Graphs and Networks Lecture 4

Power Laws and Preferential Attachment

Daniel A. Spielman September 10, 2013

4.1 Disclaimer

These notes are not necessarily an accurate representation of what happened in class. They are a combination of what I intended to say with what I think I said. They have not been carefully edited.

You should be able to find a diary of my Matlab session from today's class. It may reveal computations that do not appear in these notes.

4.2 Overview

I begin by saying a little about power law degree distributions.

We will examine models of graphs that exhibit degree distributions that satisfy power laws. These are inspired by the models in [BA99, KRR+00]. We will see:

1. Simulations results.
2. The method of heuristic analysis from [BA99].
3. A heuristic treatment, from [Mit03], of the analysis of [KRR+00]. This may be made rigorous using the Martingale technique of Wormald [Wor95].

4.3 A History of Power Laws

In the late 90's, a number of papers were written that claimed that the distributions of the degrees of vertices in real-world graphs followed power laws. That is, they asserted that for most graphs there are constants a and c so that the fraction of vertices of degree d equals

$$ad^{-c}.$$

The constant c was usually between 2 and 3. This was an attempt to quantify the observation that real graphs have an unusually large number of vertices of high degree. Some papers didn't assert this for all degrees k, instead just claiming it for all k greater than some initial k_0.

If the degree distribution of a graph does satisfy a power law, then it should show up when we plot degrees vs. fraction of nodes of those degrees on a log-log scale. The log of the fraction of nodes of degree d would be

$$a - c \log d.$$

So, this should be a straight line in a log-log plot.

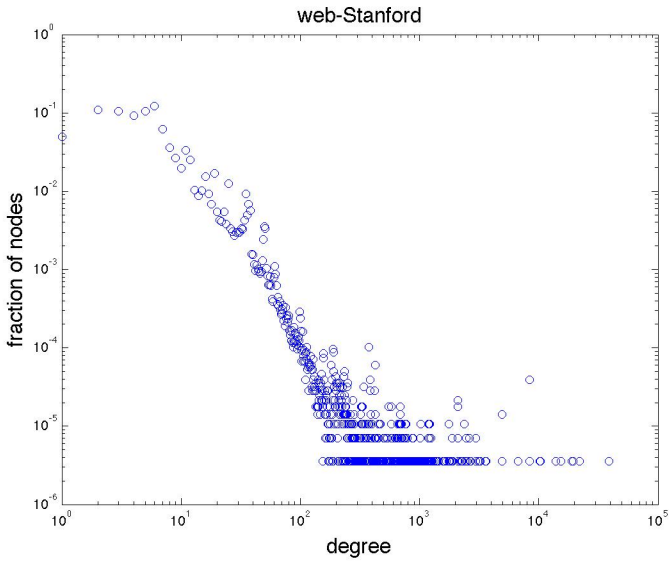
But, this doesn't work too well for the high-degree nodes, as there is likely only one of each. Newman's book describes better ways of visualizing these, including exponential binning. The one that I first try is examining the Cumulative Distribution Function. This is the fraction of nodes of degree *bigger than or equal to* d. We would expect this to be

$$\sum_{x \geq d} a x^{-c} \approx \int_{x \geq d} a x^{-c} = a x^{-c+1}/(c-1).$$

This should look linear on a log-log plot as well, but with a different slope.

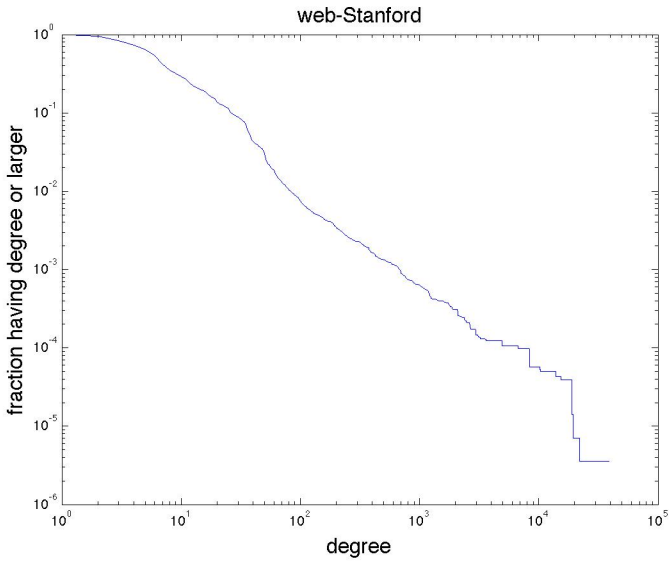
Let's try this for some of the graphs in our database. I'll generate the factions of nodes of each degree and plot it using the following code.

```
>> dat = load('web-Stanford');
>> a = dat.a;
>> a = a - diag(diag(a));
>> a = double(a+a'>0);
>> n = length(a);
>> degs = sum(a);
>> count = zeros(1,max(degs));
>> for i = 1:n, d = degs(i);
if (d > 0), count(d) = count(d)
end
>> frac = count/n;
>> loglog(frac,'o')
>> xlabel('degree');
>> ylabel('fraction of nodes')
>> gn(gn=='_') = ' ';
>> title(gn)
```

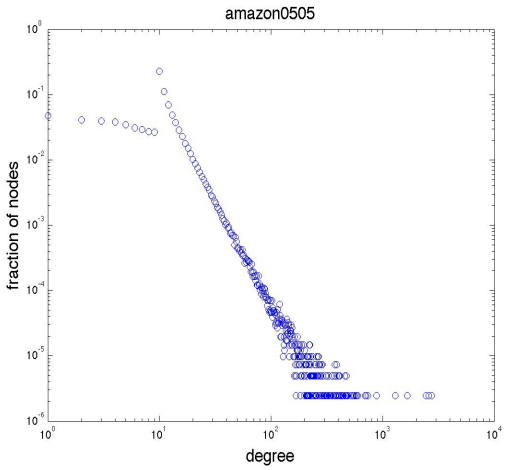


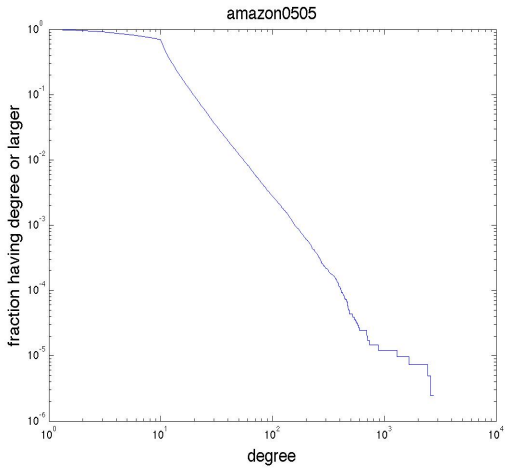
Here is the cumulative distribution function

```
>> cdf(1) = sum(frac);
>> cdf(2:length(frac)+1) = 1-cum
>> loglog(cdf)
>> xlabel('degree');
>> ylabel('fraction having degre
>> title(gn)
```



Here are similar plots for some of our other graphs.

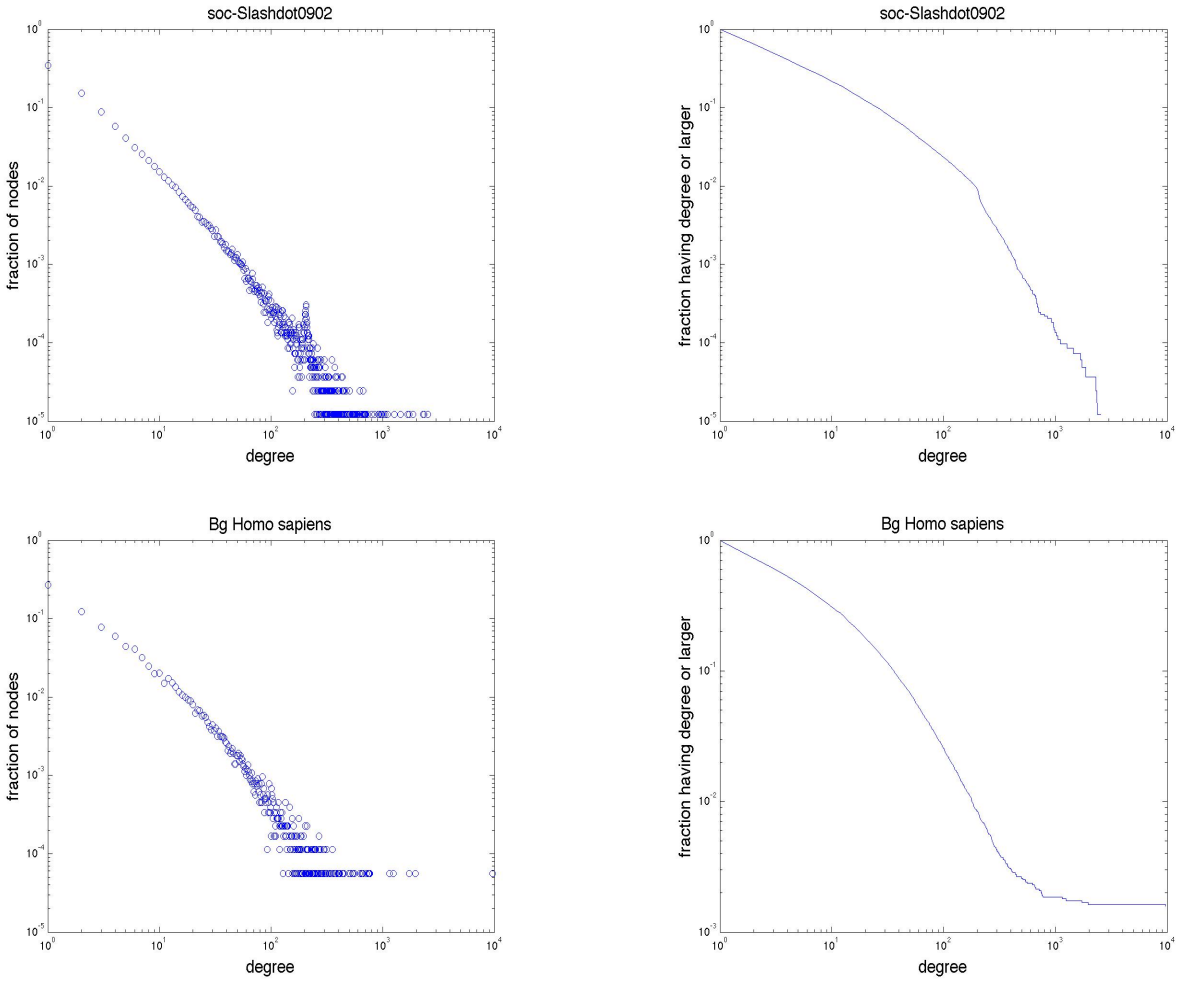




For some reason that I don't quite understand, this generated a lot of excitement. Part of it might be due to some people mistakenly believing that this implied something about how the graphs were generated, or that there was one process underlying all of these graphs. This led to them being called "scale-free" graphs. I do not like this term, as I can see scales in all of these graphs. For a treatment of the history of this business, I recommend reading [Mit03]. For those who would like to know more, I also recommend [FK05, Mit05, New05, LADW05].

A few important things to know about power laws are:

- Power laws had been observed in other (and related) fields before.
- Most of the early work used statistics that were not adequate to estimate c, or to even test whether the observed distribution was a power law.



- Some of the observations of power laws could be explained by the methods used to collect partial graphs.
- Whether or not the degrees in real graphs actually satisfy a power law is not important: it doesn't seem to predict anything. But, it is important to realize that real graphs have degrees much higher than we would expect from Erdös-Rényi graphs.

4.4 Preferential Attachment Models

We will now see a model of random graphs that does produce a power-law distribution of degrees.

The preferential attachment models start with a small graph, say with just one vertex and maybe a self-loop. As each time step, a new vertex is added to the graph. The endpoints of edges from this new vertex are biased towards other vertices of higher degree. The model proposed in [BA99] required each new vertex to have a fixed degree, and required that the endpoints of its edges be distributed proportionally to the degrees of existing vertices.

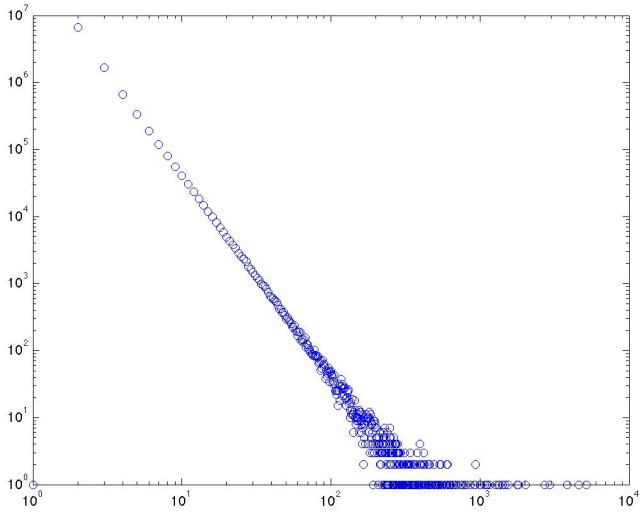
Kumar *et. al.* [KRR+00] were interested in the web, so they used directed graphs. They imagined that each new vertex would choose its links uniformly with some probability and as copies of the links of some other node otherwise. For simplicity, we will consider this model in the case that each new vertex has out-degree 1.

I can now state exactly the model we will examine. We begin with one node, which contains a directed edge to itself. This is the only self-loop we will include in the construction. We then choose some probability p. We then add vertices one-by-one, creating one edge leaving each. When we add vertex $t+1$, we do one of two things. With probability p we choose the endpoint of the edge uniformly from the t existing vertices. With probability $1-p$, we choose a random edge already in the graph, and use its endpoint.

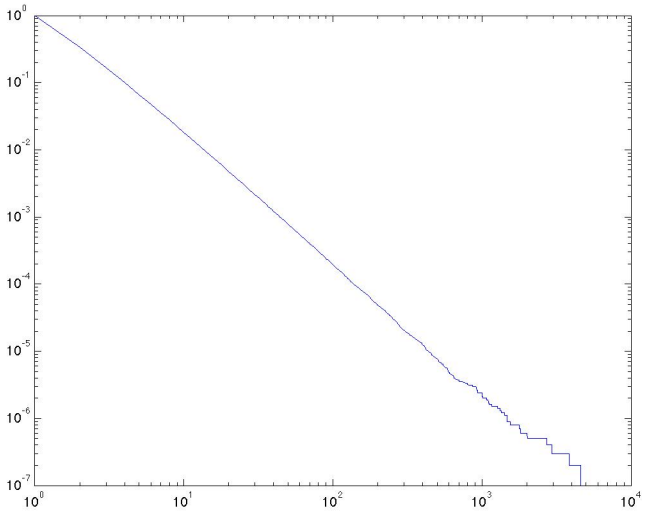
4.5 Simulation

I'll show a simulation of this for $p = .5$ and $n = 10^7$. We will see both plots of the number of vertices of each degree and the number of vertices exceeding each degree (the cumulative degree distribution). In a log-log scale these are pretty straight, suggesting that the distribution may satisfy a power law. You can find the Matlab code for `prefAttach` under the "Resources" directory.

```
>> p = .5;
>> n = 10^7;
>> [degs,dist] = prefAttach(p,n)
>> >> loglog(dist,'o')
% the observed degrees
```



```
% the cumulative distribution
>> cd = sum(dist)-cumsum(dist);
>> loglog(cd/n);
```



4.6 The distribution of low in-degrees

I will now present an analysis of the fraction of nodes of low in-degrees. While I will take a somewhat informal approach, this analysis can be made rigorous. I will say a little bit about how. Or, you can consult the papers on which this analysis is based: [KRR+00] and [Mit03].

First, let's figure out the expected number of nodes of in-degree 0. A node only goes from in-degree 0 to in-degree 1 when it is chosen as a uniform random neighbor of some node that comes along later. If node j has in-degree 0 when node $t+1$ is added, then the chance that node $t+1$ links to node j is p/t. So, the chance that node j has in-degree 0 after n nodes have been added is

$$\prod_{t=j+1}^{n} \left(1 - \frac{p}{t-1}\right)$$

For j bigger than $\sqrt{n}$ this is well-approximated by

$$\prod_{t=j+1}^{n} \exp\left(-\frac{p}{t-1}\right) = \exp\left(-p\sum_{t=j}^{n-1}\frac{1}{t}\right).$$

To compute this to first order, we recall a theorem of Euler's which says that

$$\sum_{i=1}^{k}\frac{1}{i} \to \ln(k) + \gamma,$$

where γ is some absolute constant. So,

$$\sum_{t=j}^{n-1}\frac{1}{t} \approx \ln(n-1) - \ln(j-1) \approx \ln(n/j).$$

This tells us that the probability that node j has in-degree 0 is approximately

$$\left(\frac{j}{n}\right)^p.$$

So, the number of nodes having in-degree zero should be approximately

$$\sum_{j=1}^{n}\left(\frac{j}{n}\right)^p = n^{-p}\sum_{j=1}^{n}(j)^p \approx n^{-p}\frac{1}{p+1}n^{1+p} = n\frac{1}{p+1}.$$

This agrees very well with our experimental data.

```
>> [round(n/(p+1)), dist(1)]

ans =

     6666667     6666960
```

Indeed, one can show that this quantity is well-concentrated. I know of two ways to do it. The easiest is to show that the events that different nodes have in-degree 0 are anti-correlated. That is, if one node has in-degree 0 then it is only less likely that another does. In this case, variables are known to have better concentration then predicted by the Chernoff bounds (see [DR98] or [DP09])[1] The other way uses Martingales, and can be extended to handle the case of other constant in-degrees.

Here is another way of thinking about this analysis. Let $X_0(t)$ be the number of vertices of in-degree 0 after t vertices have been added. When we add node $t+1$, it will have in-degree 0. Thus, the number of in-degree 0 nodes will increase unless that node points to another of in-degree 0, which happens with probability $pX_0(t)/t$. So,

$$X_0(t+1) = \begin{cases} X_0(t) & \text{with probability } pX_0(t)/t, \text{ and} \\ X_0(t)+1 & \text{with probability } 1-pX_0(t)/t. \end{cases}$$

That is,

$$\mathbf{E}\left[X_0(t+1)\right] = \mathbf{E}\left[X_0(t)\right] + 1 - p\,\mathbf{E}\left[X_0(t)\right]/t.$$

We now define Y_0 to be the expectation of X_0 by setting

$$Y_0(t+1) = Y_0(t) + 1 - pX_0(t)/t.$$

We hope that Y_0 approaches $c_0 t$ for some constant c_0, as t becomes large. In this case, c_0 should satisfy

$$c_0 = 1 - pc_0 t/t = 1 - pc_0.$$

This would give

$$c_0 = \frac{1}{1+p},$$

[1]The events that different nodes are isolated from Problem 1 are positively correlated. One approach to solving Problem 3 is to exploit negatively correlation. But, all these problems can be solved with mere union bounds.

the bound we obtained before.

For higher in-degrees, let $X_k(t)$ be the number of node of in-degree k after t have been added. These variables can either increase or decrease. X_k decreases if the edge from node $t+1$ hits a node of in-degree k, which happens with probability

$$\frac{pX_k(t)}{t} + \frac{(1-p)kX_k(t)}{t}.$$

Similarly, X_k increases if the edge hits a node of in-degree $k-1$, which happens with probability

$$\frac{pX_{k-1}(t)}{t} + \frac{(1-p)(k-1)X_{k-1}(t)}{t}.$$

So, the expectation of $X_k(t+1) - X_k$ is

$$\frac{1}{t}\left(pX_{k-1} + (1-p)(k-1)X_{k-1} - pX_k - (1-p)kX_k\right).$$

Setting $Y_k(t)$ to be our guess for the expectation of $X_k(t)$, we get

$$Y_k(t+1) = Y_k(t) + \frac{1}{t}\left(pX_{k-1} + (1-p)(k-1)X_{k-1} - pX_k - (1-p)kX_k\right).$$

If we look for a solution to these equations of the form

$$Y_k = c_k t,$$

we get

$$c_k = pc_{k-1} + (1-p)(k-1)c_{k-1} - pc_k - (1-p)kc_k.$$

Re-arranging, we get

$$\frac{c_k}{c_{k-1}} = \frac{p + (1-p)(k-1)}{1 + p + (1-p)k} = 1 - \frac{2-p}{(k+1) - p(k-1)}.$$

We should now check that these estimates agree extremely well with our experimental data.

```
>> c(1) = 1 /(1+p);
>> for k = 1:1000,
>>   c(k+1) = c(k)*(1 - (2-p)/((k+1) - p*(k-1)));
>> end
>> [(0:10)', dist(1:11)'/n, c']
%       degree  actual frac   prediction
            0      0.66659      0.66667
            1      0.16686      0.16667
            2     0.066601     0.066667
            3     0.033363     0.033333
            4     0.019016     0.019048
            5     0.011901     0.011905
            6    0.0079173    0.0079365
            7    0.0055513    0.0055556
            8    0.0040334    0.0040404
            9     0.003022    0.0030303
           10     0.002304     0.002331
```

The estimates are actually a pretty good fit for larger in-degres too.

```
>> ind = [20:10:100, 200:100:500];
>> [(ind)', dist(ind+1)'/n, c(ind+1)']

          20    0.0003769   0.00037644
          30    0.0001223   0.00012219
          40      5.4e-05    5.402e-05
          50     3.04e-05   2.8458e-05
          60     1.92e-05   1.6788e-05
          70      9.2e-06   1.0719e-05
          80      6.7e-06   7.2558e-06
          90        5e-06   5.1375e-06
         100      3.4e-06   3.7697e-06
         200        2e-07   4.8531e-07
         300        1e-07   1.4523e-07
         400        3e-07   6.1572e-08
         500            0   3.1619e-08
```

We can also see that they give a power-law distribution. As k grows large, these give

$$\frac{c_k}{c_{k-1}} \approx 1 - \frac{2-p}{k(1-p)},$$

and so

$$\frac{c_k}{c_j} \approx \prod_{i=j+1}^{k} 1 - \frac{2-p}{1-p}\frac{1}{i} \approx \left(\frac{j}{k}\right)^{\frac{2-p}{1-p}} = \left(\frac{j}{k}\right)^{1+\frac{1}{1-p}}.$$

This is equivalent to saying that for large k,

$$c_k \approx \beta \left(\frac{1}{k}\right)^{1+\frac{1}{1-p}},$$

for some constant β.

4.7 Making That Rigorous

Kumar *et. al.* [KRR$^+$00] make this argument rigorous by using Martingale arguments. Unlike Chernoff bounds, these do not require the variables under consideration to be independent. Rather, they consider sums of variables that are sampled in some order and require that the addition of each new variable does not change the expectation of the sum. These arguments imply that the fraction of nodes of each degree is tightly concentrated around the means that we have established. But, it is not clear that these arguments can be made rigorous for larger degrees.

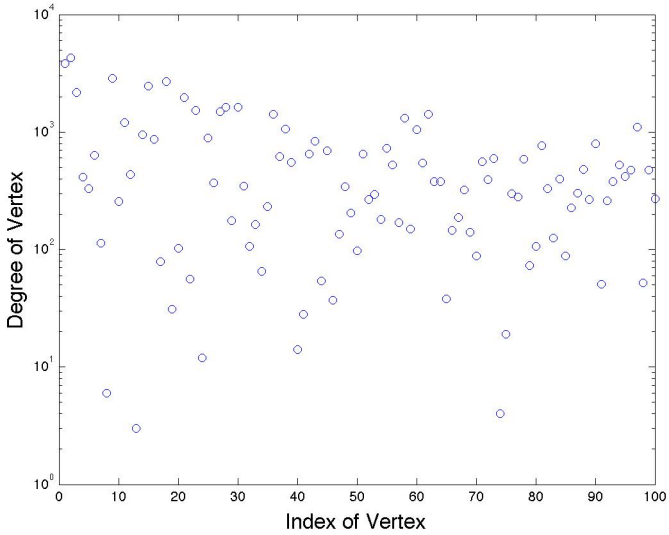
A more systematic approach to making these arguments rigorous begins by modeling them by a system of differential equations, and then applying arguments of Wormald [Wor95].

4.8 High-Degree Nodes

There is another heuristic argument that is common in the literature. The idea, which I think has its origins in the paper of Barabasi, Albert and Jeong[BAJ99], is to study the expected degree of the jth vertex added, for each j. It then derives the expected number of vertices of each degree from the expected degree of each vertex. The approach looks similar to that of the previous section. However, I do not know any way to make this argument rigorous.

I think the main obstacle to making this sort of argument rigorous is that the degrees of the individual nodes *are not concentrated*. To see this, I'll plot the degrees of the first 100 nodes.

```
>> semilogy(degs(1:100),'o')
```



However, it is very easy to miss this if one only examines the end-result of the arguments, which are predictions for the fraction of nodes of each in-degree.

4.9 Conclusion

This model does have both a good story and its does produce power-law distributions. But, do the graphs it produces resemble the graphs we see in the real world? There are many ways in which they differ, and many ways in which people have corrected the models to fix these differences. The difference that I find most profound is that real-world graphs are very far from being expanders. We now know that most real-world graphs have reasonably large sets of vertices (of sizes 100 to 1000) that are poorly connected to the rest of the graph. Very few models produce this.

References

[BA99] Albert-Laszlo Barabasi and Reka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, October 1999.

[BAJ99] Albert-László Barabási, Réka Albert, and Hawoong Jeong. Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications*, 272(1):173–187, 1999.

[DP09] Devdatt Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, New York, NY, USA, 2009.

[DR98] Devdatt Dubhashi and Desh Ranjan. Balls and bins: A study in negative dependence. *Random Structures and Algorithms*, 13(2):99–124, 1998.

[FK05] Evelyn Fox Keller. Revisiting scale-free networks. *BioEssays*, 27(10):1060–1068, 2005.

[KRR+00] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the web graph. *Foundations of Computer Science, Annual IEEE Symposium on*, 0:57, 2000.

[LADW05] Lun Li, David Alderson, John C Doyle, and Walter Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, 2(4):431–523, 2005.

[Mit03] Michael Mitzenmacher. A brief history of generative models for power law and lognormal distributions. *Internet Mathematics*, 1(2):226–251, 2003.

[Mit05] Michael Mitzenmacher. Editorial: The future of power law research. *Internet Mathematics*, 2(4):525–534, 2005.

[New05] Mark EJ Newman. Power laws, pareto distributions and zipf's law. *Contemporary physics*, 46(5):323–351, 2005.

[Wor95] Nicholas C. Wormald. Differential equations for random processes and random graphs. *The Annals of Applied Probability*, 5(4):pp. 1217–1235, 1995.