On Reshaping of Clustering Coefficients in Degree-Based Topology Generators

Xiafeng Li, Derek Leonard, and Dmitri Loguinov

Texas A&M University, College Station, TX 77843, USA
{xiafeng, dleonard, dmitri}@cs.tamu.edu

Abstract. Recent work has shown that the Internet exhibits a power-law node degree distribution and *high* clustering. Considering that many existing degree-based Internet topology generators do not achieve this level of clustering, we propose a randomized algorithm that increases the clustering coefficients of graphs produced by these generators. Simulation results confirm that our algorithm makes the graphs produced by existing generators match clustering properties of the Internet topology.

1 Introduction

Many studies [5], [7], [10] examine the properties of the Internet connectivity graph and attempt to understand its evolution. Results in [10], [19] demonstrate that both AS-level and router-level Internet topology exhibits three important characteristics: *power-law degree distribution*, *small diameter*, and *high clustering*. To satisfy these properties, many degree-based generators have been proposed to model the Internet graph [5], [7], [10], [21], [33]. Although most of them can achieve the necessary degree distributions and diameter, they are often not successful in producing high levels of clustering. As shown in [10], the graphs produced by the existing Internet generators always exhibit much lower clustering than the Internet graph. In order to make these graphs better match the Internet, we propose a randomized algorithm that increases the clustering of synthetic graphs while preserving their power-law degree distributions and small diameters.

Considering that the algorithm spends most of its time on computing clustering coefficients during each iteration, we use sampling theory and let the method utilize approximate values instead of computing them exactly, which reduces the time complexity of the algorithm from $\Theta(mn)$ to $\Theta(m)$, where m is the number of edges and n is the number of nodes. This paper makes two contributions. First, it proposes a new and practical method to estimate the clustering of massive graphs by using sampling theory. Second, it offers a solution to the low-clustering problem of existing Internet topology generators.

The remainder of the paper is organized as follows. First, we review the important properties of the Internet topology and classify existing generators in section 2. Then, we abstract the low clustering problem in these generators and propose our algorithmic solution in section 3. Following that, the analysis of the

S. Leonardi (Ed.): WAW 2004, LNCS 3243, pp. 68–79, 2004.

algorithm and the corresponding simulation results are given in section 4. We conclude the paper in section 5.

2 Background

In this section, we first review properties of the Internet topology and then discuss existing degree-based Internet topology generators.

2.1 Internet Properties

It is important to study the properties of the Internet topology, because they not only affect the deployment of Internet services [18], [20], but also impact the performance of existing Internet protocols [22], [23], [30]. Past work has showed that the Internet AS-level graph exhibits the following three properties.

The first property of the Internet is related to its degree distribution. In 1999, Faloutsos *et al.* [19] observed that the CDF of node degree X (both AS-level and router-level) follows a power-law (Pareto) distribution:

$$P(X \leq d) = 1 - cd^{-\alpha} \ , \tag{1}$$

where $\alpha \approx 1.2$ is the shape parameter and c is the scale parameter.

The second property of the Internet topology is its high clustering. In 2002, Bu *et al.* [10] pointed out that the Internet is highly clustered. The paper showed that the *clustering coefficient* [32], which is a metric of the probability that two neighbors share a common third neighbor, of the Internet topology is much higher than that of random graphs.

The third property is that the Internet graph has a small diameter. Bu *et al.* [10] showed that the *average shortest path length* between each pair of nodes at the AS-level is very small and is close to that of random graphs.

To reproduce these properties, many generators have been proposed to model the Internet [2], [3], [4], [9], [10], [33]. As most of them seek to satisfy the first property (the power-law degree distribution), they are also called *degree-based* generators. Next, we review several widely-used generators and classify them into two types.

2.2 Existing Internet Topology Generators

Despite the various implementations, existing degree-based Internet generators can be categorized into two collections: *evolving* and *non-evolving* generators.

Non-evolving methods do not model the evolution of the Internet and produce graphs with a given fixed number of nodes n. *GED* [14], [26] and *PLRG* [5] are classical examples belonging to this collection. In *GED* (Given Expected Degree) [14], [26], [27], [28], n pre-assigned weights $(w_1, \ldots, w_n)$ are drawn from a Pareto distribution. Edge (i, j) exists with probability p_{ij}:

$$p_{ij} = \frac{w_i w_j}{\sum_{k=1}^{n} w_k} \ . \tag{2}$$

To make p_{ij} less than or equal to 1, the pre-assigned degree sequence is assumed to satisfy the following condition [14], [26]:

$$w_{max}^2 \leq \sum_{k=1}^{n} w_k \ , \tag{3}$$

where $w_{max} = \max_{i=1}^{n}\{w_i\}$.

To relax this assumption, the PLRG model [5] is proposed. In PLRG, a power-law weight sequence $\{w_i\}$ is first pre-assigned to n nodes. After that, w_i virtual copies of node i are produced and are randomly selected to form links with equal probability. Because of their simplicity, PLRG and GED are good theoretical models for complex networks and many analytical results [5], [14], [15], [26] are based on them. However, they exhibit much lower clustering than the real Internet.

Unlike non-evolving generators, evolving methods focus on modelling the evolution of the Internet topology. In 1999, Barabasi *et al.* [7] proposed the *BA* model, in which the graph evolves by adding new nodes that attach to existing nodes with a so-called *linear preferential* probability $\prod(d_i)$:

$$\prod(d_i) = \frac{d_i(t)}{\sum_j d_j(t)} \ , \tag{4}$$

where d_i is the degree of node i at time t. As shown in [7], the BA model is scale-free and generates graphs with a power-law degree distribution. However, the shape parameter of the power-law function does not match that of the Internet. Moreover, the clustering of a BA-generated graph is much smaller than that of the Internet, which limits the model's usefulness in simulating the Internet.

In order to make the BA model match the Internet more accurately, many BA-like models [1], [6], [10], [31], [33] have been proposed. One of the most successful methods is GLP, which extends the BA model by using a Generalized Linear Preference probability:

$$\prod(d_i) = \frac{d_i(t) - \beta}{\sum_j (d_j(t) - \beta)} \ , \tag{5}$$

where $\beta \in (-\infty, 1]$ is a tunable parameter. Simulation results in [10] show that GLP not only improves the power-law distribution of BA, but also has clustering as high as 0.35. However, compared with the value of 0.45 in the Internet, GLP also needs to increase its clustering [10].

3 Clustering Problem and Algorithmic Solution

Recall that the Internet structure exhibits high clustering, but most existing Internet generators fail to imitate this property. To confirm the clustering inconsistency between the Internet and its generators, we next compare the clustering evolution of the Internet topology with that of its generators.

3.1 Clustering Evolution

To obtain the clustering evolution of the Internet graph, we examined AS (autonomous system) BGP logs from the National Laboratory for Applied Network Research (NLANR) [29], which has been recording a snapshot of the Internet topology every day since 1997. Obvious errors (i.e., duplicate links between some nodes and disconnected components) in a few graphs indicate that part of the data is unreliable and must be removed before the analysis. The assumption on which we filter the data is that the Internet topology must be a connected, simple graph and the number of ASes should increase over time. Based on this assumption, we removed self-loops, merged duplicate links, and discarded graphs that were disconnected or had much fewer ASes than in previously recorded graphs. After preprocessing the data in this manner, we obtained 253 snapshots of the Internet that correspond to the AS-level Internet topology from November 1997 to January 2000.

The clustering coefficients of the obtained graphs are plotted in Figure 1 (left), where γ increases from 0.34 in 1997 to 0.42 in 2000 at a slow, but steady rate. At the same time, the number of nodes in the system also grows as shown on the right side of the figure.

To compare the clustering evolution of the Internet with that of its generators, we simulate the clustering evolutions of GED, PLRG, BA and GLP in

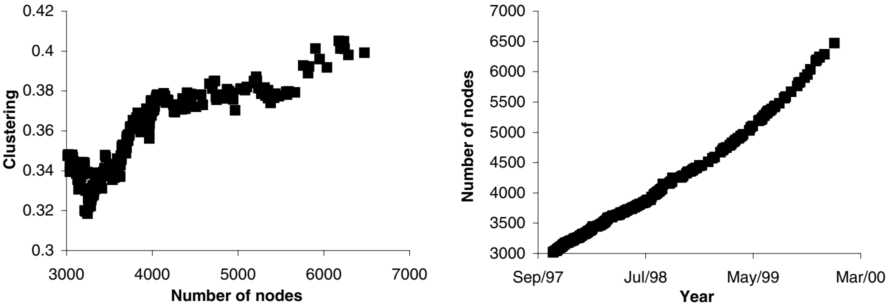

Fig. 1. Evolution of clustering (left) and the number of nodes (right) in the Internet

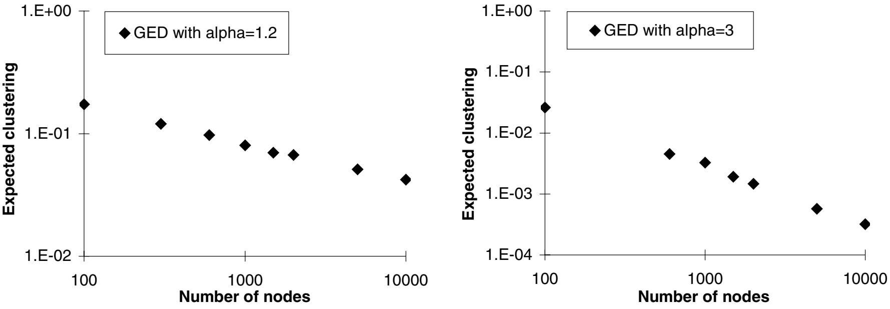

Fig. 2. Clustering evolution of GED with $\alpha = 1.2$ (left) and $\alpha = 3$ (right)

Figures 2, 3, 4 and 5. In the simulations, the number of nodes n increases while other parameters in these models are fixed. Comparing Figure 1(left) with Figures 2, 3, 4 and 5, we conclude that the graphs produced by those generators should be altered to exhibit clustering as high as that of the Internet. Consid-

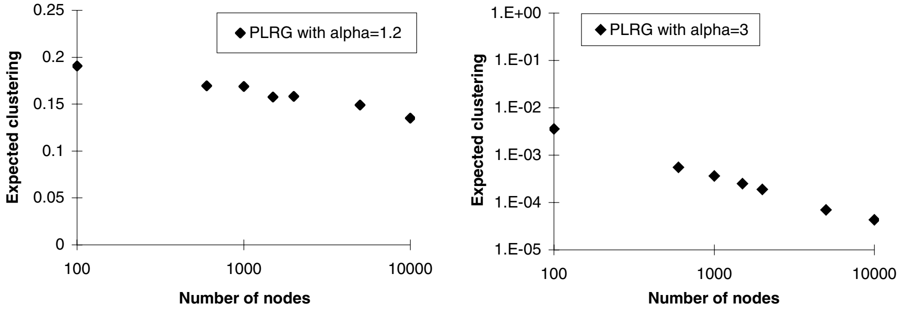

Fig. 3. Clustering evolution of PLRG with $\alpha = 1.2$ (left) and $\alpha = 3$ (right)

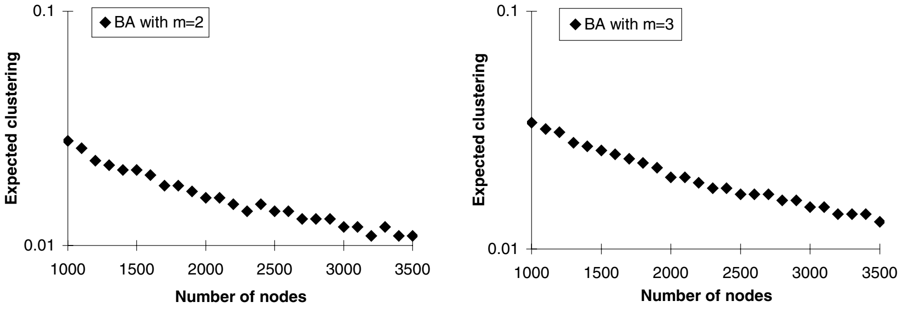

Fig. 4. Clustering evolution of BA with $m = 2$ (left) and $m = 3$ (right)

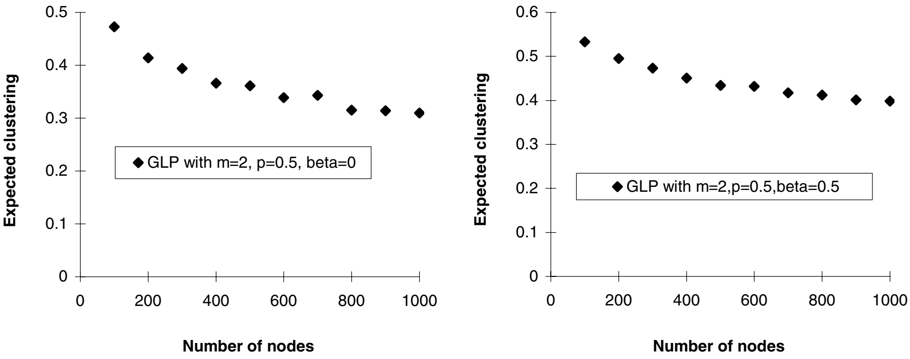

Fig. 5. Clustering evolution of GLP with $m = 2, p = 0.5, \beta = 0$ (left) and $m = 2, p = 0.5, \beta = 0.5$ (right)

ering that the power-law degree sequence and low diameter are also necessary properties in these graphs, we need to keep these properties unchanged while increasing clustering of the graph. The details of the problem can be described as follows.

3.2 Clustering Problem

Given a *connected* graph G and a target clustering value γ_T, rewire G's edges and produce a new graph G' satisfying the following four conditions:

1. G' is connected.
2. The degree sequence in G is the same as that in G'.
3. G' has low diameter.
4. Clustering of G' is larger than or equal to γ_T, which means $\gamma(G') \geq \gamma_T$.

3.3 Clustering and Triangles

To better understand this problem, we next review the concept of clustering and reveal how it relates to triangles.

The *clustering coefficient of a graph* $G(V, E)$, denoted by $\gamma(G)$, is the average clustering coefficient of each node with degree larger than 1:

$$\gamma(G) = \frac{\sum_{v \in V - V^{(1)}} \gamma_v}{|V| - |V^{(1)}|} , \tag{6}$$

where $V^{(1)}$ is the set of degree-1 nodes in G, γ_v is the *clustering coefficient of node* v and $|V|$ is the number of nodes in G. Here, γ_v characterizes the probability that the neighbors of node v are adjacent to each other. More precisely,

$$\gamma_v = \frac{N_v}{d_v(d_v - 1)/2} , \tag{7}$$

where N_v is the number of edges among the neighbors of node v and d_v is its degree. An example of computing clustering is shown in Figure 6, where graph G contains five nodes A, B, C, D, and E. According to the definition, the clustering coefficient of each node is:

$$\gamma_A = \frac{1}{2(2-1)/2} = 1 , \quad \gamma_B = \frac{1}{2(2-1)/2} = 1 ,$$

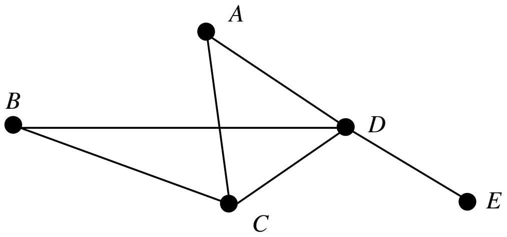

Fig. 6. Clustering in graph G

$$\gamma_C = \frac{2}{3(3-1)/2} = \frac{2}{3}\ , \quad \gamma_D = \frac{2}{4(4-1)/2} = \frac{1}{3}\ .$$

And the clustering coefficient of graph G is

$$\gamma(G) = \frac{\gamma_A + \gamma_B + \gamma_C + \gamma_D}{4} = \frac{3}{4}\ . \tag{8}$$

Note that N_v in (7) is essentially the number of triangles containing node v. For example, in Figure 6, $N_A = 1$ because it is in one triangle (DAC); $N_C = 2$ because it is contained by two triangles (DCB) and (DCA). Therefore, for any node with fixed degree d, the more triangles it includes, the higher clustering it has. Also note that the clustering of a graph is the average of each node's clustering. Intuitively, increasing the number of triangles is a promising way to increase the clustering of a graph.

3.4 Our Algorithm

The key idea of our algorithm is to increase the number of triangles for each node. According to condition 2 in the clustering problem, the degree of each node v should not be changed, which indicates that increasing N_v in (7) will increase the clustering of node v. Therefore, rewiring the links in G to produce more triangles for each node will increase the clustering of the whole graph.

To better describe our algorithm, we first give the definition of *unsatisfied* and *satisfied* nodes as follows.

Definition 1. *A node $v' \in G'$ is unsatisfied if $d_v > d'_v$, $v \in G$. Otherwise, v' is satisfied.*

For example, in Figure 6, if we remove edge (A, C) from the graph, nodes A and C are unsatisfied because their degree decreases. This simple definition facilitates the explanation of our algorithm, which can be separated into four steps. The first step finds all triangles in G and *marks* the corresponding links in these triangles. Then, it randomly picks a node w and searches for k-length ($k \geq 4$) loops starting from node w. At each time when such a loop is found, our algorithm randomly breaks an *unmarked* link (u, v) from that loop and marks nodes u, v *unsatisfied.* In the third step, the algorithm adds links between any pair of *unsatisfied* nodes so that at least one new triangle is generated. This step is repeated until the clustering of current graph is larger than γ_T or there are no *unsatisfied* nodes remaining. Finally, if the current clustering $\gamma_c(G)$ is larger than γ_T, the algorithm randomly adds links between *unsatisfied* nodes and outputs G'. Otherwise, the method loops back to step two.

In step four, the time complexity of computing current clustering $\gamma_c(G)$ is $\Theta(nm)$, while step 1 to step 3 will only cost $\Theta(m)$. Obviously, reducing the time complexity of computing $\gamma_c(G)$ will improve the performance of our algorithm. Therefore, in step four we randomly sample s nodes and approximate the clustering of the whole graph by the average clustering of the sampled nodes. By applying this randomized sampling technique, the time complexity of step four

```
*Input:* a connected, power-law graph $G$ and target clustering $\gamma_T$.
*Output:* a connected, power-law graph $G'$, such that $\gamma(G') \geq \gamma_T$.

Copy graph $G$ to graph $G'$.
Use *BFS* to find all triangles in $G'$ and mark all corresponding edges in the triangles.
Randomly sample $s$ nodes and compute $\gamma_s(G')$, which is the average clustering coefficient of the $s$ nodes.
*While* $\gamma_s(G') < \gamma_T$ *do*
    Randomly pick a node $w$ in $G'$.
    Start from $w$ and apply *BFS* to find all $k$-cycles ($k \geq 4$) in the graph.
    *If* there are no such cycles, output *Fail.*
    *Else For* each $k$-cycle $l$, randomly break its unmarked edge $(u, v)$.
    *While* there exist at least two unsatisfied nodes *do*
      *If* there exist unsatisfied nodes $s$ and $t$ such that edge $(s, t) \notin G'$
      *AND* connecting $s$ and $t$ creates at least one triangle *do*
        Connect $s$ and $t$
      *Else if* there exist unsatisfied nodes $u, v$, and $w$ such that there are no edges among them and $d_u - d'_u \geq 2$, $d_v - d'_v \geq 2$, and $d_w - d'_w \geq 2$;
      *AND* connecting nodes $u, v$ and $w$ creates one new triangle
        connects links $(u, v), (u, w)$ and $(v, w)$.
      *Else*
        break the while loop;
      *Endif*
   *EndWhile*;
   Randomly sample $s$ nodes and compute $\gamma_s(G')$, the average clustering of the $s$ nodes.
*EndWhile*;
Randomly connect unsatisfied nodes and output $G'$.
```

Fig. 7. Algorithm to increase clustering coefficients of random graphs

is reduced to $\Theta(sm) = \Theta(m)$. A detailed description of the algorithm is shown in Figure 7.

4 Analysis of the Algorithm

There are two issues that must be explored in order to show that our algorithm is effective. We first study through simulations the effect of running our algorithm on random graphs created by several of the methods mentioned before. Since an approximation of clustering in the graph is used in the algorithm, we then analytically determine how accurate these approximations can become.

To show that our algorithm indeed increases the clustering in a wide range of random graphs, we ran sample graphs created by BA, GED, and PLRG through the algorithm. The results are displayed in Fig. 8 and Fig. 9. In each case the graph contains 1000 nodes. For the graph generated by BA, $m = 2$. In the case of both PLRG and GED, $\alpha = 1.5$. Note that duplicate links and self-loops were removed from these graphs before we ran the algorithm. As shown in the

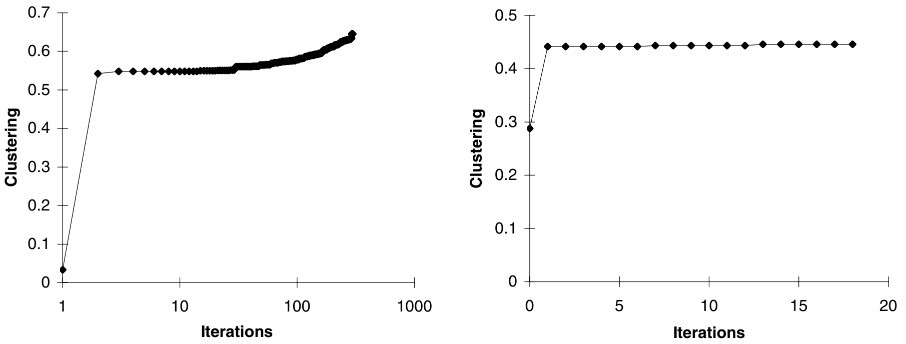

Fig. 8. Increase in clustering for BA graph of 1000 nodes with $m = 2$ (left) and for a GED graph of 1000 nodes with $\alpha = 1.5$ (right)

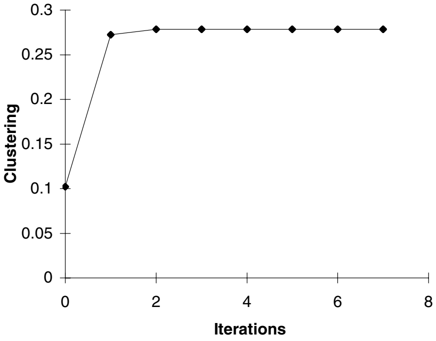

Fig. 9. Increase in clustering for a PLRG generated graph of 1000 nodes with $\alpha = 1.5$

figures, there is a marked increase in clustering for each example graph in few iterations.

We next determine the validity and accuracy of using an approximate value for the clustering of a graph instead of requiring that the exact value be know. There is obviously some error between the approximate and actual values, but according to sampling theory, increasing the sample size s will reduce this error. However, when the s exceeds a certain threshold, further increasing it does not significantly decrease the error. To determine a proper sample size, we provide the following lemma.

Lemma 1. *When the sample size $s = Z^2_{\frac{\rho}{2}}/(2E)^2$, the error between the approximate and actual clustering does not exceed E with probability at least $1 - \rho$.*

Proof. Denote by σ^2 the variance of node clustering in the graph, γ_s the sampled clustering of s nodes in the graph, and γ_a the actual clustering of the graph. According to sampling theory [24], when the sample size is:

$$s = \frac{Z^2_{\frac{\rho}{2}} \sigma^2}{E^2} \ , \tag{9}$$

error $|\gamma_s - \gamma_a|$ does not exceed *error margin* E with probability $1 - \rho$, where $\rho \in (0, 1)$ is a *significance level* and $z_{\rho/2}$ is the positive z value that is at the vertical boundary for the area of $\rho/2$ in the right tail of the standard normal distribution.

Note that the clustering of each node is between 0 and 1. When half of the total nodes have clustering 0 and the other half have clustering 1, the variance of node clustering reaches its maximum point, where the clustering of the graph is 0.5. Therefore:

$$\sigma^2 \leq \frac{\sum_{i=1}^{n} 0.5^2}{n} = 0.5^2 \ . \tag{10}$$

Using (9) and (10), we conclude that when sample size:

$$s = \frac{Z^2_{\frac{\rho}{2}}}{(2E)^2} \ , \tag{11}$$

$|\gamma_s - \gamma_a|$ does not exceed E with probability at least $1 - \rho$.

Thus by Lemma 1, we can determine the sample size s based on an error margin E and significance level ρ. For the error margin $E = 0.1$ and $\rho = 0.05$, we only need to sample $s = 1.96^2/0.2^2 \approx 97$ nodes to contain the absolute error within 0.1 of the correct value with probability at least 0.95. This result shows that we are indeed able to approximate the clustering of a graph with very few samples, which justifies its inclusion in our algorithm.

5 Conclusion

In this paper, we offer an algorithmic solution to the clustering problem and show that we can frequently improve this metric in graphs produced by existing degree-based generators to values well over 0.5. This in turn allows those generators to better simulate the AS-level Internet topology.

References

1. R. Albert and A. Barabasi, “Topology of Evolving Network: Local Events and Universality,” *Physica Review Letters* 85, 2000.
2. R. Albert, H. Jeong, and A. Barabasi, “Diameter of the World Wide Web”, *Nature* 401, 1999.
3. R. Albert, H. Jeong, and A. Barabasi, “Error and Attack Tolerance in Complex Networks,” *Nature* 406, July 2000.
4. A. Barabasi, H. Jeong, R. Ravasz, Z. Neda, T. Vicsek, and A. Schubert, ”On the Topology of the Scientific Collaboration Networks,” *Physica A 311*, 2002.
5. W. Aiello, F. R. K. Chung, and L. Lu, “A Random Graph Model for Massive Graphs,” *ACM STOC*, 2000.

6. W. Aiello, F. R. K. Chung, and L. Lu, “Random Evolution in Massive Graphs,” *IEEE FOCS*, 2001.
7. A. Barabasi, R.Albert, and H.Jeong, “Mean-field Theory for Scale-free Random Networks,” *Physica A* 272, 1999.
8. A. Barabasi and R. Albert, “Emergence of Scaling in Random Networks,” *Science*, October 1999.
9. A. Barabasi, R. Albert, and Hawoong Jeong, “Scale-free Characteristics of Random Networks: The Topology of the World Wide Web,” *Physica A 281*, 69-77 (2000).
10. T. Bu and D. Towsley, “On Distinguishing between Internet Power Law Topology Generators,” *IEEE INFOCOM*, June 2002.
11. K. Calvert, M. Doar, and E. Zegura, “Modeling Internet Topology,” *IEEE Communications Magazine*, June 1997.
12. H. Chang, R.Govindan, S. Jamin, S. Shenker, and W. Willinger, “Towards Capturing Representative AS-Level Internet Topologies,” *University of Michigan Technical Report CSE-TR-454-02*, 2002.
13. Q. Chen, H. Chang, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, “The Origin of Power-laws in Internet Topologies Revisited,” *IEEE INFOCOM*, June 2002.
14. F. R. K. Chung, “Connected Components in Random Graphs with Given Expected Degree Sequences,” *Annals of Combinatorics* 6, 2002.
15. F. R. K. Chung, “The Spectra of Random Graphs with Given Expected Degree,” *http://math.ucsd.edu/ fan/wp/specp.pdf.*
16. M. Doar, “A Better Model for Generating Test networks,” *IEEE GLOBECOM*, November 1996.
17. P. Erdos and A. Renyi, “On Random Graphs,” *I, Publication Math.* Debrecen 6, 290-291, 1959.
18. T. S. E. Ng and H. Zhang, “Predicting Internet Network Distance with Coordinates-Based Approaches,” *IEEE INFOCOM*, 2002.
19. M. Faloutsos, P. Faloutsos, and C. Faloutsos, “On power-Law Relationships of the Internet Topology,” *ACM SIGCOMM*, August 1999.
20. P. Francis *et al.*, “IDMaps: A Global Internet Host Distance Estimation Service,” *IEEE/ACM Transactions on Networking*, vol. 9, no. 5, October 2001.
21. C. Jin, Q. Chen and S. Jamin, “Inet: Internet Topology Generator,” *University of Michigan Technical Report CSE-RT-433-00*, 2000.
22. G. Huston, “Architectural Requirements for Inter-Domain routing in the Internet,” *IETF Draft draft-iab-bgparch-01.txt.*
23. C. Labovitz, A. Ahuja, R. Wattenhofer, and S. Venkatachary, “The Impact of Internet Policy and Topology on Delayed Routing Convergence,” *IEEE INFOCOM*, 2001.
24. Y. Leon, “Probability and Statistics with applications,” *International Textbook Company*, 1969.
25. http://mathworld.wolfram.com/HypergeometricFunction.html
26. M. Mihail and C.H. Papadimitriou, “On the Eigenvalue Power Law,” *RANDOM*, 2002.
27. M. Mihail and N. Visnoi, “On Generating Graphs with Prescribed Degree Sequences for Complex Network Modeling applications,” *ARACNE*, 2002.
28. M. Molloy and B. Reed, “A Critical Point for Random Graphs with a Given Degree Sequence,” *Random Structures and Algorithms*, 6:161-180, 1995.
29. National Laboratory for Applied Network Research “Global ISP Interconnectivity by AS Number,” *http://moat.nlanr.net/as/.*

30. P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, and D. Estrin, "On Characterizing Network Topologies and Analyzing Their Impact on Protocol Design," *USC Technical Report 00-731*, February 2000.
31. E. Ravasz and A. Barabasi, "Hierarchical Organization in Complex Networks," *Physical Review* E(in press).
32. D. J. Watts, "Small World," *Princeton University Press*, 1999.
33. S. Yook, H. Jeong, and A. Barabasi, "Modeling the Internet's Large-scale Topology," *Proceedings of the Nat'l Academy of Sciences* 99, 2002.