A Generating Function Approach to Analyze Random Graphs

Presented by - Vilas Veeraraghavan
Advisor - Dr. Steven Weber
Department of Electrical and Computer Engineering
Drexel University

April 8, 2005

Presentation to the
Center for Telecommunications and Information Networking group
List of References

This talk is a compilation of results as presented in [5]

References

Abstract

Social Networks like the Internet show interesting behavior that can be modeled with the help of Random Graphs. Random Graphs are named so because the properties of these graphs, namely the number of vertices, edges, and the connection between them are determined in some random manner. Though Erdos-Renyi Random Graphs were used traditionally to analyze network metrics, real-world characteristics were displayed better by networks that obeyed a power-law degree distribution. It has been proposed\cite{5} that the analyses of these metrics can be performed by using simple polynomials (Generating Functions) whose coefficients are the respective probability distributions. We investigate the validity and the effectiveness of this method with simulations. We aim to use this method to simplify the complex calculations involved in the analysis of large social, biological and most importantly, information flow networks.
Outline of talk

• Introduction to Random Graphs
• How do Random Graphs fit in?
• Introduction to Probability Generating Functions.
• Calculation and Analysis of metrics
• Future work and addressing these problems.
• Conclusion
• Questions?
Introduction to Random Graphs

- Random Graphs are graphs in which the properties of the graphs, namely the number of vertices, edges, and the connection between them are determined in some random manner.

- Example: Erdos-Renyi random graphs are minimal models of graphs with $n$ nodes and the edges between any two nodes are chosen uniformly at random.

- In particular, in the version we are looking at, each edge will be present with a probability $p$.

- Terms associated with random graphs - Phase Transition, Giant Component, Preferential Attachment.
Introduction to Random Graphs

Lets see an example of how Random Graphs are constructed - Let us generate a random graph on a set of $n = 7$ vertices. We assume that each edge between any randomly chosen pair of nodes has a probability of $p$ to be selected.

![Figure 1: A Random Graph on 7 vertices](image)

Note: Even though there are 21 possible edges in this graph, only 6 are chosen because of the probability.
Phase Transition and Giant Component

- **Phase Transition**: In a random graph, the probability at which a graph with several separate but connected components switches to a graph with a single Giant Component with several disconnected components.

- This phase transition can be observed in two cases[3] -
  - Keeping $p$ constant, while allowing the graph to evolve over time
  - Changing the probability of connection between vertices over non-evolving graphs (nodes constant).

- Explanation of 1st case - Take a set of nodes and start connecting them one pair every timestep (with a probability $p$ of course!). At each timestep, at most one edge is added. After $n$ time steps, there are at most $n$ edges.

- There is going to be a time point when one can imagine two connected components of relatively small size joined by a single edge which makes a large component
Figure 2: first case

Figure 3: Adding an edge creates a bigger connected component
The other case - phase transition based on probability - depends on whether the probability of the connection between the nodes is high or low. At low $p$, the graph will be largely segmented and at very high $p$, the graph will be almost fully connected. There is a threshold value of $p$ at which there is a transition from segmented pieces to the formation of a single large entity.
Degree Distribution is Poisson!

- For any Random Graph, the degree distribution follows the binomial distribution. That is, \( N(p) \sim \text{binomial}(n - 1, p) \). Why?

- But we know that as \( n \to \infty \), the binomial distribution will tend to the Poisson Distribution. We see the demonstration of the same effect by comparing simulations with the actual distributions.

- There is predominant clustering around the mean which indicates the Poisson effect.

- Poisson Distribution given by - \( P(N(p) = k) = \frac{e^{-\lambda} \lambda^k}{k!} \)
Degree Distribution is Poisson

- The red histograms are the degree distributions of Random Graphs that were constructed using a simulation.

- The blue histograms are the degree distributions that a poisson distribution generates.
Outline of talk

• Introduction to Random Graphs
• How do Random Graphs fit in?
• Introduction to Probability Generating Functions.
• Calculation and Analysis of metrics
• Future work and addressing these problems.
• Conclusion
• Questions?
How do Random Graphs fit in?

- Random Graphs show presence of giant component beyond the phase transition
- They also demonstrate the Small-world property\([6]\)
- However, most noticeably there is an absence of clustering (table 1)
- Degree distributions don’t match - *Preferential Attachment*

There is a lot of statistical data that shows that the Erdos Renyi model with a Poisson-like distribution is not the answer to model real-world networks.
Clustering Coefficient

- Clustering coefficient - What is the probability that my neighbor’s neighbor is my neighbor?

- In real-world networks, clustering is high. eg. Internet - *Why*?

- However, in case of ER Random Networks, the probabilities of vertex pairs being connected by edges are by definition independent, *So?* - *Disparity!"
Clustering Coefficient

- For an ER graph, the clustering coefficient is approximately \( \frac{z}{n} \), where \( z \) = average degree of the graph and \( n \) is the total number of nodes.

- Hence as \( n \to \infty \), the clustering coefficient goes to zero.

- In case of the corrected Model, the clustering coefficient is -

\[
C = \frac{z}{n} \left[ c_v^2 + \frac{z-1}{z} \right]^2
\]

- The disparity between real-world models and Random Graphs is seen clearly in the table.
## Clustering Coefficient

<table>
<thead>
<tr>
<th>network</th>
<th>$n$</th>
<th>$z$</th>
<th>measured</th>
<th>random graph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet (autonomous systems)$^a$</td>
<td>6374</td>
<td>3.8</td>
<td>0.24</td>
<td>0.00060</td>
</tr>
<tr>
<td>World-Wide Web (sites)$^b$</td>
<td>153127</td>
<td>35.2</td>
<td>0.11</td>
<td>0.00023</td>
</tr>
<tr>
<td>power grid$^c$</td>
<td>4941</td>
<td>2.7</td>
<td>0.080</td>
<td>0.00054</td>
</tr>
<tr>
<td>biology collaborations$^d$</td>
<td>1520251</td>
<td>15.5</td>
<td>0.081</td>
<td>0.000010</td>
</tr>
<tr>
<td>mathematics collaborations$^e$</td>
<td>253339</td>
<td>3.9</td>
<td>0.15</td>
<td>0.000015</td>
</tr>
<tr>
<td>film actor collaborations$^f$</td>
<td>449913</td>
<td>113.4</td>
<td>0.20</td>
<td>0.00025</td>
</tr>
<tr>
<td>company directors$^f$</td>
<td>7673</td>
<td>14.4</td>
<td>0.59</td>
<td>0.0019</td>
</tr>
<tr>
<td>word co-occurrence$^g$</td>
<td>460902</td>
<td>70.1</td>
<td>0.44</td>
<td>0.00015</td>
</tr>
<tr>
<td>neural network$^c$</td>
<td>282</td>
<td>14.0</td>
<td>0.28</td>
<td>0.049</td>
</tr>
<tr>
<td>metabolic network$^h$</td>
<td>315</td>
<td>28.3</td>
<td>0.59</td>
<td>0.090</td>
</tr>
<tr>
<td>food web$^i$</td>
<td>134</td>
<td>8.7</td>
<td>0.22</td>
<td>0.065</td>
</tr>
</tbody>
</table>


Figure 7: Table 1 taken from [5]
Preferential Attachment

- New nodes added to the network are more likely to connect to the more popular nodes (nodes with higher degree).
- Older nodes have a greater opportunity to acquire links, by virtue of being older and newer nodes more likely to connect with high-degree nodes than low-degree nodes.
Preferential Attachment

For example: scientific literature citations

- older papers more likely to be cited than newer papers, more exposure
- newer papers more likely to cite older papers, more prestige
- Preferential Attachment gives the Random Graph a power-law distribution for its degree.
Make things better

- Generalize the Erdos-Renyi random graph to mimic clustering and degree distributions of the real-world networks.

- Restrict the degree distribution to a specific degree sequence

- Fraction of vertices having degree $k$ will tend to the desired degree distribution $p_k$ as $n$ becomes large.

- Generate 'stubs' projecting out of each of the nodes equal to the degree of the node.

- The stubs are then chosen randomly in pairs and connected to form edges.
Construction Example

The degree distribution considered here is - \( D = \{2, 5, 5, 4, 4, 4, 1, 1, 6\} \)
Outline of talk

• Introduction to Random Graphs
• How do Random Graphs fit in?
• Introduction to Probability Generating Functions.
• Calculation and Analysis of metrics
• Future work and addressing these problems.
• Conclusion
• Questions?
The probability generating function for a random variable is a polynomial whose coefficients are the probabilities.

Alternative representation for the probability distribution.

For the probability distribution $p_k$ the corresponding generating function is given by $G_0(x) = \sum_{k=0}^{\infty} p_k x^k$.

The function generates the distribution - by simple differentiation:

$$p_k = \frac{1}{k!} \left. \frac{d^k G_0}{dx_k} \right|_{x=0}$$
Example for Generating Functions

- Consider the case of a fair 6-sided die. The Generating function will be -
  \[ F_0(x) = 0x^0 + \frac{1}{6}(x^1 + x^2 + x^3 + x^4 + x^5 + x^6) \]

- It is obvious that we can "generate" the distribution out of the polynomial.

- Note - If a certain property is described by a gen. function, then its sum over \(m\) independent realizations is generated by the \(m^{th}\) power of the gen. function.

- For the earlier example shown, we will have PGF as -
  \[ G_0(x) = \frac{2}{9}x + \frac{1}{9}x^2 + \frac{1}{3}x^4 + \frac{2}{9}x^5 + \frac{1}{9}x^6 \]
Still on Generating Functions

Properties of interest -

- Properly Normalized if $G_0(1) = \sum_k p_k = 1$
- Mean and other moments can be calculated by simple differentiation
- Mean $= G'_0(1) = \sum_k kp_k = \langle k \rangle$

For the graph we generated,

- $p_1 = \frac{1}{1!} \frac{d}{dx} \left( \frac{2}{9}x + \frac{1}{9}x^2 + \frac{1}{3}x^4 + \frac{2}{9}x^5 + \frac{1}{9}x^6 \right)|_{x=0} = \frac{2}{9}$
- $p_2 = \frac{1}{2!} \frac{d^2}{dx^2} \left( \frac{2}{9}x + \frac{1}{9}x^2 + \frac{1}{3}x^4 + \frac{2}{9}x^5 + \frac{1}{9}x^6 \right)|_{x=0} = \frac{1}{9}$
- and so on...
Properties

- Normalized - $G_0(1) = \sum_k p_k = \frac{2}{9} + \frac{1}{9} + \frac{1}{3} + \frac{2}{9} + \frac{1}{9} = 1 - \text{Yes!}$
- Mean - $G'_0(1) = \sum_k k p_k = \langle k \rangle = \frac{2}{9} + \frac{2}{9} + \frac{4}{3} + \frac{10}{9} + \frac{6}{9} = 3.55$
Outline of talk

- Introduction to Random Graphs
- How do Random Graphs fit in?
- Introduction to Probability Generating Functions.
- Calculation and Analysis of metrics
- Future work and addressing these problems.
- Conclusion
- Questions?
The Main Question

- Question - What is the Probability Distribution for the cluster size containing a randomly chosen node?
- Solution - If components are tree-like then we can follow edges starting from the node till we reach a dead end.
- What to we need? - Degree distribution of a randomly chosen node, and the degree distribution of the randomly chosen neighbor of a randomly chosen node.
**Metrics**

- Degree distribution of the vertices is governed by $p_k$.

- Degree distribution of the first neighbor - *tricky!*

- Let's say that the degree distribution of the neighbor is given by $q_k$.

- Randomly choose an edge from the graph and travel to one of the endpoints.
Since higher degree vertices have more edges connected to them, there is a greater chance of reaching a higher degree vertex.

Thus the degree distribution of the vertex that is reached is proportional to $kp_k$ and not just $p_k$.

However, we are not interested in the degree distribution of the vertex, we are interested in counting the number of edges attached to the vertex other than the one we arrived on. Why?
Derivation

Figure 14: The formula [1]

- Basically the number we are looking for is one less than the degree of the vertex. Hence we can say -

\[ q_k = \frac{(k+1)p_{k+1}}{\sum_j jp_j} \]
Derivation

- Generating function for this degree distribution can be derived from degree distribution of the vertices - \( G_0(x) \).

- \( G_1(x) = \frac{G_0'(x)}{z} \) where \( z \) is the average degree of the graph. \( z = \sum_k kp_k \)

- The main metrics we want to analyze are - distribution of component sizes and the mean component size of the graph.

- We also want to get an understanding of the phase transition
Component Size Distribution

Figure 15: different possibilities [5]

- Assumption 1 - we are below the phase transition - *no giant component*
- Assumption 2 - all finite components are tree-like - no closed loops.
- Finding Clusters - We choose a node at random and follow the node to a point where there are no more nodes to be traversed. The resulting set will form a cluster.
- Let $H_1(x)$ be the generating function that generates the distribution of the sizes of these clusters.
Consider the following experiment: choose an edge at random, choose a direction on that edge, i.e., choose one of the two vertices, and count the number of vertices in the component (tree) at that vertex.

- Define the random variable $M$ as the number of vertices so counted.

- Explanation - we know that $H_1(x)$ generates the distribution for the component size that each node belongs to. Take the randomly chosen node and travel once along each one of its edges.
Derivation

- Clearly \( M = 1 + M_1 + \cdots + M_L \), where \( L \sim q \).
- Define \( H_1(x) \) as the pgf for \( M \).
- This is so because, each \( M_i \) has a cluster size given by \( H_1(x) \).
- Hence the generating function takes the form of
  \[
  H_1(x) = x \sum_{k=0}^{\infty} q_k [H_1(x)]^k = xG_1(H_1(x))
  \]
Similarly...

Figure 18: For a randomly chosen node

- Now we can get a Generating function for the distribution of the component size of a randomly chosen node that has degree governed by $p_k$.

- $H_0(x) = x \sum_{k=0}^{\infty} p_k [H_1(x)]^k = xG_0(H_1(x))$ Why?
Mean Component Size

- Assumption - below phase transition.

- Since the component size distribution cannot be calculated in closed form most of the times, a useful statistic is the mean component size.

- Hence mean component size $\langle s \rangle = H_0'(1) = 1 + G_0'(1)H_1'(1)$

- Using the fact that properly normalized generating functions have a value of 1 at $x = 1$, we consider $G_0(1) = H_1(1) = 1$ giving us on simplification -

$$H_1'(1) = \frac{1}{1 - G_1'(1)}$$

- This will give us $\langle s \rangle = \frac{G_0'(1)}{1 - G_1'(1)}$
Mean Component Size

• But we know $G'_0(1) = \langle k \rangle = \sum_k k p_k = z_1$

• and also $G'_1(1) = \frac{\sum_k k(k-1)p_k}{\sum_k k p_k} = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} = \frac{z_2}{z_1}$

• From this, we can simplify and obtain

$$\langle s \rangle = 1 + \frac{z_1^2}{z_1 - z_2}$$

• Hence the phase transition happens at $z_1 = z_2$. This is because the equation diverges.
Summary of Results from [5]

\[ G_0(x) = \sum_k p_k x^k \]

\[ G_1(x) = \sum_k q_k x^k \]

\[ H_1(x) = x G_1(H_1(x)) \]

\[ H_0(x) = x G_0(H_1(x)) \]
Simulation Results

Package used for graphing methods and algorithms - JUNG."[4]

Figure 23: Distinct difference in the degree distribution for the ER graph and the Scale-free Graph. Case 1 - \( n = 100, p = 0.05 \) for ER graph and \( n = 100 \) for scale-free
Simulation Results

Figure 24: Distinct difference in the degree distribution for the ER graph and the Scale-free Graph. Case 1 - $n = 1000$, $p = 0.005$ for ER graph and $n = 1000$ for scale-free
Simulation Results

- The values of the distributions of $G_0(x), G_1(x), H_0(x), H_1(x)$ are found using algebraic manipulations that follow the analysis.

- For the value of $n = 2000$, $p = 0.005$, the values obtained by using a model that "corrects" the ER random Graph model, we get -

  - $G_0(x) = 0.5915x^1 + 0.2145x^2 + 0.084x^3 + 0.031x^4 + 0.03x^5 + 0.0145x^6 + 0.0105x^7 + 0.0045x^8 + 0.0065x^9 + 0.0030x^{10} + 0.0015x^{11} + 0.0020x^{12} + 0.0010x^{13} + 0.0020x^{14} + 0.0015x^{15} + 0.0010x^{16} + (5.0E-4)x^{18}$

  - $G_1(x) = 0.30170877477684265x^0 + 0.2188217291507269x^1 + 0.1285386381025249x^2 + 0.06324917112981383x^3 + 0.07651109410864576x^4 + 0.044376434583014546x^5 + 0.03749043611323643x^6 + 0.018362662586074982x^7 + 0.029839326702371848x^8 + 0.015302218821729153x^9 + 0.008416220351951035x^{10} + \cdots$
\begin{align*}
0.012241775057383322x^{11} &+ 0.006630961489415967x^{12} + \\
0.014282070900280544x^{13} &+ 0.0114766664116296864x^{14} + \\
0.00816118337158882x^{15} &+ 0.004590665646518746x^{17} \\
\bullet \quad H_0(x) &= 0.17846072430502427x^2 + 0.019525542138958282x^3 + \\
&0.00230697555567951458x^4 + 2.568699511614761E - 4x^5 + \\
&7.499991416541853E - 5x^6 + 1.0936929589700816E - 5x^7 + \\
&2.3894866879221294E - 6x^8 + 3.0896958703813303E - 7x^9 + \\
&1.3464941707215986E - 7x^{10} + 1.8749957082733816E - 8x^{11} + \\
&2.8285130360716792E - 9x^{12} + 1.1378495015464754E - 9x^{13} + \\
&1.7164957413025768E - 10x^{14} + 1.0357635613164747E - 10x^{15} + \\
&2.3437419530171958E - 11x^{16} + 4.714182998247629E - 12x^{17} + \\
&2.1456172210499953E - 13x^{19} + 0 \\
\bullet \quad H_1(x) &= 0.3017087477684265x^1 + 0.0660204298865876x^2 + \\
&0.011700636805385354x^3 + 0.0017370749021908777x^4 + 6.33980677645127E - \\
&4x^5 + 1.1094095948978005E - 4x^6 + 2.8277948969492658E - 5x^7 + 
\end{align*}
\[ 4.178794076593516E - 6x^8 + 2.048765433050615E - 6x^9 + \\
3.169899760394559E - 7x^{10} + 5.2601256799304266E - 8x^{11} + \\
2.308401355631057E - 8x^{12} + 3.772518112752916E - 9x^{13} + \\
2.451511387731301E - 9x^{14} + 5.943555248564316E - 10x^{15} + \\
1.2751805236172792E - 10x^{16} + 6.529350799475894E - 12x^{18} + 0 \]
Probability Distributions

G0(X)

![Graph G0(X)](graph1)

H0(x)

![Graph H0(x)](graph2)

G1(X)

![Graph G1(X)](graph3)

H1(x)

![Graph H1(x)](graph4)
Outline of talk

- Introduction to Random Graphs
- How do Random Graphs fit in?
- Introduction to Probability Generating Functions.
- Calculation and Analysis of metrics
- Future work and addressing these problems.
- Conclusion
- Questions?
Future work

• Ananth already talked about the work on wireless sensor networks. Recap

• Our work needed some strong mathematical analysis to back it up.

• This method of generating functions gives us an avenue to explore tough metrics using a simple differentiation/substitution technique and involves polynomials (easy!)

• We look to apply the Probability Generating function approach to the analysis and integrate it with the ideas in percolation theory

• Simulations are being generated as a part of a larger ”complete” visualization unit.

• It is still a long way to go!!!
Outline of talk

• Introduction to Random Graphs
• How do Random Graphs fit in?
• Introduction to Probability Generating Functions.
• Calculation and Analysis of metrics
• Future work and addressing these problems.
• Conclusion
• Questions?
Conclusion

- Random Graphs are effective models of real-world networks
- Probability Generating functions are an effective tool to analyze graphs. We wait to see how effectively they perform in our case.
- It has been noted that all social networks, biological networks and information flow networks that share characteristics like preferential attachment, clustering, etc conform to the same model.
- The usage of Generating Functions will give us more flexibility in dealing with these networks.