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 7: Models of Network Formation

Generative network models

- **The network models we have seen until now are mainly used to study structural properties of networks**
  - I.e., some parameters are fixed (e.g., number of nodes, number of edges, degree distribution etc.) and we study the properties of the graph (e.g., path lengths, component sizes etc.)
- **There are other network models that model the mechanism that drive the network formation**
  - If the structures resemble real world structures, then this mechanism ***might*** be at work in real networks

Price Model

- **Many real world networks exhibit power law distributions**
  - What are the underlying processes ?
  - How can power laws be generated ?

- **Price was the first to propose a single and elegant model of network formation**
  - He was studying citation networks but his model was inspired by the work of Hebert Simon, an economist
  - Simon proposed and explanation for the wealth distribution
    - ✓ People who have more money already, gain more at a rate proportional to how much they already have
    - ✓ This can lead to power law distribution for the wealth
      - “Rich-get-richer”, “cumulative advantage”, “preferential attachment”

Price Model

- **Price adopted with modifications Simon's model in the context of (citation) networks**
  - Every new paper (node of the graph) cites on average *c* other papers (c is thus, the average out degree)
  - This newly appearing paper cites previously published papers at random

NOT uniformly at random

At random with probability proportional to the number of citations those previous papers have

Price Model

- **The above cannot be precisely true**
  - A newly published paper has zero citations → a strict proportionality rule would give 0 citations to this paper from then on

- **Price added a constant factor a > 0 to the citation probability**
  - Some citations come for "free"
  - An alternative interpretation of this factor is that a part of the papers that one cites are chosen uniformly at random, while the rest are chosen with a probability proportional to the citations these other papers already have

Price Model

- **How to initialize the model?**
  - At the limit of a large network the initialization does not alter any of the predictions
  - Hence, we can start with a few papers that have zero citations each
- **In summary,**
  - Price's model is a growing network
    - ✓ New nodes are added continually and never removed
  - Average out degree is c
  - The citations a paper receives are proportional to its in degree and a constant factor a
  - The model allows for multi-edges but as with random graphs they do not affect the results at the limit of large network

Price Model

- **Price model creates acyclic graphs**
  - This aligns with the initial goal to model citation networks
  - What about other directed networks that are not acyclic? E.g., the Web

- **What is the in-degree of vertices in the price model?**
  - We denote for simplicity the in-degree of vertex j as q_j
  - Let $p_q(n)$ be the fraction of vertices that have in-degree q when the total number of vertices in the network are n
  - What happens when a new vertex is added in the network?

Price Model

- **This new vertex will create new citations/edges**

- **The probability that a new edge points to a node i is proportional to q_i+a:**

$$\frac{q_i+a}{\sum_i (q_i+a)} = \frac{q_i+a}{n\langle q\rangle + na} = \frac{q_i+a}{n(c+a)}$$

  - This new vertex has c citations → the expected number of new in edges of node i is c times the above quantity

- **The expected number of new edges pointing to vertices of in-degree q is:**

$$np_q(n)\times c\times\frac{q+a}{n(c+a)} = \frac{c(q+a)}{c+a}p_q(n)$$

Price Model

- **Let's calculate what is the number of vertices with degree q after the addition of this new vertex**
  - A node with in-degree q can appear because a node with degree q-1 received a new citation
  - However, a node with in-degree can disappear because it received a new citation and hence, its in-degree is now q+1

- **Hence the expected number of nodes with in-degree q when we add one vertex (i.e., total nodes n+1) is:**

$$(n+1)p_q(n+1) = np_q(n) + \frac{c(q-1+a)}{c+a} p_{q-1}(n) - \frac{c(q+a)}{c+a} p_q(n)$$

Master equation

Price Model

- **The master equation we saw previously does not hold true as is for the case of q=0**
  - There are no vertices of lower in-degree so the second term cannot appear
  - However, when we add a new vertex its in-degree is by default 0 → we add one node of degree 0

$$(n+1)p_0(n+1) = np_0(n) + 1 - \frac{ca}{c+a} p_0(n)$$

- **Let's examine the asymptotic behavior of the distribution**
  - Assuming that it indeed converges for large n, we use the shorthand $p_q = p_q(\infty)$

Price Model

- **Then we get:**

$$p_q = \frac{c}{c+a}[(q-1+a)p_{q-1} - (q+a)p_q] \qquad \text{for } q \geq 1,$$

$$p_0 = 1 - \frac{ca}{c+a}p_0 \qquad \text{for } q = 0.$$

- **Re-writing the above equations we can get:**

$$p_0 = \frac{1+a/c}{a+1+a/c}, \quad p_q = \frac{q+a-1}{q+a+1+a/c}p_{q-1}$$

- **After some involved (but only algebraic) calculations p_q can be written as:**

$$p_q = \frac{B(q+a, 2+a/c)}{B(a, 1+a/c)}$$

  - B(x,y) is the Euler's beta function

Price Model

- **It can further been shown that:**

$$B(x,y) \cong x^{-y}\Gamma(y)$$

  - Where $\Gamma(y)$ is the gamma function
  - Hence for large x, B(x,y) falls off as a power law with exponent y

- **Using the above results we can see that for large values of in-degree: $p_q \sim (q+a)^{-\alpha}$ or simply $p_q \sim q^{-\alpha}$**
  - $\alpha=2+(a/c)$

- **While the Price model does not consider many aspects of the citation process, even with its overly simplistic assumptions can reproduce structures similar to the ones emerging in real life**

Price Model – The constant factor a

- **Based on Price model the probability that an outgoing edge attaches to vertex i is:** $\theta_i = \dfrac{q_i + a}{n(c+a)}$

- **Let's consider a slightly different process with which we do the following:**
  - With probability φ we attach the edge to a vertex chosen strictly in proportion to its current in-degree: $\dfrac{q_i}{\sum_j q_j} = \dfrac{q_i}{nc}$
  - With probability 1-φ , we attach the edge to a vertex chosen uniformly at random from all n nodes, that is, the probability for each vertex getting the edge is 1/n

Price Model – The constant factor a

- **Hence the total probability of attaching to vertex i via this process is:**

$$\theta_i' = \varphi \frac{q_i}{nc} + (1-\varphi)\frac{1}{n}$$

- **By choosing φ=c/(c+a) we get:**

$$\theta_i' = \frac{c}{c+a}\frac{q_i}{nc} + (1 - \frac{c}{c+a})\frac{1}{n} = \frac{q_i + a}{n(c+a)} = \theta_i$$

  - So an alternative way for running the Price model is:
    - ✓ With probability c/(c+a) choose a vertex in strict proportion to in-degree. Otherwise choose a vertex uniformly at random from the set of all vertices

Barabasi-Albert Model

- **The most well-known generative model for power law distribution is that of Barabasi-Albert**
  - Also known as “preferential attachment”
  - Deals with undirected networks

- **Vertices are added one by one**
  - Each new vertex creates exactly c edges
  - Each edge is attached to a vertex i randomly with a probability directly proportional to i’s degree k_i
  - Since vertices and edges are never removed the minimum degree in the network is c
    - ✓ For this we would need to initialize the network with m_0 nodes with c edges each ($c < m_0$)

Barabasi-Albert Model

- **Barabasi-Albert (BA) model is a special case of Price model**
- **Let's assume that each edge has a direction from the newly created vertex to the old vertex chosen to be attached**
  - The out-degree of every vertex is exactly c
  - The total/undirected degree of vertex i is then $k_i = q_i + c$, where q_i are the edges that point to i
  - Given that the probability in the BA of an edge attaching to a node is simply proportional to k_i, this means that it is proportional to $q_i + c$
    - ✓ This is the Price model with a=c

Barabasi-Albert Model

- **Hence the distribution of the in-degree for this "directed" version of the BA networks is:**

$$p_q = \frac{B(q+c,3)}{B(c,2)}$$

- **Substituting q+c with k we get the degree distribution for the undirected BA network:**

$$p_k = \begin{cases} \frac{B(k,3)}{B(c,2)}, & k \ge c \\ 0, & k < c \end{cases} \Rightarrow p_k = \frac{2c(c+1)}{k(k+1)(k+2)}, k \ge c$$

  - In the limit of large k, $p_k \sim k^{-3}$
    - ✓ Power law with exponent $\alpha=3$

Barabasi-Albert Model

- **BA model is**
  - Simple – no requirement for setting parameter a
  - Degree distribution can be expressed without the need of beta or gamma functions

- **However it comes with the price of being able to create power law degree distribution with only one exponent (α=3)**

Effect of time creation on degree distribution

- **Consider again the general preferential attachment model**

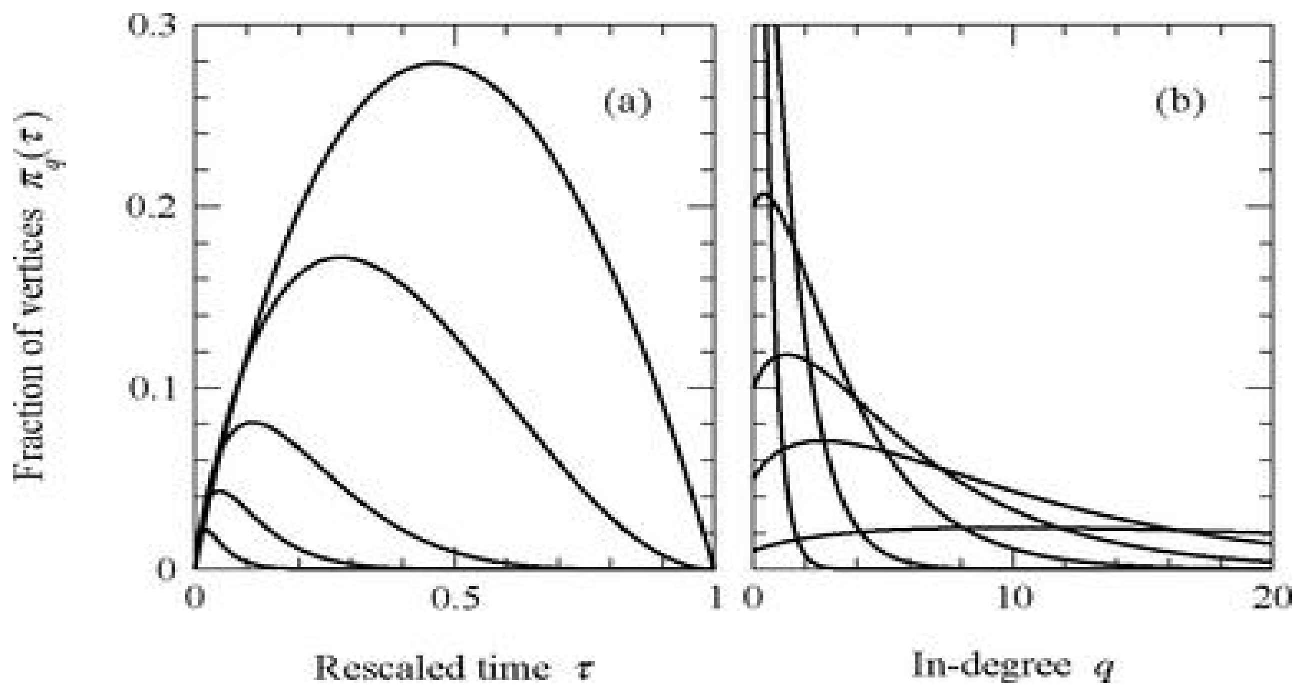
- **$p_q(t,n)$ is the probability that a node created in time t, will have degree q after at time n**
  - Time is discrete and increases every time a new node is created
    - ✓ t=1 corresponds to the creation of the first node and so on

- **In general, it is easier to use instead of the above probability its density**
  - We rescale the time using: $\tau = \frac{t}{n}$
  - Then we have $\pi_q(\tau,n)$ being the density, that is, $\pi_q(\tau,n)d\tau$ is the probability that vertices created in the interval $[\tau, \tau+d\tau]$ have degree q after absolute time n

Effect of time creation on degree distribution

- **Using again the master equation method we can obtain a differential equation for the density function**

- **Eventually we get:** $$\pi_q(\tau) = \frac{\Gamma(q+a)}{\Gamma(q+1)\Gamma(a)} \tau^{ca/(c+a)} (1-\tau^{c/(c+a)})^q$$



Effect of time creation on degree distribution

- **As we can see nodes of a specific degree are concentrated around a particular era in the growth of the graph**
  - The higher the degree the earlier this era is

- **If we focus now on nodes that were created around the same era in the growth of the graph, we see that their degree distribution differs**
  - Older nodes exhibit flat degree distribution, while as we move to younger nodes we see an exponential decay
  - *There is no power law if we focus on nodes that were created around the same time*
    - ✓ *However, when we integrate over all times τ we see the power law*

Extensions of preferential attachment

- **Price and BA models include many oversimplified assumptions for the way networks grow**
  - In particular, they assume that nodes and edges are only created (never deleted) and a vertex can initiate new edges only at the time of creation
- **While these assumptions might hold with good approximation for citation networks they do not hold true for other types of networks**
  - In Web links are not permanent and can be created at any time
  - Entire webpages can also disappear (vertex removal)
  - Why the preferential attachment should be linear in the degree?

Addition of extra edges

- **While in citation networks no new edges can be created by a vertex after the time of the node generation this is not true for the majority of the networks**

- **A network evolution model that can deal with the creation of new edges is the following:**
  - At each step the new node creates exactly c new edges which attach to other vertices with probability proportional to degree k
  - In addition, at each step some number w of extra edges are added to the network with both ends attaching to vertices chosen proportional to their degree
    - ✓ Thus, when the network has n vertices it will have n(c+w) edges

Addition of extra edges

- **Every new node brings c+2w possibilities for every existing node to attach to a new edge**

- **The normalization factor for the attaching probabilities is the sum of the degree of all nodes**
  - This is equal to twice the number of edges, that is, 2n(c+w)

- **The number of nodes with degree k that will attract one of the new edges is:**

$$np_k(n) \times (c+2w) \times \frac{k}{2n(c+w)} = \frac{c+2w}{2(c+w)} kp_k(n)$$

  - $p_k(n)$ is the fraction of nodes with degree k when the network has n vertices

Addition of extra edges

- **We can then write the master equation:**

$$(n+1)p_k(n+1) = np_k(n) + \frac{c+2w}{2(c+w)}[(k-1)p_{k-1}(n) - kp_k(n)], \quad k > c$$

$$(n+1)p_c(n+1) = np_c(n) + 1 - \frac{c+2w}{2(c+w)}cp_c(n), \qquad k = c$$

- **Taking the limit of large n and solving the equations we get:**

$$p_k = \frac{B(k,a)}{B(c,a-1)}, \quad a = 2 + \frac{c}{c+2w}$$

  - Again if we take the asymptotic behavior of Beta function we see that the degree distribution has a power law tail with exponent a
    - ✓ For w=0 we get a=3 as we should have expected (BA model)
    - ✓ For w>0, we get 2<a<3

Removal of edges

- **Edges can be removed as well in many networks**
  - Here we begin with an extension of the original BA model to account for deletion of edges at each step (edge additions are still only happening at the initial creation of a vertex)
  - In particular, at each step a new vertex is created with c edges associated with him
    - ✓ These edges are attached to existing nodes with the original BA process, that is, proportional to the exiting nodes degrees
  - At each step we also remove u edges from the existing ones
    - ✓ The edges that are deleted are picked uniformly at random
    - ✓ The probability that a node of degree k_i loses one of its edges is:

$$\frac{2k_i}{\sum_i k_i}$$

Removal of edges

- **In order for the edges in the network to grow and not shrink with time it needs to hold: u<c**

- **The total number of edges after n steps is n(c-u)**

- **When writing the master equation we need to be careful since there are more ways now to get nodes of degree k**
  - The expected number of vertices of degree k that will get a new edge at the (n+1)th step is: $$np_k(n) \times c \times \frac{k}{2n(c-u)} = \frac{c}{2n(c-u)} kp_k(n)$$
  - The expected number of vertices of degree k that will lose one edge at the (n+1)th step is: $$np_k(n) \times 2u \times \frac{k}{2n(c-u)} = \frac{u}{c-u} kp_k(n)$$

Removal of edges

- **As it should be evident nodes now can have degrees less than c as well**

- **The master equation takes the following form:**

$$(n+1)p_k(n+1) = np_k(n) + \frac{c}{2(c-u)}(k-1)p_{k-1}(n) + \frac{u}{c-u}(k+1)p_{k+1}(n) - \frac{c}{2(c-u)}kp_k(n) - \frac{u}{c-u}kp_k(n), \qquad k \neq c$$

$$(n+1)p_c(n+1) = np_c(n) + 1 + \frac{c}{2(c-u)}(c-1)p_{c-1}(n) + \frac{u}{c-u}(c+1)p_{c+1}(n) - \frac{c}{2(c-u)}cp_c(n) - \frac{u}{c-u}cp_c(n), \qquad k = c$$

  - These equations can be combined to:

$$(n+1)p_k(n+1) = np_k(n) + \delta_{kc} + \frac{c}{2(c-u)}(k-1)p_{k-1}(n) + \frac{2u}{2(c-u)}(k+1)p_{k+1}(n) - \frac{c+2u}{2(c-u)}kp_k(n)$$

    - ✓ Problem arises for k=0, since it involves $p_{-1}(n)$. To overcome this problem we define $p_{-1}=0$

Removal of edges

- **This extension of BA model accounts only for removal of edges, while it allows creation of edges only during the initial creation of a vertex**

- **We can easily add to the above process the addition of w edges (on top of the c edges created by the new vertex) at every node creation instance**
  - The master equation takes the form:

$$(n+1)p_k(n+1) = np_k(n) + \delta_{kc} + \frac{c+2w}{2(c+w-u)}(k-1)p_{k-1}(n) + \frac{u}{c+w-u}(k+1)p_{k+1}(n) - \frac{c+2w+2u}{2(c+w-u)}kp_k(n)$$

  - In this case we require u<w+c

Removal of edges

- **Taking the limit of large n we can obtain the equations for p_k**
  - However, as we can observe they are different in some aspect from what we have seen until now
  - In particular, they involve probabilities for 3 degrees
    - ✓ k-1, k and k+1

- **The presence of three different degree terms makes it substantially more difficult to solve**
  - We need to make use of the generating functions and eventually we obtain a first-order linear differential equation

Removal of edges

- **The final result for a large k (and u<w+(1/2)c) is:**

$$p_k \sim \frac{\Gamma(a-1)}{(1-\gamma)^{a-1}} k^{-a}, \quad a = 2 + \frac{u-w}{c+2w-2u} \quad \gamma = \frac{2u}{c+2w}$$

- **As we can see this extension of the model provides a power law tail again**
  - The exponent can take values both greater and smaller than 2
  - However, for u=w+(1/2)c, the exponent becomes infinite
    - ✓ At this point we lose the power law behavior and the distribution becomes a stretched exponential
  - Also for u >w+(1/2)c, the solution becomes nonsensical and we need to solve the differential equation with other methods

Non-linear preferential attachement

- **In the models that we have considered until now the attachment is linear to the degree of a vertex**
  - However, there is no apparent reason for such a linear process
  - In fact, there are studies that have found that in some networks the attachment of edges is non-linear
    - ✓ Goes as some power γ of the degree different than 1

- **What is the effect of a non-linear preferential attachment process? Do we still see power laws?**
  - The answer depends on the actual form of the attachment process

Non-linear preferential attachment

- **We define an <u>attachment kernel a_k</u> as the functional *form* of the attachment probability**
  - For instance for BA model $a_k=k$

- **The attachment kernel is not the attachment probability**
  - The correctly normalized attachment probability for a vertex with degree k_i is:

$$\frac{a_{k_i}}{\sum_{k_j} a_{k_j}}$$

Non-linear preferential attachment

- **Let us consider an identical to BA model with a general attachment kernel**
  - Then, the expected number of vertices of degree k that will receive an edge during the next vertex addition is:

$$np_k(n) \times c \times \frac{a_k}{\sum_j a_{k_j}} = \frac{c}{\mu(n)} a_k p_k(n)$$

  - Where, $\mu(n) = \frac{1}{n}\sum_{i=1}^{n} a_{k_i} = \sum_k a_k p_k(n)$

- **The master equation becomes:**

$$(n+1)p_k(n+1) = np_k(n) + \frac{c}{\mu(n)}\left[a_{k-1}p_{k-1}(n) - a_k p_k(n)\right], \quad k > c$$

$$(n+1)p_c(n+1) = np_c(n) + 1 - \frac{c}{\mu(n)} a_c p_c(n), \qquad k = c$$

Non-linear preferential attachment

- **Taking the limit of large n we get:**

$$p_k = \frac{c}{\mu}[a_{k-1}p_{k-1} - a_k p_k], \quad k > c$$

$$p_c = 1 - \frac{ca_c}{\mu} p_c \qquad k = c$$

  - μ depends on the degree distribution but since it is independent of k we can still solve the above set of equations
    - ✓ We solve having p_k=f(μ) and at the end we utilize the definition of μ to get it's value

$$p_k = \frac{\mu}{ca_k}\prod_{r=c}^{k}\left[1+\frac{\mu}{ca_r}\right]^{-1}$$

$$\mu = \sum_{k=c}^{\infty} a_k p_k = \frac{\mu}{c}\sum_{k=c}^{\infty}\prod_{r=c}^{k}\left[1+\frac{\mu}{ca_r}\right]^{-1} \Rightarrow \sum_{k=c}^{\infty}\prod_{r=c}^{k}\left[1+\frac{\mu}{ca_r}\right]^{-1} = c$$

The last equation is usually hard to solve in a closed form.
However, even without a value for μ we can still get many interesting results.

Non-linear preferential attachment

- **Assuming $a_k=k^\gamma$ we can get some idea for the differences from linear preferential attachment**
  - ½ < γ < 1: $$p_k \sim k^{-\gamma}\exp\left(-\frac{\mu k^{1-\gamma}}{c(1-\gamma)}\right)$$ Falls slower than a pure exponential, but still faster than power law
    - ✓ Stretched exponential
  - 1/3 < γ < ½: $$p_k \sim k^{-\gamma}\exp\left(-\frac{\mu k^{1-\gamma}}{c(1-\gamma)}+\frac{\mu^2 k^{1-2\gamma}}{2c^2(1-2\gamma)}\right)$$
  - And similar solutions for other cases
    - ✓ For instance for γ>1 (super linear) the typical behavior is for one vertex to emerge as the leader in the network, gaining a non-zero fraction of all edges, with the rest of the vertices having small degree (almost all having degree less than some fixed constant)

Vertices of varying quality or attractiveness

- **The models we have seen until now consider the current vertex degree as the only feature that plays role in the *attractiveness* of a node obtaining new connections**

- **If we allow for variations in the intrinsic quality or attractiveness of vertices, the power law behavior is lost**
  - The degree distribution of vertices with the same attractiveness (or fitness as it is called) still follows power law distribution

Vertices of varying quality or attractiveness

- **Vertices are added as before in the BA model**
  - Each vertex i now has a fitness value η_i assigned to them at the time of creation
  - η_i is a real value
    - ✓ η is drawn from a probability distribution $\rho(\eta)$ such that the probability that the fitness value falls in $[\eta, \eta+d\eta]$ is $\rho(\eta)d\eta$

- **The kernel attachment $a_k(\eta)$ depends now on both the degree of the node and the fitness**
  - Simplest form: $a_k(\eta)=k\eta$

Vertices of varying quality or attractiveness

- Let us define $p_k(\eta,n)d\eta$ as the fraction of vertices with degree k that have fitness in [η, η+dη] when the network has n vertices

- Writing down the master equation again at the limit of large n we have:

$$\left.\begin{array}{l} p_k(\eta) = \dfrac{c}{\mu}\left[a_{k-1}(\eta)p_{k-1}(\eta) - a_k(\eta)p_k(\eta)\right], \quad k > c \\[2ex] p_c(\eta) = \rho(\eta) - \dfrac{ca_c(\eta)}{\mu}p_c(\eta), \quad k = c \\[2ex] \mu = \dfrac{1}{n}\displaystyle\sum_{i=1}^{n} a_{k_i} = \sum_{k=c}^{\infty}\int_{-\infty}^{\infty} a_k(\eta)p_k(\eta)\,d\eta \end{array}\right\} \Rightarrow p_k(\eta) = \rho(\eta)\frac{\mu}{ca_k(\eta)}\prod_{r=c}^{k}\left[1 + \frac{\mu}{ca_r(\eta)}\right]^{-1}$$

Vertices of varying quality or attractiveness

- **If $a_k(\eta)=k\eta$ we have:**

$$p_k(\eta) = \rho(\eta)\frac{B(k, 1+\mu/c\eta)}{B(c, \mu/c\eta)}$$

  - Given that beta function for large k follows a power law with exponent its second argument, we have that the degree distribution of vertices of a given fitness η has power law tail with exponent:

$$\alpha(\eta) = 1 + \frac{\mu}{c\eta}$$

- **The overall degree distribution depends on the fitness distribution ρ(η)**
  - If η is broadly distributed, the degree distribution will be a sum over power laws with a wide range of exponents, which will not in general yield another power law

Vertices of varying quality or attractiveness

- **What is the average degree?**
  - Short answer: 2c (there are c edges added for every vertex)
  - The computation reveals something interesting for the model that cannot be captured from $p_k(\eta)$

$$\langle k\rangle = \sum_{k=c}^{\infty}\int_0^{\infty} k p_k(\eta)\,\mathrm{d}\eta = \sum_{k=c}^{\infty}\int_0^{\infty} k\rho(\eta)\,\frac{\mathrm{B}(k,1+\mu/c\eta)}{\mathrm{B}(c,\mu/c\eta)}\,\mathrm{d}\eta$$

$$= \int_0^{\infty}\frac{\rho(\eta)}{\mathrm{B}(c,\mu/c\eta)}\sum_{k=c}^{\infty} k\,\mathrm{B}(k,1+\mu/c\eta)\,\mathrm{d}\eta.$$

$$\sum_{k=c}^{\infty} k\,\mathrm{B}(k,1+\mu/c\eta) = \frac{c}{1-c\eta/\mu}\,\mathrm{B}(c,\mu/c\eta).$$

  - The above result for the sum works only if $\eta<\mu/c$
    - ✓ If this is not satisfied the average degree of the network is diverging
      - This cannot be true since it is clear that the average degree is 2c

Vertices of varying quality or attractiveness

- **In order to avoid divergence we should have $\rho(\eta)=0$ for $\eta \geq \eta_0$, with $0 \leq \eta_0 < \mu/c$**
  - Assuming that this holds true we have:

$$\langle k \rangle = c\int_0^{\eta_0} \frac{\rho(\eta)d\eta}{1-{}^{c\eta}/_{\mu}} \xrightarrow{\langle k \rangle = 2c} \int_0^{\eta_0} \frac{\rho(\eta)d\eta}{1-{}^{c\eta}/_{\mu}} = 2$$

- **The integral is a monotonically decreasing function of μ**
  - When μ goes to infinity the integral takes its lowest value of 1
  - When $\mu \rightarrow c\eta_0$ the integral takes its largest value
    - ✓ Depending on the choice of ρ(η) this largest value can be less than 2
      - How can this be possible???!

Vertices of varying quality or attractiveness

- **The answer to the above question is subtle and interesting at the same time**

- **A degree probability distribution cannot capture all the details for the vertices**
  - If we have a fixed number of nodes whose degree scale in proportion (i.e., keeps increasing) to the network size, then:
    - ✓ These nodes do not contribute anything to the degree distribution at the limit of large n (i.e., even if we remove these nodes for the computation of the degree distribution the result will be the same for all practical purposes)
    - ✓ However, they make non-zero contribution to the average degree of a network

Vertices of varying quality or attractiveness

- **Bianconi and Barabasi referred to the existence of such vertices as *condensate***
  - For some choices of ρ(η) this condensate indeed appears
    - ✓ "Superhubs" with high degree
- **In this condensate the average degree can be written as:**

$$\langle k \rangle = \frac{K}{n} + \sum_{k=c}^{\infty} k p_k(\eta) d\eta = \frac{K}{n} + c \int_0^{\eta_0} \frac{\rho(\eta) d\eta}{1 - {}^{c\eta}\!/_{\mu}}$$

  - Where K is the sum of the degrees of the nodes that are in the condensate
  - Now, no matter what the value of the integral is, we can make the average degree equal to 2c by making K sufficiently large

Vertices of varying quality or attractiveness

- **The integral takes its largest value for $\mu \rightarrow c\eta_0$, which means that K should scale with the size n of the network:**

$$K > nc\left[2 - \int_0^{\eta_0} \frac{\rho(\eta)d\eta}{1 - \eta/\eta_0}\right]$$

- **In conclusion, *depending on the choice of the fitness distribution the network can exhibit two states:***
  - **Case 1:** The distributions of the degrees of vertices with any given fitness follows power law with a fitness-dependent exponent, but no vertices exhibit special behavior (i.e., no overall power law)
  - **Case 2:** A condensate of one or more “superhubs” exists. The rest vertices still follow a power law for each value of fitness

Vertices of varying quality or attractiveness

- **Intuitively, the above results can be interpreted as follows**
  - Nodes with large fitness, can enter the system in late stages of the network evolution and still obtain high degrees
    - ✓ Multiscaling: the time dependence of a node’s connectivity depends on the fitness model

- **If the fitness distribution is not bounded the asymptotic results still hold true for arbitrarily large time periods**
  - There is still a fittest vertex in the network and the latter cannot know whether the fitness value is bounded
  - The main difference is that μ changes with time
  - Furthermore, as time goes by the changes in the fittest value are more rare

Vertex copying models

- **By defining:** $$a = c\left(\frac{1}{\gamma} - 1\right) \Rightarrow \gamma = \frac{c}{c+a}$$
  - The previous equations gets:

$$\left[\gamma q + (1-\gamma)c\right] p_q(n) = \frac{c(q+a)}{c+a} p_q(n)$$

  - This is exactly the same equation as for Price model!

- **Following exactly same analysis as for the Price model we can write the master equation and solve for the degree distribution**
  - The result is that the in-degree distribution follows a power law tail with exponent: $$\alpha = 2 + \frac{a}{c} = 1 + \frac{1}{\gamma}$$

Vertex copying models

- **Since $\gamma \geq 0$, the exponent takes values from 2 to infinity**
  - Faithful copies (γ close to one) lead to exponents close to two
- **Other calculations from the Price model carry over as well**
  - However, the two models while giving the same degree distribution are not the same!
  - The networks created from these two processes can differ in various structural properties
    - ✓ For instance, the outgoing edges of a node in the vertex copying model will be similar to those of at least one other vertex
      - Similar correlations do not exist in Price model

Vertex copying models

- **The importance of vertex copying models is also related with the fact that not all power law networks originate from preferential attachment processes!**
  - We need to be very careful identifying the reasons that lead to the structure of real networks

- **While preferential attachment can be an intuitive explanation for many networks, it appears to be implausible for others**
  - E.g., biological networks

Network optimization models

- **The models that we have examined up to now consider the network evolution through a growth process**
  - New nodes are added, these are adding new edges, edges can be deleted etc.
  - Decentralized process, unaware of the large scale topology created

- **Some type of networks evolve due to the need for *optimization***
  - In many cases (e.g., transport, distribution etc.), networks need to achieve specific goals and this drives their design and structure
    - ✓ *The topology cannot be explained through a growth process*

Travel time and cost trade-offs

- **Case study: airline network**
- **In an airline network there are two conflicting factors**
  - Airline companies – operating on very small profit margins – want to minimize every possible operation cost
    - ✓ Less connection flights correspond to less cost
  - Airline customers want to reach to their destination fast (i.e., less connections)
    - ✓ This would require more connection flights (fully connected network)
- **While the costs and benefits are way more complicated simple models can help obtain intuitions as in social networks**

Ferrer i Cancho & Sole model

- **In this model the cost of operating the network is represented by the number of edges m present**

- **The customer satisfaction is captured from the average distance l between two vertices (i.e., average number of legs required to travel between two points)**
  - In reality l is a measure of dissatisfaction

- **The goal is to design a network that minimizes both m and l at the same time**
  - This is not possible in general
    - ✓ The minimum value of l is achieved by having a full mesh/clique, which maximizes m

Ferrer i Cancho & Sole model

- **Ferrer i Cancho and Sole introduced the quality function:**

$$E(m,\ell) = \lambda m + (1-\lambda)\ell, \quad 0 \le \lambda \le 1$$

- **We can calculate E for a given network**
  - Assuming we have a given number of vertices n, what is the topology that minimizes E?
  - λ controls the trade-off between the operational cost and customer satisfaction
    - ✓ If λ=0 → purely maximize customer satisfaction
    - ✓ If λ=1 → purely minimize network operational cost

Ferrer i Cancho & Sole model

- **We restrict our study in networks with one single connected component**

- **The smallest value for m=n-1**
  - Tree networks with n vertices

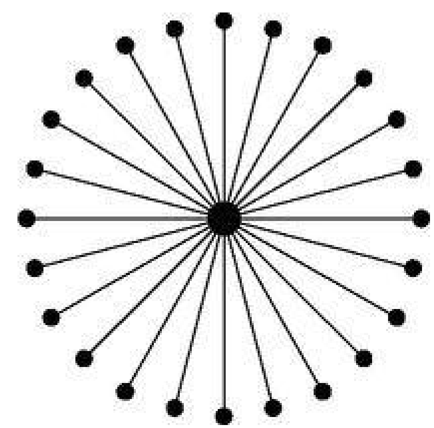
- **Given that λ is reasonably large (i.e., we place some importance on minimizing the operational cost), the optimal network is found by setting m=n-1**
  - This means that we have to search among all possible trees with n vertices that minimize the average path between two nodes

Ferrer i Cancho & Sole model

- **The tree that minimizes the average path length is a <u>star graph</u> (why?)**

- **It can be shown that for:** $\lambda \geq \frac{2}{n^2+2}$ **the hub and spoke topology is the optimal**
  - From the above conditions it appears that even for moderate size networks the star graph is the optimal solution

- **This model is trivial and the solution can be analytically found**
  - Global optimal solution



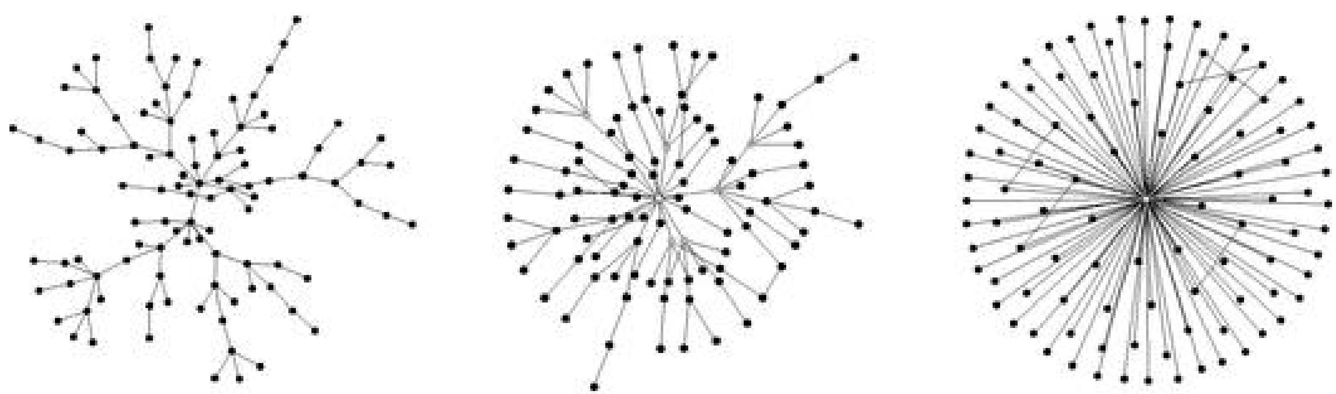
Ferrer i Cancho & Sole model

- **However, many times the global optimal solution is not easy to be obtained**
  - Utility function can be complicated
- **In these cases we can look for *local minima* using a *random hill climber* algorithm**
  - We start from a random network
  - We randomly pick a pair of vertices
    - ✓ If they are connected, we calculate the value of E(m,l) if we remove this edge
    - ✓ If they are not connected, we calculate the value of E(m,l) if we add an edge between them
      - In both cases if E(m,l) reduces we keep the change
  - We continue this process until the value of E(m,l) stops improving

Ferrer i Cancho & Sole model

- **Clearly optimization processes like the above can be trapped to local minima**
  - The fact that an addition or deletion of a single edge is not improving the utility function does not mean that we cannot globally improve the network

- **Using similar greedy algorithms the solutions we get in our case depend on the value of λ**
  - Larger values of λ appear to be harder to deal with



Gastner & Newman model

- **Gastner and Newman consider not only the number of legs of a journey but also the total distance traveled**
  - There is a delay associated with every plane change/journey leg (e.g., taxiing, security, disembarking, embarking etc.)
  - And there is a delay associated with the actual flight time between two points
  - Assuming that the first delay is constant for every leg, while the latter depends on the actual spatial distance between i and j we have:

$$t_{ij} = \mu + \nu r_{ij}$$

  - By varying the values of μ and v we can place more importance to the number of legs traversed or the distance traveled

Gastner & Newman model

- **Now the quality function becomes:**

$$E(m,t) = \lambda m + (1-\lambda)t, \quad 0 \le \lambda \le 1$$

  - Where t is the average shortest path distance between pair of vertices when distances are measured in terms of travel time
- **While the two models appear to be very similar there is a crucial difference**
  - Gastner and Newman's model require vertices to be placed on specific positions on the map (spatially embedded)