School of Information Sciences
University of Pittsburgh

TELCOM2125: Network Science and Analysis

Konstantinos Pelechrinis

Spring 2015

Figures are taken from:
M.E.J. Newman, “Networks: An Introduction”

Part 6: Random Graphs with General Degree Distributions

Generating functions

- **Consider a probability distribution of a non-negative, integer random variable p_k**
  - E.g., the distribution of the node degree in a network
  - The (probability) generating function for the probability distribution p_k is:

$$g(z) = p_0 + p_1 z + p_2 z^2 + p_3 z^3 + \ldots = \sum_{k=0}^{\infty} p_k z^k$$

  - Hence, if we know g(z) we can recover the probability distribution:

$$p_k = \frac{1}{k!} \frac{d^k g}{dz^k}\bigg|_{z=0}$$

- **The probability distribution and the generating function are two different representations of the same quantity**

Examples

- **Consider a variable that only takes 4 values (e.g., 1, 2, 3, 4)**
  - p_k = 0 for k=0 or k>4
  - Let us further assume that p_1=0.4, p_2=0.3, p_3=0.1 and p_4=0.2
  - Then:

$$g(z) = 0.4z + 0.3z^2 + 0.1z^3 + 0.2z^4$$

- **Now let us assume that k follows a Poisson distribution:**

$$p_k = e^{-c}\frac{c^k}{k!}$$

  - Then the corresponding probability generating function is:

$$g(z) = e^{-c}\sum_{k=0}^{\infty}\frac{(cz)^k}{k!} = e^{c(z-1)}$$

Examples

- **Suppose k follows an exponential distribution:**

$$p_k = Ce^{-\lambda k}, \quad \lambda > 0 \quad C = 1 - e^{-\lambda}$$

  - Then the generating function is:

$$g(z) = (1 - e^{-\lambda}) \sum_{k=0}^{\infty} (e^{-\lambda} z)^k = \frac{e^{\lambda} - 1}{e^{\lambda} - z}$$

  - The above function converges iff $z < e^{\lambda}$
  - Given that we are only interested in the range 0≤z≤1, this holds true

Power-law distributions

- **As we have seen many real networks exhibit power-law degree distribution**
  - To reiterate, in its pure form we have:

$$p_k = Ck^{-\alpha}, \quad \alpha > 0 \quad k > 0 \quad and \quad p_0 = 0$$

  - The normalization constant is: $C\sum_{k=1}^{\infty} k^{-\alpha} = 1 \Rightarrow C = \frac{1}{\zeta(\alpha)}$
    - ✓ Where ζ(α) is the Riemman zeta function
  - Then the probability generating function is:

$$g(z) = \frac{1}{\zeta(\alpha)} \sum_{k=1}^{\infty} k^{-\alpha} z^k = \frac{Li_\alpha(z)}{\zeta(\alpha)}$$

    - ✓ Where Li_α is the polylogarithm of z

$$\frac{\partial Li_\alpha(z)}{\partial z} = \frac{\partial}{\partial z} \sum_{k=1}^{\infty} k^{-\alpha} z^k = \sum_{k=1}^{\infty} k^{-(\alpha-1)} z^{k-1} = \frac{Li_{\alpha-1}(z)}{z}$$

Power-law distribution

- **Real networks, as we have seen, do not follow power-law over all the values of k**
  - Power-law is generally followed at the tail of the distribution after a cut-off value k_{min}
  - In this case the more accurate generating function is:

$$g(z) = Q_{k_{\min}-1}(z) + C \sum_{k=k_{\min}}^{\infty} k^{-\alpha} z^k$$

Lerch transcendent

  - $Q_n(z)$ is a polynomial in z of degree n
  - C is the normalizing constant

Normalization and moments

- **If we set z=1 at the generating function we get:**

$$g(1) = \sum_{k=0}^{\infty} p_k$$

  - If the underlying probability distribution is normalized to unity, g(1)=1
    - ✓ This is not always the case – recall the distribution of small components for a random graph
- **The derivative of the generating function is:** $g'(z) = \sum_{k=0}^{\infty} k p_k z^{k-1}$
  - Evaluating at z=1 we get:

$$g'(1) = \sum_{k=0}^{\infty} k p_k = \langle k \rangle$$

Normalization and moments

- **The previous result can be generalized in higher moments of the probability distribution:**

$$\langle k^m \rangle = \left[\left(z \frac{d}{dz} \right)^m g(z) \right]_{z=1} = \frac{d^m g}{d(\ln z)^m} \bigg|_{z=1}$$

- **This is convenient since many times we first calculate the generating function and hence, we can compute interesting quantities directly from g(z)**
  - Possible even in cases we do not have a closed form for g(z)

Powers of generating functions

- **A very important property of generating functions is related with their powers**

- **In particular let us assume g(z) that represents the probability distribution of k (e.g., degree)**
  - *If we draw m numbers – independently - from this distribution, the generating function of this sum is the m-th power of g(z)!*
  - This is a very important property that we is extensively used in derivations for the configuration model and beyond

Powers of generating functions

- **Given that the m numbers are drawn independently from the distribution, the probability that they take a particular set of values $\{k_i\}$ is:** $\prod_i p_{k_i}$
  - Hence the probability π_s that they will sum up to s, is given if we consider all the possible combinations of k_i values that sum up to s:

$$\pi_s = \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} \delta(s, \sum_i k_i) \prod_{i=1}^{m} p_{k_i}$$

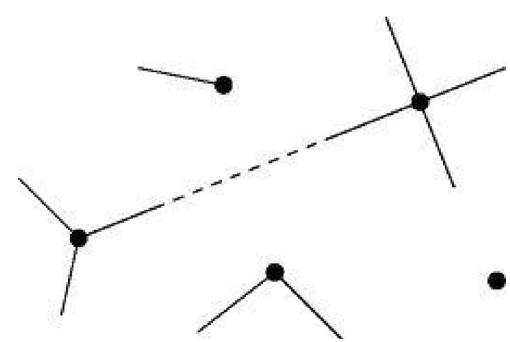
  - Substituting to the generation function h(z) for π_s:

$$\begin{aligned} h(z) &= \sum_{s=0}^{\infty} \pi_s z^s \\ &= \sum_{s=0}^{\infty} z^s \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} \delta(s, \sum_i k_i) \prod_{i=1}^{m} p_{k_i} \\ &= \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} z^{\sum_i k_i} \prod_{i=1}^{m} p_{k_i} \\ &= \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} \prod_{i=1}^{m} p_{k_i} z^{k_i} = \left[\sum_{k=0}^{\infty} p_k z^k \right]^m \\ &= [g(z)]^m. \end{aligned}$$

Configuration model

- **In the configuration model we provide a given degree <u>sequence</u>**
  - This sequence has the exact degree of every node in the network
    - ✓ The number of edges in the network is fixed → Generalization of G(n,m)

- **Each vertex i can be thought as having k_i "stubs" of edges**
  - We choose at each step two stubs uniformly at random from the still available ones and connect them with an edge



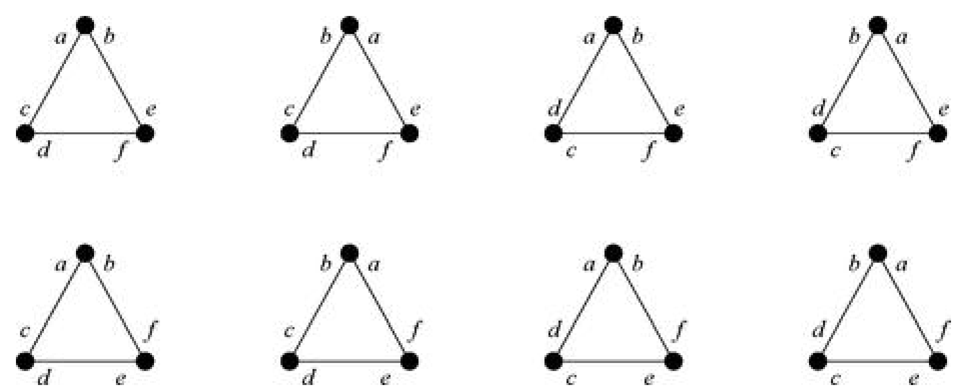
Configuration model

- **The graph created from running once the above process is just one possible matching of stubs**
  - All possible matchings appear with equal probabilities
  - Hence, the configuration model can be thought as the ensemble in which each matching with the chosen degree sequence appear with equal probability

- **However, configuration model has a few catches**
  - The sum of the degrees need to be even
  - Self-edges and multi-edges might appear
    - ✓ If we modify the process to remove these edges then the network is no longer drawn uniformly from the set of possible matchings
    - ✓ It can be shown that the density of these edges tends to 0

Configuration model

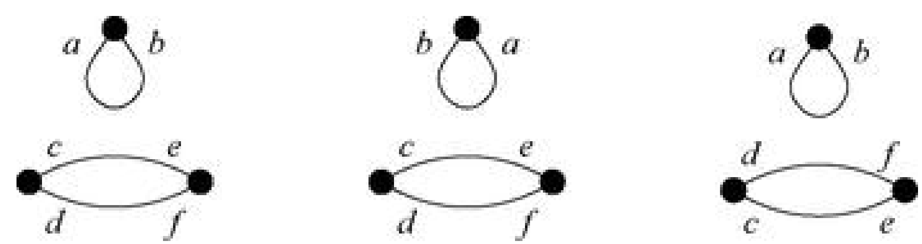
- **While all matchings appear with equal probabilities, not all networks appear with equal probability!**
  - One network might correspond to multiple matchings
  - We can create all the matchings for a given network by permuting the stubs at each vertex in every possible way
    - ✓ Total number of matches for a given network: $N(\{k_i\}) = \prod_i k_i!$
      - Independent of the actual network
  - With $\Omega(\{k_i\})$ being the number of total matchings, each network indeed appears with equal probability N/Ω



Configuration model

- **However in the above we have assumed only simple edges**
  - When we add multi- or self-edges things become more complicated
    - ✓ Not all permutations of stubs correspond to different matchings
- **Two multi-edges whose stubs are permuted simultaneously result in the same maching**
  - Total number of matchings is reduced by $A_{ij}!$
    - ✓ A_{ij} is the multiplicity of the edge (i,j)
- **For self-edges there is a further factor of 2 because the interchange of the two edges does not generate new matching**

$$N = \frac{\prod_i k_i!}{(\prod_{i<j} A_{ij}!)(\prod_i A_{ii}!!)}$$



Configuration model

- **The total probability of a network is still N/Ω but now we have N to depend on the structure of the network itself**
  - Hence, different networks have different probabilities to appear

- **In the limit of large n though, the density of multi- and self-edges is zero and hence, the variations in the above probabilities are expected to be small**

A slight modification

- **Some times we might be given the degree distribution p_k rather than the degree sequence**

- **In this case we draw a specific degree sequence from this distribution and work just as above**

- **The two models are not very different**
  - The crucial parameter that comes into calculations is the fraction of nodes with degree k
    - ✓ In the extended model this fraction is p_k in the limit of large n
    - ✓ In the standard configuration model this fraction can be directly calculated from the degree sequence given

Edge probability

- **What is the probability of an edge between nodes i and j?**
  - There are k_i stubs at node i and k_j at j
    - ✓ The probability that one of the k_i stubs of node i connects with one of the stubs of node j is $k_j/(2m-1)$
    - ✓ Since there are k_i possible stubs for vertex i the overall probability is:

$$p_{ij} = \frac{k_i k_j}{2m-1} \cong \frac{k_i k_j}{2m}$$

  - The above formula is the expected number of edges between nodes i and j but in the limit of large m the probability and mean values become equal (why??)

Edge probability

- **What is the probability of having a second edge between i and j?**

$$p_{ij,2} = \frac{k_i k_j (k_i - 1)(k_j - 1)}{(2m)^2}$$

  - This is basically the probability that there is a multi-edge between vertices i and j

- **Summing over all possible pairs of nodes we can get the expected number of multi-edges in the network:**

$$\frac{1}{2(2m)^2} \sum_{ij} k_i k_j (k_i - 1)(k_j - 1) = \frac{1}{2}\left[\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}\right]^2$$

  - The expected number of multi-edges remains constant as the network grows larger, given that the moments are constant

Edge probability

- **What is the probability of a self edge ?**
  - The possible number of pairs between the k_j stubs of node j is $\frac{1}{2}k_j(k_j-1)$. Hence:

$$p_{jj} = \frac{\frac{1}{2}k_j(k_j-1)}{2m-1} \cong \frac{k_j(k_j-1)}{4m}$$

- **Summing over all nodes we get the expected number of self edges:**

$$\sum_i p_{ii} = \frac{\langle k^2\rangle - \langle k\rangle}{2\langle k\rangle}$$

Edge probability

- **What is the expected number n_{ij} of common neighbors between nodes i and j?**
  - Consider node i → Probability that i is connected with l: $k_i k_l/2m$
  - Probability that j is connected to l (after node i connected to it): $k_j(k_l-1)/(2m-1)$
  - Hence:

$$n_{ij} = \sum_l \frac{k_i k_l}{2m} \frac{k_j (k_l - 1)}{2m} = \frac{k_i k_j}{2m} \frac{\sum_l k_l (k_l - 1)}{n \langle k \rangle} = p_{ij} \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}$$

Random graphs with given expected degree

- **The configuration model can be thought as an extension of the G(n,m) random graph model**

- **Alternatively, we can assign to each vertex i in the graph a parameter c_i and create an edge between two nodes i and j with a probability $c_i c_j/2m$**
  - We need to allow for self- and multi-edges to keep the model tractable
  - Hence the probability between edges i and j is:

$$p_{ij} = \begin{cases} \frac{c_i c_j}{2m}, & i \neq j \\ \frac{c_i^2}{4m}, & i = j \end{cases} \qquad \sum_i c_i = 2m$$

Random graphs with given expected degree

- **Based on the above graph generation process we have:**
  - Average number of edges in the network:

$$\sum_{i\le j} p_{ij} = \sum_{i<j} \frac{c_i c_j}{2m} + \sum_i \frac{c_i^2}{4m} = \sum_{ij} \frac{c_i c_j}{4m} = m$$

  - Average degree of node i:

$$\langle k_i \rangle = 2p_{ii} + \sum_{j\ne i} p_{ij} = \frac{c_i^2}{2m} + \sum_{j\ne i} \frac{c_i c_j}{2m} = \sum_j \frac{c_i c_j}{2m} = c_i$$

- **Hence, c_i is the average degree of node i**
  - The actual degree on a realization of the model will differ in general from c_i
  - It can be shown that the actual degree follows Poisson distribution with mean c_i (unless if c_i=0)

Random graphs with given expected degree

- **Hence in this model we specify the expected degree sequence {c_i} (and consequently the expected number of edges m), but not the actual degree sequence and number of edges**
  - This model is analogous to G(n,p)

- **The fact that the distribution of the expected degrees c_i is not the same as the distribution of the actual degrees k_i makes this model not widely used**
  - Given our will to be able to choose the actual degree distribution we will stick with the configuration model even if it is more complicated

Neighbor’s degree distribution

- **Considering the configuration model, we want to find what is the probability that the neighbor of a node has degree k**
  - In other words, we pick a vertex i and we follow one of its edges. What is the probability that the vertex at the other end of the edge has degree k?

- **Clearly it cannot be simply p_k**
  - Counter example: If the probability we are looking for was p_k it means that the probability of this neighbor vertex to have degree of zero is p_0 (which is in general non-zero). However, clearly this probability is 0!

Neighbor's degree distribution

- **Since there are k stubs at every node of degree k, there is a k/(2m-1) probability the edge we follow to end to a specific node of degree k**
  - In the limit of large network this probability can be simplified to k/2m
  - The total number of nodes with degree k is np_k
  - Hence the probability that a neighbor of a node has degree k is:

$$\frac{k}{2m} np_k = \frac{kp_k}{\langle k \rangle}, \quad \text{since} \quad 2m = n\langle k \rangle$$

Average degree of a neighbor

- **What is the average degree of an individual's network neighbor?**
  - We have the degree probability of a neighbor, so we simply need to sum over it:

$$\text{average degree of a neighbor} = \sum_k k \frac{kp_k}{\langle k \rangle} = \frac{\langle k^2 \rangle}{\langle k \rangle}$$

  - Given that: $\frac{\langle k^2 \rangle}{\langle k \rangle} - \langle k \rangle = \frac{1}{\langle k \rangle}\left(\langle k^2 \rangle - \langle k \rangle^2\right) = \frac{\sigma_k^2}{\langle k \rangle} \geq 0$

    ✓ **Your friends have more friends than you!** (Friendship paradox)
  - Even though this result is derived using the configuration model it has been shown to hold true in real networks as well!

Excess degree distribution

- **In many of the calculations that will follow we want to know how many edges the neighbor node has except the one that connects it to the initial vertex**

- **The number of edges attached to a vertex other than the edge we arrived along is called *excess degree* q_k**
  - The excess degree is 1 less than the actual degree. Hence:

$$q_k = \frac{(k+1)p_{k+1}}{\langle k \rangle}$$

Clustering coefficient

- **Recall that clustering coefficient is the probability that two nodes with a common neighbor are neighbors themselves**
  - Consider node u that has at least two neighbors, i and j
  - If the excess degrees of i and j are k_i and k_j respectively, then the probability that they are connected with an edge is $k_i k_j/2m$
  - Averaging over the excess distribution and both i and j we get:

$$
\begin{aligned}
C &= \sum_{k_i,k_j=0}^{\infty} q_{k_i} q_{k_j} \frac{k_i k_j}{2m} = \frac{1}{2m}\left[\sum_{k=0}^{\infty} k q_k\right]^2 \\
&= \frac{1}{2m\langle k\rangle^2}\left[\sum_{k=0}^{\infty} k(k+1) p_{k+1}\right]^2 \\
&= \frac{1}{2m\langle k\rangle^2}\left[\sum_{k=0}^{\infty} (k-1) k p_k\right]^2 \\
&= \frac{1}{n}\frac{\left[\langle k^2\rangle - \langle k\rangle\right]^2}{\langle k\rangle^3},
\end{aligned}
$$

Clustering coefficient

- **As with the Poisson random graph, the clustering coefficient of the configuration model goes as n^{-1} and vanishes in the limit of large networks**
  - Not very promising model for real networks with large clustering coefficient

- **However, in the enumerator of the expression, there is $\langle k^2 \rangle$, which can be large in some networks depending on the degree distribution**
  - E.g., power law

Generating functions for degree distributions

- **We will denote the generating functions for the degree distribution and the excess degree distribution as $g_0(z)$ and $g_1(z)$ respectively**

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

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

- **We can get the relation between the two generating functions:**

$$g_1(z) = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} (k+1) p_{k+1} z^k = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} k p_k z^{k-1} = \frac{1}{\langle k \rangle} \frac{dg_o}{dz} = \frac{g_o'(z)}{g_0'(1)}$$

  - In order to find the excess degree distribution we simply need to find the degree distribution

Generating functions for degree distributions

- **Let us assume that the degree distribution follows a Poisson distribution**

$$p_k = e^{-c}\frac{c^k}{k!}$$

$$g_o(z) = e^{c(z-1)} \Rightarrow g_1(z) = \frac{ce^{c(z-1)}}{c} = e^{c(z-1)} = g_o(z)$$

- **The two generating functions are identical**
  - This is one reason why calculations on Poisson random graph are relatively straightforward

Generating functions for degree distributions

- **Let us assume a power law degree distribution:**

$$p_k = \frac{k^{\alpha}}{\zeta(\alpha)} \Rightarrow g_0(z) = \frac{Li_{\alpha}(z)}{\zeta(\alpha)}$$

$$g_1(z) = \frac{Li_{\alpha-1}(z)}{zLi_{\alpha-1}(z-1)} = \frac{Li_{\alpha-1}(z)}{z\zeta(\alpha-1)}$$

Number of second neighbors of a vertex

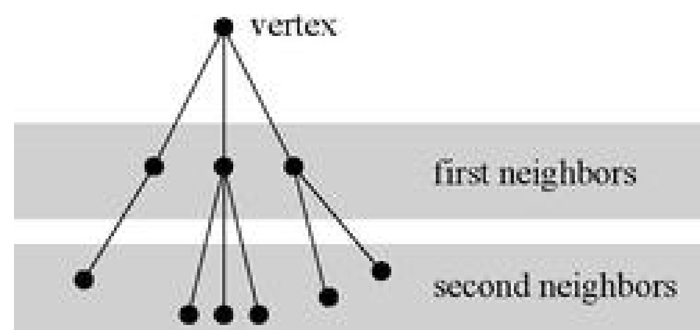
- **Let us calculate the probability that a vertex has exactly k second neighbors**

$$p_k^{(2)} = \sum_{m=0}^{\infty} p_m P^{(2)}(k \mid m)$$

  - $P^{(2)}(k|m)$ is the conditional probability of having k second neighbors given that we have m direct neighbors

- **The number of second neighbors of a vertex is essentially the sum of the excessive degrees of its first neighbors**
  - The probability that the excess degree of the first neighbors is $j_1,\dots,j_m$ is:

$$\prod_{r=1}^{m} q_{j_r}$$



Number of second neighbors of a vertex

- **Summing over all sets of values that sum up to m we get:**

$$P^{(2)}(k \mid m) = \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \delta(k, \sum_r j_r) \prod_{r=1}^{m} q_{j_r}$$

- **Therefore**

$$p_k^{(2)} = \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \sum_{j_2=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \delta(k, \sum_r j_r) \prod_{r=1}^{m} q_{j_r}$$

- **Using the probability generator function $g^{(2)}(z)$ we get:**

$$\begin{aligned} g^{(2)}(z) &= \sum_{k=0}^{\infty} p_k^{(2)} z^k \\ &= \sum_{k=0}^{\infty} z^k \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \delta(k, \textstyle\sum_{r=1}^{m} j_r) \prod_{r=1}^{m} q_{j_r} \\ &= \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} z^{\sum_{r=1}^{m} j_r} \prod_{r=1}^{m} q_{j_r} \\ &= \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \prod_{r=1}^{m} q_{j_r} z^{j_r} \\ &= \sum_{m=0}^{\infty} p_m \left[\sum_{j=0}^{\infty} q_j z^j \right]^m . \end{aligned}$$

Number of second neighbors of a vertex

- The quantity in brackets is the probability generator function of q_k

$$g^{(2)}(z) = \sum_{m=0}^{\infty} p_m (g_1(z))^m = g_0(g_1(z))$$

- The above equation reveals that once we know the generating functions for the vertices degrees and the vertices excessive degree we can find the probability distribution of the second neighbors

- Is there an easier way to derive the above result ?

Number of d-hop neighbors

- **Similarly we can calculate the number of 3-hop neighbors**
  - Assuming m second neighbors (2-hop neighbors), the number of 3-hop neighbors is the sum of the excess degree of each of the second neighbors
    - ✓ $P^{(3)}(k|m)$ is the probability of having k 3-hop neighbors, given that we have m 2-hop neighbors
      - Similar to above $P^{(3)}(k|m)$ has generating function $[g_1(z)]^2$

$$
\begin{aligned}
g^{(3)}(z) &= \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} p_m^{(2)} P^{(3)}(k|m) z^k = \sum_{m=0}^{\infty} p_m^{(2)} \sum_{k=0}^{\infty} P^{(3)}(k|m) z^k \\
&= \sum_{m=0}^{\infty} p_m^{(2)} [g_1(z)]^m = g^{(2)}(g_1(z)) \\
&= g_0(g_1(g_1(z))).
\end{aligned}
$$

Number of d-hop neighbors

- **This can generalize to d-hop distance neighbors:**

$$g^{(d)}(z) = g_0(\underbrace{g_1(\dots g_1(z)\dots)}_{d-1})$$

- **The above holds true for all distances d in an infinite graph**

- **At a finite graph, it holds true for small values of d**

- **It is difficult to use the above equation to obtain closed forms for the probabilities of the size of d-hop neighborhoods**
  - We can calculate averages though

Average number of d-hop neighbors

- **What is the average size of the 2-hop neighborhood?**
  - We need to evaluate the derivative of $g^{(2)}(z)$ at z=1

$$\frac{dg^{(2)}}{dz} = g_0'(g_1(z))g_1'(z)$$

  - $g_1(1)=1$ and hence the average number of second neighbors is $c_2=g'_0(1)g'_1(1)$
    - ✓ $g'_0(1)=\langle k\rangle$ and

$$\begin{aligned} g_1'(1) &= \sum_{k=0}^{\infty} k q_k \\ &= \frac{1}{\langle k\rangle}\sum_{k=0}^{\infty} k(k+1)p_{k+1} = \frac{1}{\langle k\rangle}\sum_{k=0}^{\infty}(k-1)kp_k \\ &= \frac{1}{\langle k\rangle}(\langle k^2\rangle - \langle k\rangle). \end{aligned}$$

$$c_2 = \langle k^2\rangle - \langle k\rangle$$

Average number of d-hop neighbors

- **The average number of d-hop neighbors is given by:**

$$c_d = \frac{dg^{(d)}}{dz}\Big|_{z=1} = g^{(d-1)'}(g_1(z))g_1^{'}(z)\Big|_{z=1} = g^{(d-1)'}(1)g_1^{'}(1) = c_{d-1}g_1^{'}(1), \quad g_1^{'}(1) = \frac{c_2}{g_o^{'}(1)} = \frac{c_2}{\langle k \rangle} = \frac{c_2}{c_1}$$

- **Hence,**

$$c_d = c_{d-1}\frac{c_2}{c_1} = \left(\frac{c_2}{c_1}\right)^{d-1} c_1$$

- **The average number of neighbors at distance d increases or falls exponentially to d**
  - If this number increase then we must have a giant component
  - Hence, the configuration model exhibits a giant component iff $c_2 > c_1$, which can be written as:

$$\langle k^2 \rangle - 2\langle k \rangle > 0$$

Giant component

- **Given that π_s is the probability that a randomly selected node belongs to a small component of size s, the probability that a randomly chosen node belongs to a small component is:** $\sum_s \pi_s = h_0(1)$
- **Hence, the probability that a node belongs to the giant component is $S = 1 - h_0(1) = 1 - g_0(h_1(1))$**
  - Note that $h_0(1)$ is not necessarily 1 as with most probability generator functions
  - Given that $h_1(1)=g_1(h_1(1)) \rightarrow S=1-g_0(g_1(h_1(1)))$
- **Setting $h_1(1)=u$ we get**
  - $u=g_1(u)$ and hence,
  - $S=1-g_o(g_1(u))=1-g_0(u)$

Giant component

- **For the above equations it is obvious that u is a fixed point of $g_1(z)$**
  - One trivial fixed point is z=1, since $g_1(1)=1$
  - With u=1 though, we have $S=1-g_0(1)=0$, which corresponds to the case we do not have giant component
  - Hence, if there is to be a giant component there must be at least one more fixed point of $g_1(z)$
- **What is the physical interpretation of u?**

$$u = h_1(1) = \sum_s \rho_s$$

  - ρ_s is the probability that a vertex at the end of any edge belongs to a small component of size s
    - ✓ Hence, the above sum is the total probability that such a vertex does not belong to the giant component

Graphical solution

- **When we can find the fixed point of g_1 everything becomes easier**
  - However, most of the times this is not possible
    - ✓ Graphical solution

- **$g_1(z)$ is proportional to the probabilities q_k and hence for $z \geq 0$ is in general positive**
  - Furthermore, its derivatives are also proportional to q_k and hence are in general positive
  - Thus, $g_1(z)$ is positive, increase and upward concave

Graphical solution

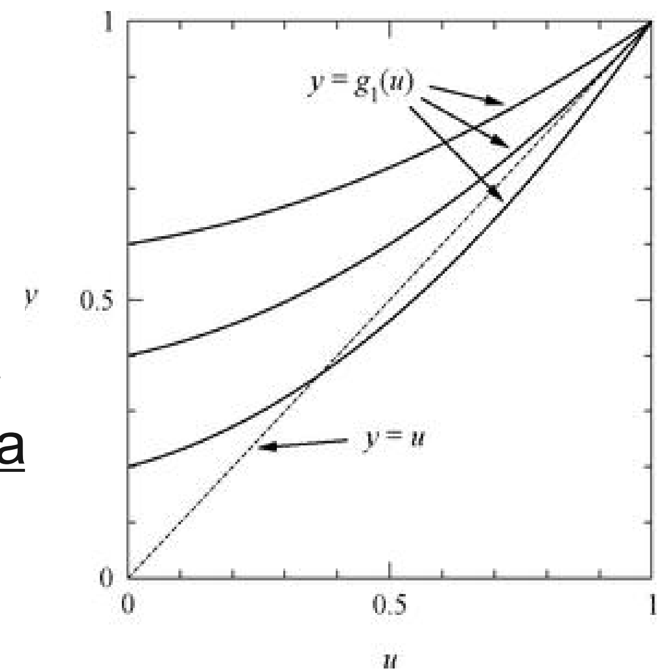
- In order for g_1 to have another fixed point u<1, its derivative at u=1 needs to be greater than 1

$$g_1'(1) = \sum_{k=0}^{\infty} k q_k = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} k(k+1) p_{k+1} = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} (k-1) k p_k = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}$$

- In order for the derivative at u=1 to be greater than 1 it needs to hold:

$$\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} > 1 \Leftrightarrow \langle k^2 \rangle - 2\langle k \rangle > 0$$

  - This is exactly the condition that we saw previously for the presence of a giant component
  - Hence, there is a giant component iff there is a fixed point u<1 for g_1



Mean component sizes

- **Using an approach similar to that with the Poisson random graph we can calculate some average quantities**

- **The mean size of the component of a randomly chosen vertex is given by:**

$$\langle s \rangle = \frac{\sum_s s\pi_s}{\sum_s \pi_s} = \frac{h_0'(1)}{1-S} = \frac{h_0'(2)}{g_0(u)}$$

  - Eventually, after some calculations we get: $\langle s \rangle = 1 + \dfrac{g_0'(1)u^2}{g_0(u)[1-g_1'(u)]}$

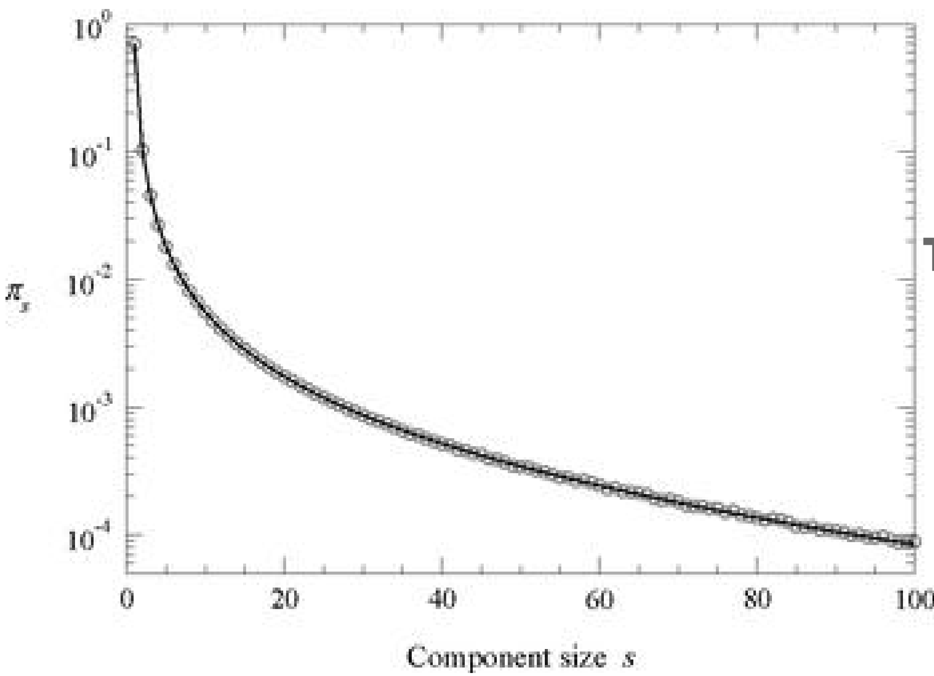
- **As with the random Poisson graph the above calculation is biased**
  - Following similar calculations we get the actual average small component size:

$$R = \frac{2}{2 - \frac{\langle k \rangle u^2}{1-S}}$$

Complete distribution of small component sizes

$$\pi_s = \frac{\langle k \rangle}{(s-1)!}\left[\frac{d^{s-2}}{dz^{s-2}}[g_1(z)]^2\right]_{z=0}, \quad s > 1$$

$$\pi_1 = p_0$$



π_s for the configuration model with exponential degree distribution with λ=1.2

Random graphs with power law degree

- **Let's start with a pure power law:**

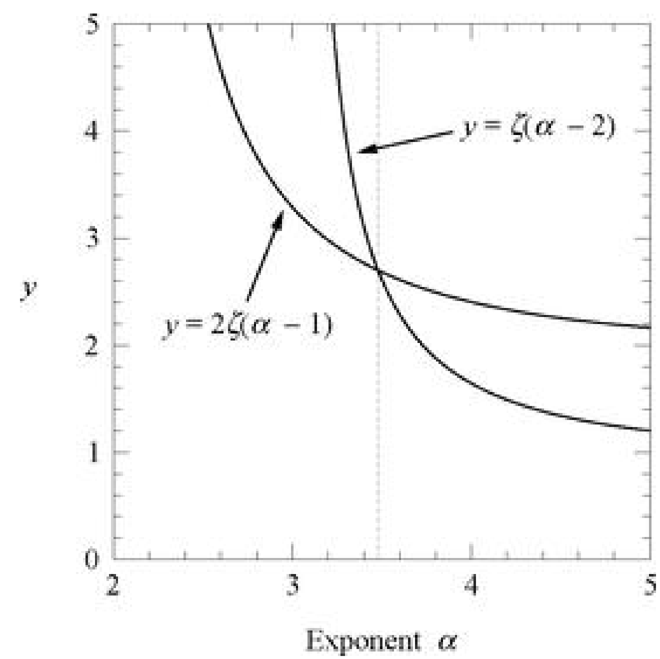
$$p_k = \begin{cases} 0 & \text{for } k=0 \\ \frac{k^{-\alpha}}{\zeta(\alpha)} & \text{for } k \geq 1 \end{cases}$$

- **A giant component exists iff [<k^2>-2<k>] > 0**

$$\langle k \rangle = \sum_{k=0}^{\infty} k p_k = \sum_{k=1}^{\infty} \frac{k^{-\alpha+1}}{\zeta(\alpha)} = \frac{\zeta(\alpha-1)}{\zeta(\alpha)}$$

$$\langle k^2 \rangle = \sum_{k=0}^{\infty} k^2 p_k = \sum_{k=1}^{\infty} \frac{k^{-\alpha+2}}{\zeta(\alpha)} = \frac{\zeta(\alpha-2)}{\zeta(\alpha)}$$

α<3.4788…



Random graphs with power law degree

- **The above result is of little practical importance since rarely we have a pure power law degree distribution**
  - We have seen that a distribution that follows a power law at its tail will have a finite $\langle k^2 \rangle$ iff $\alpha>3$, and a finite $\langle k \rangle$ iff $\alpha>2$
    - ✓ Hence, if $2<\alpha\leq3$ → a giant component always exists
    - ✓ When $\alpha>3$ → a giant component might or might not exist
    - ✓ When $\alpha\leq2$ → a giant component always exists

Random graphs with power law degree

- **What is the size S of the giant component when one exists?**
  - Recall, $S=1-g_0(u)$, where u is a fixed point of g_1
  - For a pure power law we have: $g_1(z) = \frac{Li_{\alpha-1}(z)}{z\zeta(\alpha-1)}$
  - Hence, $u = \frac{Li_{\alpha-1}(u)}{u\zeta(\alpha-1)} = \frac{\sum_{k=1}^{\infty} k^{-\alpha+1}u^k}{u\zeta(\alpha-1)} = \frac{\sum_{k=0}^{\infty} (k+1)^{-\alpha+1}u^k}{\zeta(\alpha-1)}$
  - The enumerator is strictly positive for non-negative values of u
    - ✓ Hence, $u=0$ iff $\zeta(\alpha-1)$ diverges
    - ✓ $\zeta(\alpha-1)$ diverges for $\alpha \le 2$

Random graphs with power law degree

- **Hence, for $\alpha \leq 2$, $u=0$ → There is a giant component with $S=1-g_0(0)=1-p_0=1$!**
  - The giant component fills the whole network !
  - Of course this holds true at the limit of large n

- **For $2<\alpha \leq 3.4788\ldots$ there is a giant component that fills a proportion S of the network**

- **For $\alpha>3.4788\ldots$ there is no giant component (i.e., $S=0$)**

Random graphs with power law degree