k=0}^{\infty} p(k) z^k$$

  - For the excess degree distribution $q(k)$:

$$g_1(z) = \sum_{k=0}^{\infty} q(k) z^k$$

- These two are related: (Exercise 4a)

$$g_1(z) = \frac{1}{\langle k \rangle} \frac{d}{dz} g_0(z) \tag{8}$$

Solving the Galton-Watson process for networks

- The number of first neighbors of a random node k_1 is drawn from the degree distribution $p(k)$

$$g_{K_1}(z) = g_0(z)$$

- Each second neighbor i adds $k_{1,i}$ new nodes, and these numbers come from the excess degree distribution $q(k)$

$$g_{K_{1,i}}(z) = g_1(z)$$

Solving the Galton-Watson process for networks

- The number of second neighbors K_2 is the sum of excess degrees $K_{1,i}$

$$K_2 = \sum_{i=1}^{K_1} K_{1,i}$$

- Using the combination property (7)

$$g_{K_2}(z) = g_0(g_1(z))$$

Solving the Galton-Watson process for networks

- The number of third neighbors K_3 is the sum of excess degrees $K_{2,i}$

$$K_3 = \sum_{i=1}^{K_2} K_{2,i}$$

- Using the combination property and $g_{K_2}(z) = g_0(g_1(z))$

$$g_{K_3}(z) = g_{K_2}(g_1(z)) = g_0(g_1(g_1(z)))$$

Solving the Galton-Watson process for networks

- We get a recursive equations

$$\begin{aligned} g_{K_1}(z) &= g_0(z) \\ g_{K_d}(z) &= g_{K_{d-1}}(g_1(z)) \end{aligned}$$

- Writing closed form solutions for $p(k_d)$ often not possible
- The expected value can be solved in closed form for any d:

$$\langle K_d \rangle = \langle q \rangle^{d-1} \langle k \rangle = \left(\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} \right)^{d-1} \langle k \rangle \qquad (9)$$

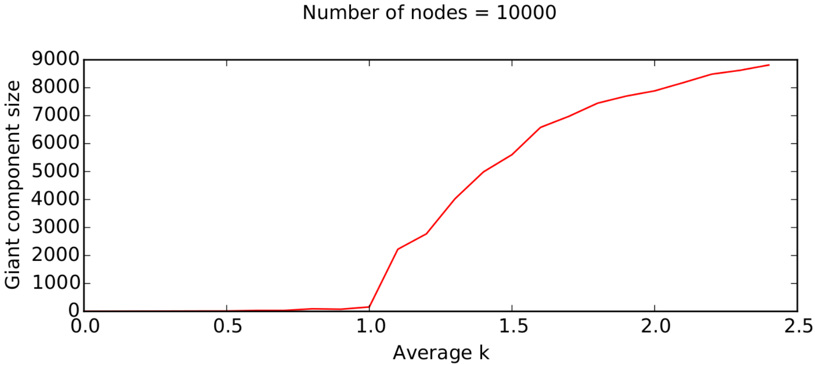
- Diverges if $\langle q \rangle > 1$

Solving the Galton-Watson process for networks

- If for some d we get $K_d = 0$ we say that there is an *extinction*
    - $\langle q \rangle > 1$: Probability of extinction smaller than 1 (supercritical)
    - $\langle q \rangle < 1$: Probability of extinction is 1 (subcritical)
- When $\langle q \rangle = 1$ the system is at *critical state*
    - The extinction d time, total number of reachable nodes $\sum_d K_d$ etc. are distributed as power-laws $p(d) \propto d^{\alpha}$
    - The exponents of these power-laws are the *critical exponents*

Example: Erdős-Rényi networks

- The "percolation threshold" for $G(N, p)$ was solved numerically in the Exercise 2.1d in CS-E5740:



Example: Erdős-Rényi networks

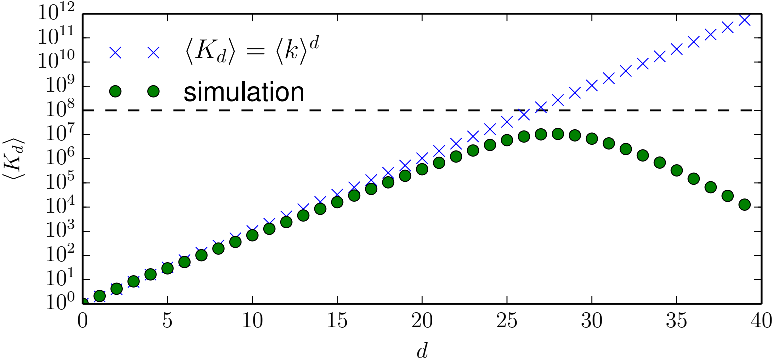
- $G(N, p)$ has Poisson degree distribution when $N \to \infty$ while $\langle k \rangle$ is constant
  - $p(k) = \frac{\langle k \rangle^k}{k!} e^{-\langle k \rangle}$.
  - Second moment $\langle k^2 \rangle = \langle k \rangle^2 + \langle k \rangle$
- Average excess degree

$$\langle q \rangle = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} = \langle k \rangle$$

- $\langle K_d \rangle = \langle k \rangle^d$
- The giant component exists iff $\langle k \rangle > 1$

Example: Erdős-Rényi networks

- Result can be compared to simulations (ER network with $N = 10^8$ and $\langle k \rangle = 2$)



Example: Erdős-Rényi networks

- We can also try to solve the distributions of each K_d for ER networks:
  - $g_0(z) = e^{\langle k \rangle (z-1)}$ (Poisson degree distribution)
  - $g_1(z) = \frac{1}{\langle k \rangle} \frac{d}{dz} g_0(z) = e^{\langle k \rangle (z-1)}$ (Also Poisson!)
  - $g_{K_2} = g_0(g_1(z)) = e^{\langle k \rangle (e^{\langle k \rangle (z-1)} - 1)}$
  - $g_{K_3} = g_0(g_1(z)) = e^{\langle k \rangle (e^{\langle k \rangle (e^{\langle k \rangle (z-1)} - 1)} - 1)}$
  - …
- We cannot write a closed form solution to the distribution of K_d for general d
  - Even K_2 difficult
  - For given d and k_d we can write $P(K_d = k_d)$
  - Results are not pretty

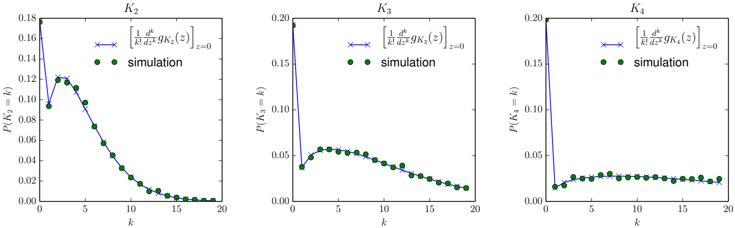
Example: Erdős-Rényi networks

- Examples for probabilities of K_d:

$$
\begin{aligned}
P(K_3 = 0) &= e^{-2+\frac{2}{e^{-\frac{2}{e^2}+2}}} \\
P(K_3 = 1) &= \frac{8}{e^2 e^{-\frac{2}{e^2}+2} e^{-\frac{2}{e^{-\frac{2}{e^2}+2}}+2}} \\
P(K_3 = 2) &= \frac{1}{e^{-\frac{2}{e^{-\frac{2}{e^2}+2}}+2}} \left(\frac{32}{e^4 e^{-\frac{4}{e^2}+4}} + \frac{16}{e^4 e^{-\frac{2}{e^2}+2}} + \frac{8}{e^2 e^{-\frac{2}{e^2}+2}} \right) \\
P(K_4 = 0) &= e^{-2+\frac{2}{e^{-\frac{2}{e^{-\frac{2}{e^2}+2}}+2}}}
\end{aligned}
$$

Example: Erdős-Rényi networks

- Result can be compared to simulations (ER network with $N = 10^6$ and $\langle k \rangle = 2$)



Solving for component size distributions

- Solving the Galton-Watson process gives us a criterion for the percolation threshold
- The expected number of nodes $\langle K_d \rangle$ in a BFS can be solved for configuration model
    - Accuracy of the approximation goes down when $\langle K_d \rangle$ approaches the network size
- The full distribution of the number of nodes $P(K_d = k)$ in a BFS can be difficult to solve
- Next week: solution for the component size distribution using GFs