CS 559: Algorithmic Aspects of Computer Networks		Spring 2010
	Lecture 22 — April 08	
Lecturer: John Byers	*BOSTON UNIVERSITY*	*Scribe: Rick Skowyra*

In today's lecture, we continued our discussion of power law random graphs. Previous classes have discussed Erdos-Renyi random graphs and Barabasi-Albert preferential attachment graphs [2]. The first half of the class covered work conducted by Chung and Lu [1]. In this work the authors presented the Generalized Random Graph (GRG) and Power Law Random Graph (PLRG) generative models.

22.1 Generalized Random Graphs (GRG)

In order to build a target graph G of n nodes, we create a degree sequence $D = (d_1, d_2, ..., d_n)$. d_1 specifies the degree of node 1, d_2 the degree of node 2, etc. The resultant graph has, *in expectation,* the desired node degrees.

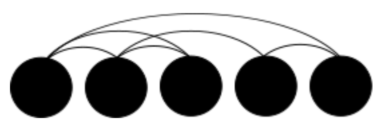

Figure 22.1. GRG Model

To build a graph using GRG:

1. Write down all vertices in the graph.
2. For each vertex v_i, draw an edge to v_j with probability $Pr[(v_i, v_j) \in E] \sim d_i d_j$

The above equation does not include a normalization factor that is present in the actual paper, but has little bearing on the model conceptually. There are a few key points to note about GRG:

- A vertex with a high degree has a high probability of many edges.
- High-degree nodes tend to pair with other high-degree nodes, due to the relatively large size of $d_i d_j$ in the above equation.
- GRG does not sample uniformly at random from all possible graphs. Some graphs are more likely to be generated than others (e.g. those with high-degree nodes attached to other high-degree nodes are more likely than those with high-degree nodes attached to only low-degree nodes).

22.2 Power Law Random Graphs (PLRG)

PLRG, like GRG, requires a (power-law) degree sequence for the target graph. It is not probabilistic, however. The result is a 'graph' with node degrees exactly as specified, with two caveats explained below.

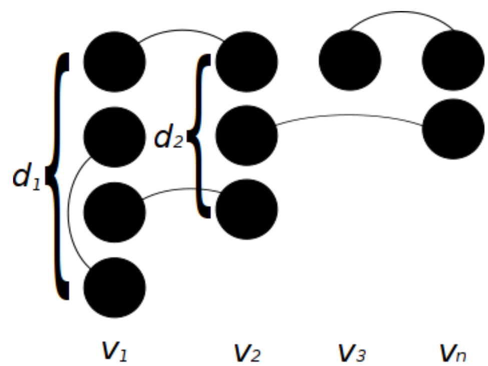

Figure 22.2. PLRG Model

To construct a PLRG:

1. Make d_i copies of each node v_i.
2. Randomly generate a matching on the copied vertices (i.e. pair every vertex with another vertex).
3. Collapse the copies of each node column-wise to create the final graph.

Caveats:

- Note that in the expanded graph above, v_1 is connected to itself. This creates a self-referential cycle in the final graph, or reduces node degree by one if it is simply ignored.
- Also note that v_1 is connected to v_2 twice. Either the final graph is a multigraph allowing multiple edges between two nodes, or both nodes are one degree lower than intended if the extra edge is ignored.

Regardless of how these issues are dealt with, it has been shown that PLRNG is asymptotically equivalent to GRG. Therefore it also samples non-uniformly from the space of all possible graphs, and will favor graphs with nodes of high degree connected to other nodes of high degree.

22.3 Li et al.

The above generative models, as well as the models that were covered in previous classes, have a common problem: they do not conform to real-world data. Despite most graph-theoretic metrics between generated graphs and real-world topologies being similar, their actual structure varied significantly.

Li et al. [3] studied this discrepancy in detail, using an ISP network. Their key observation is that the Internet is not composed of a few highly-connected 'center' 'nodes and a large number of low-degree nodes on the edges. In fact, low degree nodes tend to form the network center. High-degree nodes are closer to the edges. This is due to router technology and the relationship between throughput and bandwidth.

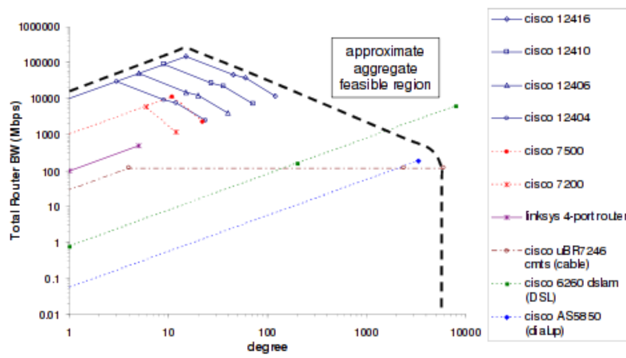

Figure 22.3. Router Technology Constraints [3]

As the degree of a router increases, the switching overhead gets larger and reduces the effective throughput of the router. This creates a configuration space with low-degree core routers serving as a backbone, and high-degree aggregation routers (e.g. DSLAMS) acting as access points for a large number of low-bandwidth links:

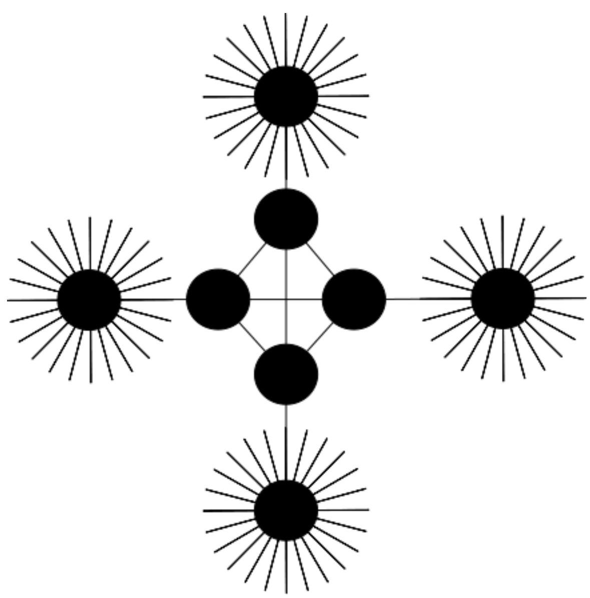

Figure 22.4. ISP-Like Graph

22.3.1 Throughput

Graph-theoretic metrics do not provide the sufficient distinguishing capability needed to select these kinds of graphs from the sort generated by the GRG, PLRG, or B-A models. Li et al. provide two new metrics: Throughput and Likelihood.

To calculate the Throughput of a graph:

1. Annotate each node with a throughput rating based on its degree by assuming a given router family in Figure 3.

2. Compute network traffic demands by assuming uniform traffic generation rates and shortest-path routes.

3. Measure how much throughput the network as a whole can deliver.

In PLRG-type graphs, the core is quickly saturated by end-to-end traffic (due to the large number of routes and low throughput) and performance suffers. These bottlenecks leave the network both fragile and unable to deliver high bandwidth to end nodes. In HOT Model graphs (the model developed by Li et al., which approximates ISP-like graphs), however, aggregation points get saturated at about the same time as core nodes. There are fewer bottlenecks, throughput is higher, and the network is more robust.

22.3.2 Likelihood

The authors note that HOT-type graphs are unlikely to be produced by previous generative models, and quantify that statement in the Likelihood metric:

$$L(g) = \sum_{e=(i,j)} d_i d_j \tag{22.1}$$

over all edges e. Note that this is a deterministic measure over an existing graph. For all graphs of size n and a given degree distribution, $\exists g_{min}$ that minimizes L(g) and $\exists g_{max}$ that maximizes L(g). This can be used to normalize:

$$= l(g) = \frac{L(g) - L(g_{min})}{L(g_{max}) - L(g_{min})} \in [0,1]' \tag{22.2}$$

- B-A, PLRG, and GRG graphs tend to approach a high likelihood ($l(g) \to 1$)
- HOT graphs tend to approach a low likelihood ($l(g) \to 0$)
- Real-world topologies have a low likelihood

The authors show the relationship between likelihood and performance in a final figure, which also serves to place the various graph generation models:

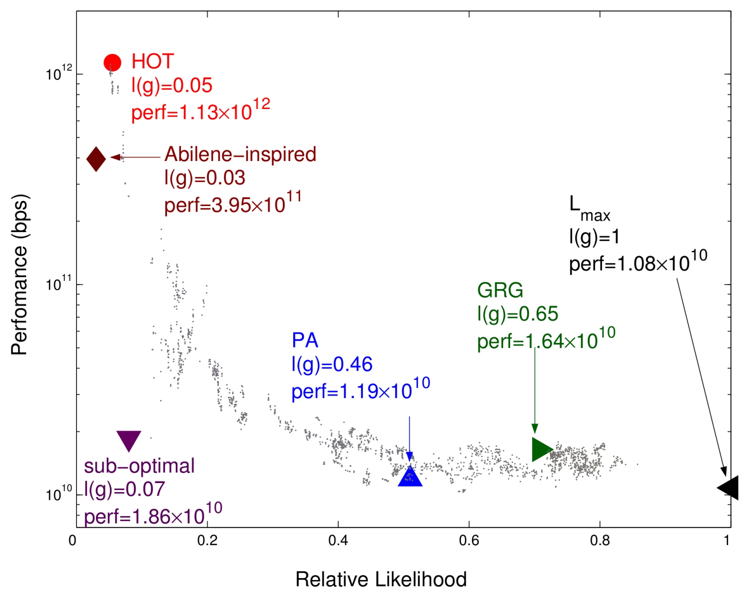

Figure 22.5. Likelihood [3]

Bibliography

[1] W. Aiello, F. Chung, and L. Lu. A random graph model for power law graphs. *Experimental Mathematics*, 10(1):53–66, 2001.

[2] A. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509, 1999.

[3] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the internet's router-level topology. *ACM SIGCOMM Computer Communication Review*, 34(4):3–14, 2004.