ORNL/TM-2015/153

Synthetic Graph Generation for Data-Intensive HPC Benchmarking: Scalability, Analysis and Real-World Application

Sarah Powers
Joshua Lothian

December 2014

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE US DEPARTMENT OF ENERGY

ORNL/TM-2015/153

Extreme Scale Systems Center, Computer Science and Mathematics Division

Synthetic Graph Generation for Data-Intensive HPC Benchmarking: Scalability, Analysis and Real-World Application

Sarah Powers, Joshua Lothian

Date Published: December 2014

Prepared by
OAK RIDGE NATIONAL LABORATORY
P.O. Box 2008
Oak Ridge, Tennessee 37831-6285
managed by
UT-Battelle, LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

LIST OF FIGURES

ABSTRACT

The benchmarking effort within the Extreme Scale Systems Center at Oak Ridge National Laboratory seeks to provide High Performance Computing benchmarks and test suites of interest to the DoD sponsor. The work described in this report is a part of the effort focusing on graph generation. A previously developed benchmark, SystemBurn, allows the emulation of a broad spectrum of application behavior profiles within a single framework. To complement this effort, similar capabilities are desired for graph-centric problems. This report described the in-depth analysis of the generated synthetic graphs' properties at a variety of scales using different generator implementations and examines their applicability to replicating real world datasets.

1. INTRODUCTION

Benchmarks provide an import tool for standardized testing, especially in High Performance Computing (HPC) where decisions regarding computer hardware appropriations carry large price tags and potentially big ramifications with respect to scientific computing capability. Many existing benchmarks, especially those running on hundreds of thousands or more cores, test the capability of machines to solve dense linear algebra systems.

There is, however, another class of problems experiencing rapid growth - data analysis tasks. Many of these "Big Data" problems can be conceptually represented by graphs, and it is often difficult to acquire and/or release production data sets of a meaningful size. Hence, there is a desire to generate synthetic data sets that resemble production data, and to perform this generation as quickly as possible. One possible use for these synthetic graphs could be as inputs to benchmarks. An example of this can be seen in the Graph500 benchmark [1].

This report is the second in the series "Synthetic Graph Generation for Data-Intensive HPC Benchmarking." For more details on the selected graph models, generators, package implementation and testing details, we refer the reader to the first report [30]. A summary of the tested generators and their limitations is shown in Table 1.

This report seeks to describe the in-depth analysis of synthetic graphs produced at different scales from each of the generators in Table 1. This analysis may be used as guidance in selecting a graph model when creating data sets that mimic applications that are of interest, in particular when real data may not be available. We examine various features/statistics of the synthetic graphs in our corpus. One may then match these features to real graphs for use in selecting a graph model.

For the purposes of this document, we will re-use the definition of a graph presented in [30]: a **graph** $G = (V, E)$ consists of a set of vertices V and a set of edges $E \subseteq V \times V$. Our current work is restricted to **undirected**, **simple** graphs, where the pairs in E are unordered, and repeated entries in E are forbidden. The semi-standard notation of $n = |V|$ will be used to denote the number of vertices (these terms may be used interchangeably).

Table 1. List of generators and limitations.

Model	Package	Limit	Limitation
Erdos-Renyi	APGL	2^{18} vertices	time
Erdos-Renyi (gnp method)	GGEN	$\sim 2^{18}$ vertices	memory
Erdos-Renyi (layer method)	GGEN	$\sim 2^{18}$ vertices	memory
Erdos-Renyi	NetworkX	depends on parameters	time
Erdos-Renyi (fast variant)	NetworkX	2^{22} vertices	time
Hyperbolic	Krioukov	2^{20} vertices	time
Inet	inet-3.0	$3037 < n \leq 2^{17}$	time, model limitation
Kronecker (R-Mat)	pywebgraph	2^{20} vertices	time
Kronecker (R-Mat+noise)	Graph500	2^{30} vertices	memory
Partial k-tree	INDDGO	2^{27} vertices	memory
PLRG	Boost	2^{30} vertices	memory
RDPG	MFR	<46000 vertices	memory
Small World (Watts-Strogatz)	APGL	2^{26} vertices	time
Small World (Watts-Strogatz)	NetworkX	2^{24} vertices	time
Preferential Attachment (B-A)	APGL	2^{16} vertices	memory and/or time
Preferential Attachment (B-A)	NetworkX	2^{23} vertices	time
Tiers	tiers	2^{26} vertices	memory
Waxman	stocksim	<180000 vertices	memory
Waxman	NetworkX	8000 vertices with default a=0.4, b=0.1	time

Tests were run on a 48-core HP DL585 G7 with 384GB of RAM.

2. ANALYSIS

New graph generators or modifications to existing models are frequently proposed under the rationale that the properties of real-world networks need to be replicated more closely. For example, Watts and Strogatz introduced the "Small World model" to address the need for higher clustering [42].
In the graph community in particular, generators are often analyzed on the basis of "features." Since there does not appear to be a consensus on a single comparison metric, we use a standard set of features encompassing a variety of graph aspects. These are summarized in Table 2 with full descriptions available in [30].

Table 2. Common features used in graph analysis

Single-valued	**Multi-valued**
Average degree	Betweenness centrality
Clustering coefficient	Degree distribution
Degeneracy	Delta hyperbolicity
Degree assortativity	Eigenvalues (spectrum)
Diameter	Expansion
Edge density	k-cores
Path length	Node diameter distribution

In this section, we provide a statistical analysis of these features for each of the generators described in [30]. Note that this list is not exhaustive, but contains the generators with existing open-source (or easily attainable) packages.
In addition, this work documents analysis of graphs generated at increasing scales. This is a novel addition to most analyses which do not make evaluations beyond what would be considered "small" in many real-world cases.
The following sections describe the evaluation of each implemented package. For many of the classical graph models, analytical formulations exist with respect to expected values of the feature statistics. If available, a comparison is provided between theoretical values and the computed features for the range of evaluated scales.
The configuration model found in the initial model list was not evaluated. It requires the user to specify the degree sequence as input to the model. Since the purpose of this study is to determine how well generators mimic real graphs, this requirement was not in line with the work. The BRITE package was also disregarded as it is no longer maintained as of 2001. An initial look at the synthetic graphs obtained using PLRG revealed that every node except one had a degree of one. For these reasons, we chose not to further examine these models.

2.1 ERDOS-RENYI

This model generates a random graph by probabilistically creating edges between pairs of vertices using independent Bernouilli trials.
Three different generator packages were used to create Ërdos-Rényi type graphs, namely: APGL, GGEN and NetworkX (*er* and *er-fast*). Each package was evaluated to determine if the resulting synthetic graphs exhibited the expected properties and if they were maintained as the graph scales increased.
Initial investigations indicated that the GGEN package was producing significantly different results when compared to the other generators in this category. Further analysis revealed a flaw in the underlying code. The authors wrote the package evaluating both $a \rightarrow b$ and $b \rightarrow a$ for the creation of an edge (note that this is a valid assumption for a directed graph). As a result, GGEN appeared to be producing double the number of edges as compared to the other packages. Slight modifications were implemented. All analysis for GGEN in this section is done using the altered version of the generator.

Connected components

The number of connected components in Ërdos-Rényi random graphs have been well-studied and shown to have the following properties [16].
Let G be a G(n,p)* Ërdos-Rényi random graph, then the following statements hold:

1. If $p > \frac{(1-\epsilon)\ln n}{n}$, the graph will almost surely be connected; while if $p < \frac{(1-\epsilon)\ln n}{n}$ the graph will almost surely have disconnected components.

2. If $np < 1$, then G will almost surely have no connected components of size larger than $O(\log(n))$.
   If $np = 1$, G will almost surely have a largest component with size $O(n^{2/3})$.
   If $np > 1$, G will almost surely have a unique giant component, while other existing components will contain no more than $O(\log(n))$ vertices.

As illustrated in Figure 1, it can be seen that $\frac{\ln n}{n}$ does indeed form a sharp threshold for this set of data. Above the cutoff, several graphs have more than one component. The largest component is however always unique, large and such that the remaining (connected) elements are no larger than $O(\log(n))$. For example, one of the graphs with $n = 2^{14}$ vertices has 71 distinct components. The largest represents over 99% of the total vertices and the remaining 70 are comprised of single isolated nodes.
These properties are verified for all 4 generators for a range of probability values p and at increasing scales up to the limits of the generators themselves ($n = 2^{16}$ for APGL and $n = 2^{18}$ for GGen and NetworkX).

Average degree

The expected mean degree or average degree $\langle k \rangle$ of vertices in Ërdos-Rényi random graphs is derived using the Binomial theorem such that

$$\begin{aligned} \langle k \rangle &= \sum_{d=0}^{n} d\binom{n}{d} p^d (1-p)^{n-d} \\ &= p(n-1) \approx pn \end{aligned} \tag{1}$$

The theoretical expected values for $\langle k \rangle$ are compared to the experimental values obtained for the generated graphs. The resulting deviations are largest for smaller graphs (Figure 2). As the size of the graphs

*recall: n denotes the number of vertices.

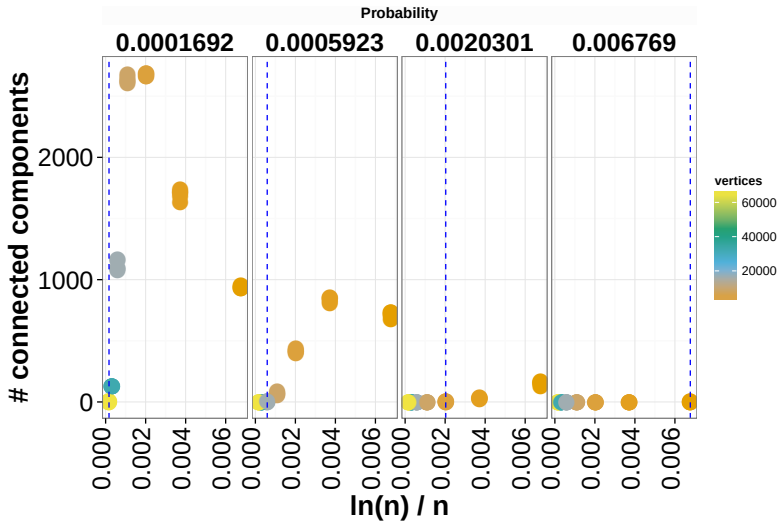


Fig. 1. Connected component transition point in Ërdos-Rényi graphs (APGL).

increase, the values converge. The graphs with the largest deviations appear to contain high numbers of isolated vertices. These observations make intuitive sense given that the larger graphs tend to be composed of a single large component.

Clustering coefficient

The clustering coefficient defines the extent to which nodes in a graph tend to be grouped together. At the individual vertex level, this is referred to as the local clustering coefficient (i.e., how close the neighbors are to being a complete graph). The global clustering coefficient is defined over the whole graph, representing the ratio of the number of actual to theoretical triplets in a graph. One can also compute the (network) average clustering coefficient, defined as the average of the local clustering coefficients over all vertices [42]. In Ërdos-Rényi G(n,p) graphs, the theoretical average clustering coefficient is $E[cc] = p$ (see [29]).

The plot of theoretical values against the expected values reveals that as the size of the graphs scale upward (and hence the size of the single connected component increases), the values converge toward the expected value of p (Figure 3).

The greatest deviations from the expected values of p are obtained for small values of $|V|$ and the smallest p values. This is intuitive since at these values the clustering coefficients themselves are quite small given the low likelihood of creating edges. As a result, a single additional edge can have a large impact on the individual and hence average clustering coefficients.

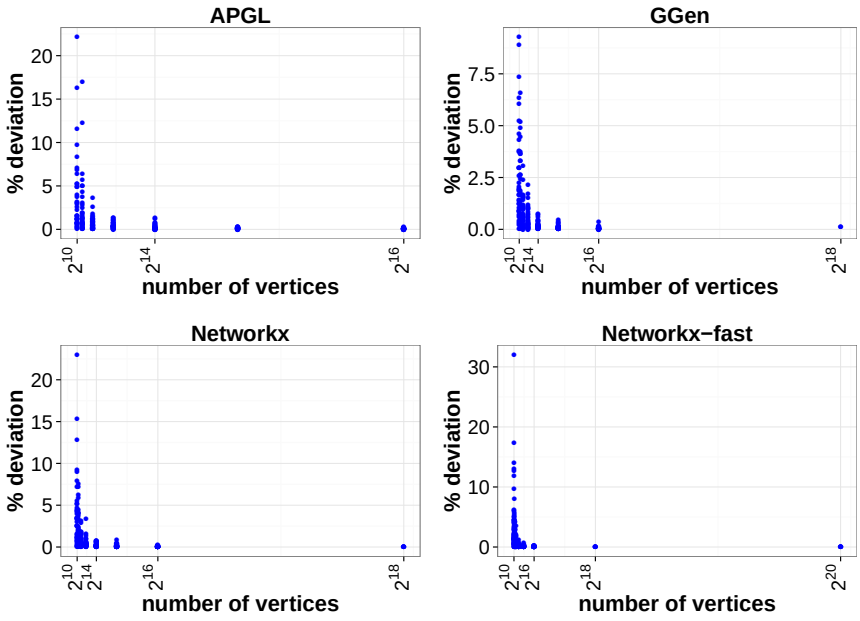
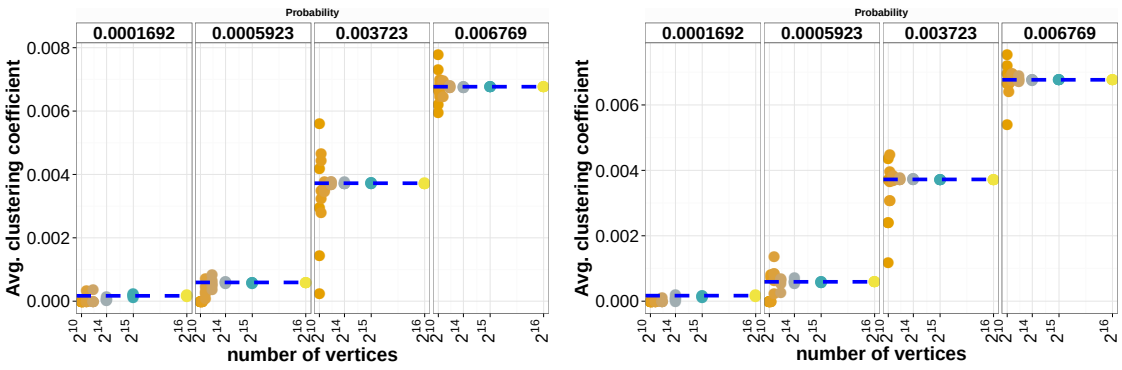


Fig. 2. Deviation of experimental average degrees (Ërdos-Rényi) from expected values.



(a) APGL **(b) GGen**

Fig. 3. Experimental average clustering coefficients compared to expected values (dotted line).

Diameter

The diameter of a graph is defined as the maximum shortest path distance between any two vertices. Chung and Lu [11] proved that almost surely the diameter of a G(n,p) random graph is:

$$D = (1 - o(1))\frac{\log n}{\log\langle k\rangle} \text{ if } np \to \infty$$

This holds true in the limit for the generated graphs (Figure 4). Overall, as the graph sizes increase for each generation method, the diameters are decreasing. As observed by Leskovec et al. [27], the diameter of the largest component shrinks as the graph sizes (and hence size of the largest connected component) increase.

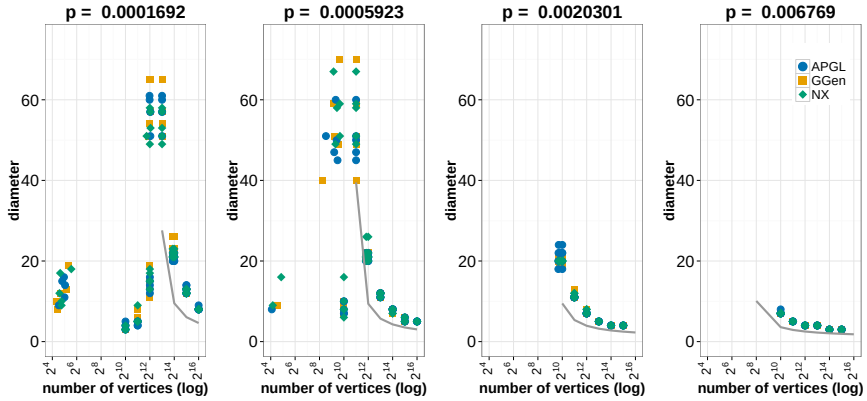


Fig. 4. Evolution of diameter size for increasing Ërdos-Rényi graph sizes. The line represents the theoretical values.

Assortativity

Wang et al. and others [40,41] state that the assortativity of Ërdos-Rényi type graphs is zero (for the largest component), while Newman [33] indicates that the value is zero in the limit for large n.
In the generated graphs, the values converge towards 0 as $n \to \infty$, but tend to be quite disassortative for very small n ($n < 100$). In reality, graphs with $n < 100$ vertices are unlikely to be of interest.

Degeneracy

The degeneracy of a graph is defined as the smallest value d, such that every subgraph has a vertex with degree at most d. This is related to the concept of graph "core" where the maximal k-core value is d.
The four tested Ërdos-Rényi models all display the same linear growth behavior as the graph sizes increase. The rate of increase is greater for larger values of the probability p.

Degree distribution

The degree distribution of Ërdos-Rényi graphs has been shown to be Poisson in the limit for constant values of np [34]. A comparison of the Poisson curves to the degree distributions at each set of np constants from the generated graphs indicates that the experimental values follow the expected distributions almost exactly (Fig. 6).

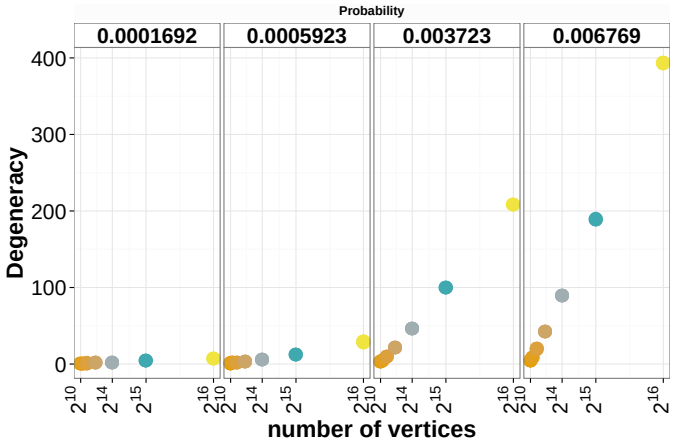


Fig. 5. Degeneracy of Ërdos-Rényi graphs generated using NetworkX at various probabilities.

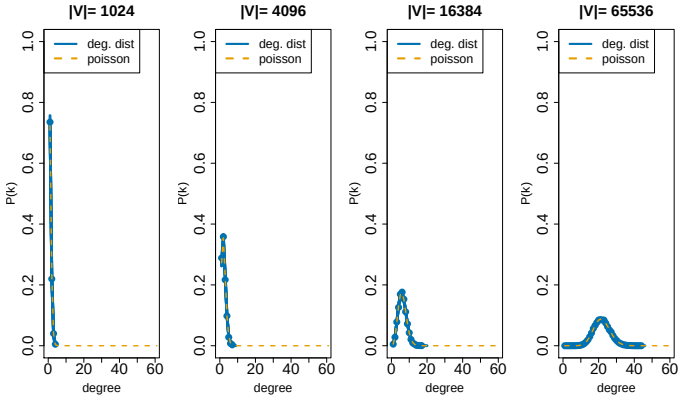


Fig. 6. Degree distributions for Ërdos-Rényi graphs generated using APGL and p=0.0001692.

Core

A k-core of a graph G is defined as the a maximal connected subgraph of G in which all vertices have degree at least k[†].

Each vertex in a graph G belongs to some core k, where k is the minimal degree of all vertices in the core and the size of the core indicates the number of vertices it contains. For fixed probability p, the size of the largest core remains unchanged with respect to the percent of nodes, but the core number grows with graph size.

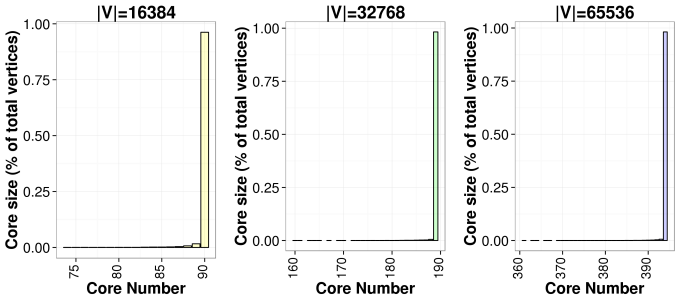


Fig. 7. k-core distributions for Ërdos-Rényi graphs generated using APGL and p=0.0001692.

Centrality

There are many centrality metrics. In this work, we use the vertex "betweenness centrality" metric to ascertain the importance of a vertex in a network, especially with respect to the flow of information along the shortest path. This statistic is calculated as the ratio of the number of shortest paths passing through a node to all shortest paths in the network and is defined for each vertex. As a result, this is a computationally intensive metric to calculate. For a vertex v, this can be expressed as:

$$bc(v) = \sum_{\forall i,j \neq v} \frac{\sigma_{ij}(v)}{\sigma_{ij}}$$

where the denominator σ_{ij} is the number of shortest paths from i to j and $\sigma_{ij}(v)$ are those that specifically pass through v.

Figure 8 shows the bell shaped distribution of centrality values. As the graph sizes increase, there are fewer distinct values (range decreases), but a greater intensity of those present.

Under the assumption that as graph sizes increase, density increases, these results might be expected. However, as seen in the Section 2.1, the average degree of Ërdos-Rényi graphs is bounded. The centrality analysis actually indicates that there are further distinctions to be made as the graph sizes increase, notably that the frequency of "highly" central vertices actually increases. With respect to network robustness, this would be a positive trend.

[†]Wikipedia

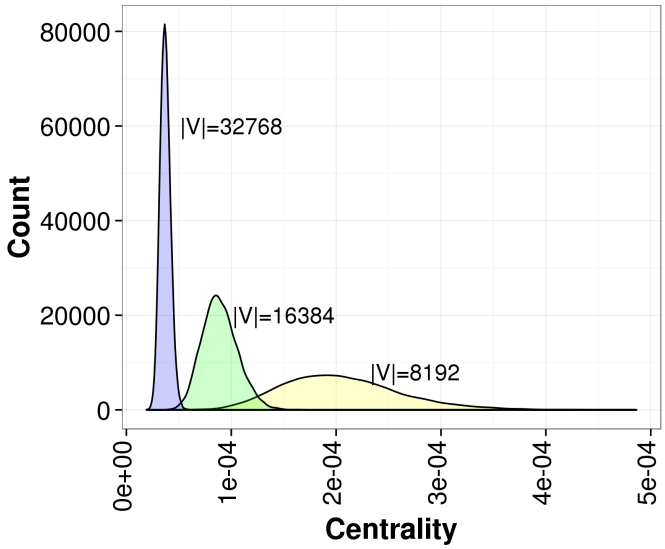


Fig. 8. Betweenness centrality values for Ërdos-Rényi (APGL) graphs with p=0.0001692.

2.2 SMALL WORLD

This random graph model was originally introduced to address the need for higher clustering coefficients by Watts and Strogatz [42]. Two different generator packages were used to create small world type graphs, namely APGL and NetworkX. Both packages generate a graph by first creating a two dimensional ring with each node connected to K neighbors, then rewiring edges based on the input probability p. Each package is evaluated to determine if the resulting synthetic graphs contained the expected properties and if they are maintained as graphs sizes increased.

Connected components

As pointed out by the authors of the NetworkX package, the generated graphs are not guaranteed to be fully connected. For both APGL and NX however, given the selected parameter inputs, all generated graphs are fully connected. Bollobás [7] showed that setting $K \gg \ln n$ guarantees a connected graph.

Average degree

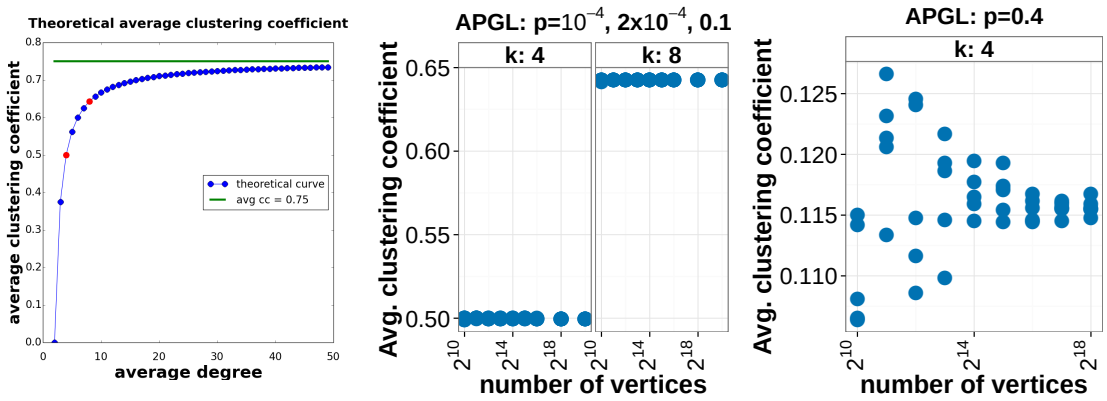
A required input parameter to the small world model is the initial number of neighbors. This, coupled with the rewiring probability p, directly influences the average degree. For the selected parameters, the average degree at all sizes are exactly K under both generators.

Clustering coefficient

For a lattice-type structure such as the Watts-Strogatz model, the (average) clustering coefficient is defined independent of the graph size as [3]:

$$C = \frac{3(K-2)}{4(K-1)} \rightarrow \frac{3}{4} \text{ as } K \rightarrow \infty$$

This equation holds for various values of p up to large p. The 2 models were tested with $K = 4$ and $K = 8$ respectively. The resulting experimental average clustering coefficients match the theoretical values almost exactly at lower p. As p tends toward 1, the graphs become more like random graphs, hence the trend towards lower clustering, such as in Figure 9b (right).



(a) Theoretical scaling **(b) Experimental values**

Fig. 9. Comparison of expected and experimental average clustering coefficients.

Average path length

By definition, small world graphs have small average path length (the arithmetic mean of all shortest path distances in the graph).
A comparison between the 2 generators indicates that they are producing comparable, though not identical, average path lengths at all values of p, $\langle k \rangle$ and computed scales. Results from the Kolmorgorov-Smirnoff test for each combination of parameters indicate that the null hypothesis that the 2 generators produce significantly different values can not be rejected for any of the input parameter combinations.
In [42], the authors indicate that if $p = 0$, the average path length will scale linearly with the graph size. As p increases however, the average path length tends towards $\frac{\ln(n)}{\ln(k)}$. The average path scales either linearly or logarithmically with N. The cutoff value depends on p and N and is defined as $N^* = p^{-1/d}$ with $d = 1$ here for a ring lattice [3]. As reported in [3], the accepted form for the average path length is:

$$\begin{aligned} l(N,p) &\sim \frac{N}{k} f(u) \text{ where } u = pkN^d \\ f(u) &= \frac{4}{\sqrt{u^2+4u}} \tanh^{-1} \frac{u}{\sqrt{u^2+4u}} \text{ and } u \gg 1 \text{ or } u \ll 1 \end{aligned} \tag{2}$$

where the scaling function $f(u)$ was derived by Newman et al. Note that the formula given in [35] is slightly different resulting in curves that are all well below the experimental data points.
Figure 10 illustrates the cutoff between linear and log scaling for the small world generators. The exact plots of l and a scaled version $c * l$ (where c is a constant) are shown as dotted lines. The experimental NetworkX values have slightly wider deviations, but both models converge to the expected values with increased graph sizes. Though not included as part of the larger study, a few graphs using the *iGraph* package were generated for comparison. For the smaller sized graphs, the path lengths follow the two other

models, but start to have lower values as the graph sizes increase. Note that the constant by which the theoretical value must be scaled varies with p.

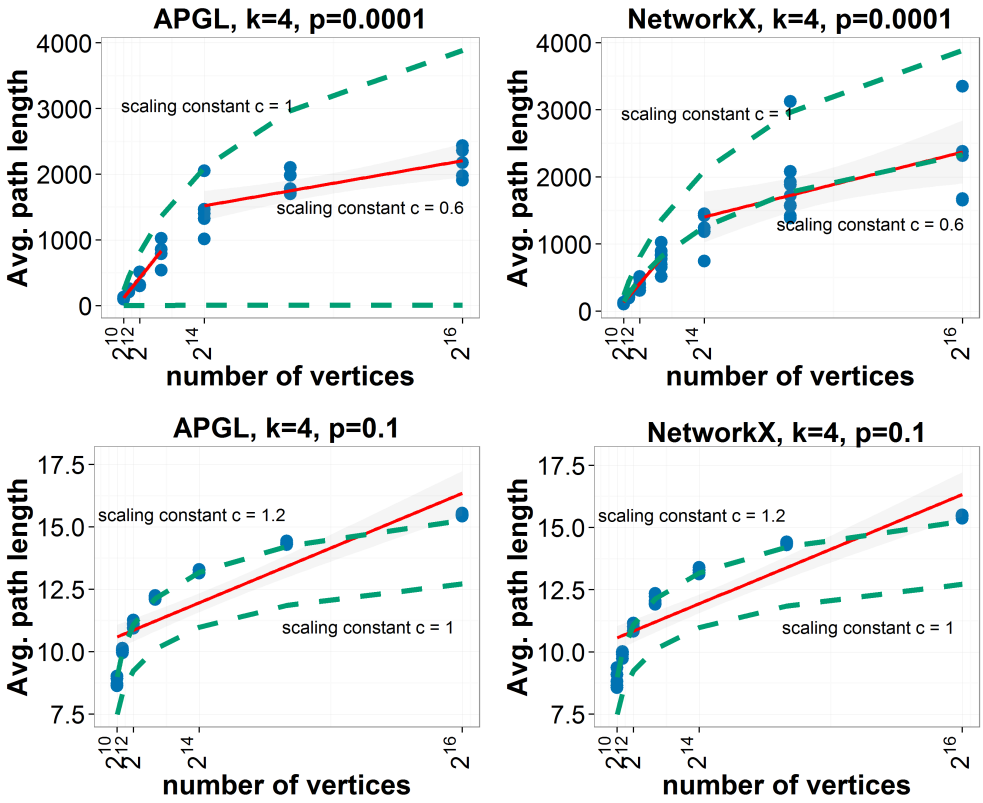


Fig. 10. Small world average path length scaling.

Assortativity

Newman [33] defines the linear degree assortative coefficient using the Pearson correlation coefficient of the degrees of the vertices:

$$\rho_D = \frac{\sum_{xy} xy(e_{xy-a_x b_y})}{\sigma_a \sigma_b}$$

Ciglan et al. [12], among others, have reformulated this for ease of computation as:

$$r = \frac{\frac{1}{M}\sum_i j_i k_i - [\frac{1}{2M}\sum_i (j_i + k_i)]^2}{\frac{1}{M}\sum_i (j_i^2 + k_i^2) - [\frac{1}{2M}\sum_i (j_i + k_i)]^2}$$

where M is the number of edges and j_i and k_i are the degrees of the ith edge.
As pointed out for random Ërdos-Rényi graphs, the assortativity values for small world graphs are close to 0, due to the fact that edges are initially placed precisely and then rewired randomly without respect to edge degree.

Degeneracy

The degeneracy of a graph is defined as the maximum k-shell number[‡]. Alternatively, the vertices in every subgraph in a k-degenerate graph will be of degree at most k. The degeneracy of k-regular graphs have a degeneracy of exactly k, otherwise the degeneracy has been shown to be strictly less than the maximum degree [22].
The graphs from both small world generators follow these findings. A closer look at the data indicates that k-degenerate graphs correspond to those where the average degree of the graph is K and every vertex has the same degree (which naturally leads to a k-degenerate graph). All other graphs have degeneracy of $K - 1$ independent of size or p.

Degree distribution

The small world models are designed such that average degree $\langle k \rangle$ is an input to the model and is maintained for nonzero values of p. The degree distribution has been shown to have the form [3]:

$$P(j) = \sum_{n=0}^{\min(j-\langle k \rangle/2, \langle k \rangle/2)} C^{n}_{\langle k \rangle/2}(1-p)^{n} p^{\langle k \rangle/2-n} \frac{(p\langle k \rangle/2)^{j-\langle k \rangle/2-n}}{(j-\langle k \rangle/2-n)!} e^{-p\langle k \rangle/2} \text{ for } j \geq \langle k \rangle/2 \tag{3}$$

This equation is independent of the size of the graph and depends primarily on the choice of $\langle k \rangle$ and p. The computed theoretical probabilities (Table 3) correspond closely to the observed values in the generated graphs. Figure 11 illustrates the impact of p on the peakedness of the curve which occurs at $\langle k \rangle$, then decays exponentially. As pointed out by [3], for the most part, nodes have similar degrees, creating a homogeneous network.

Table 3. Theoretical degree distribution probabilities for k=4 and p=0.0001.

k=4, p=0.0001		k=8, p=0.0002	
j	**P(j)**	**j**	**P(j)**
0	0	*0*	0
1	0	*1*	0
2	9.998e-9	*2*	0
3	1.999e-4	*3*	0
4	**0.9996**	*4*	1.599e-15
5	1.999e-4	*5*	3.197e-11
6	3.998e-8	*6*	2.397e-7
7	1.333e-12	*7*	7.989e-4
8	6.664e-17	***8***	**0.9984**
9	2.665e-21	*9*	7.987e-4
10	8.885e-26	*10*	3.195e-7

[‡]Note: the k in the k shell is distinct from K (the number of neighbors each node is connected to at the initial setup of the graph ring) and k_i (the degree of the i^{th} edge)

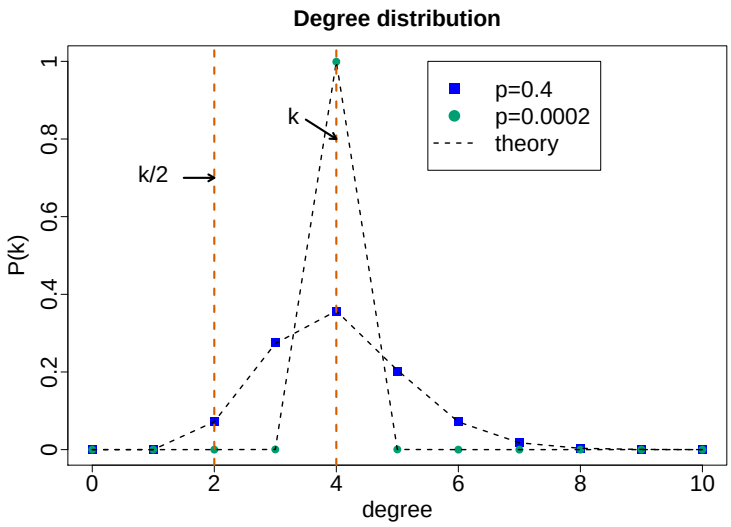


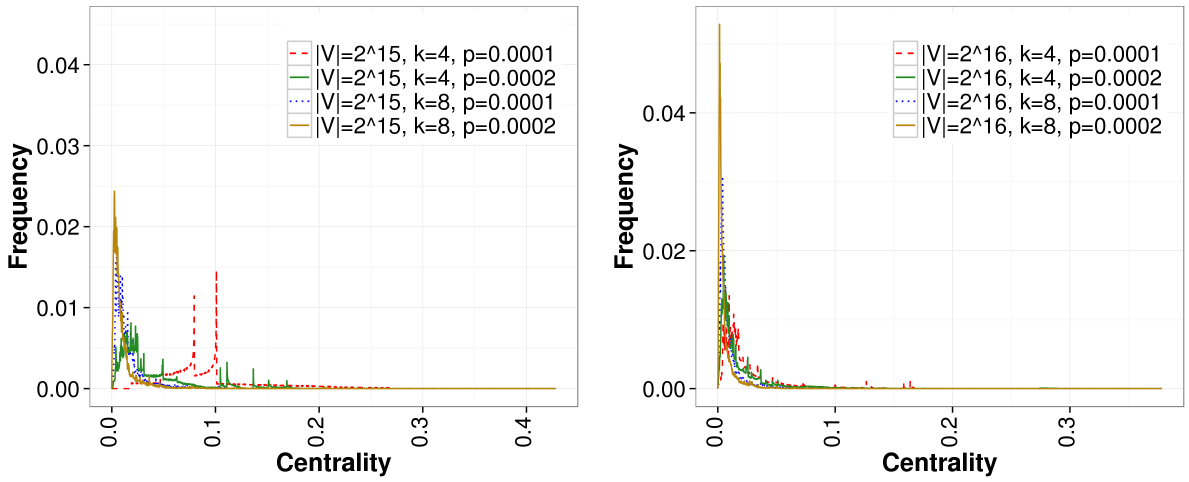
Fig. 11. Degree distributions for a small world graph generated using APGL.

Centrality

The "betweenness centrality" plots shown in Figure 12 indicate that as the average degree k increases, the distributions become more skewed toward zero. This trend is also present with increased graph scales. Overall, the majority of vertices tend to have small non-zero centrality values. Since the small world graphs were created with the intent of having high clustering, this translates into only a few highly central nodes as is evident here.

Core

The small world models create networks where, for any of the tested parameter sets, all of the vertices belong to same (small) k-core. A shift in the trend is noticeable as the the graphs become more random, starting around p=0.1 for the tested cases.



(a) $|V| = 2^{15}$ **(b)** $|V| = 2^{16}$

Fig. 12. Distribution of betweenness centrality values for different small world parameter inputs.

2.3 PREFERENTIAL ATTACHMENT

This random graph model was originally introduced by Barabási and Albert to address the unrealistic degree distributions generated by both the Ërdos-Rényi and Watts-Strogatz models [5]. The approach adds new vertices to the graph by creating m edges preferentially with existing vertices that have higher degrees. The resulting graph is "scale-free" (i.e., a graph whose degree distribution follows a power law).
The APGL and NetworkX generator packages implement this model. Each package requires specification of the total number of desired vertices and the number of edges m to add to each new vertex. The resulting synthetic graphs are evaluated against the theoretically expected behaviors.

Connected components

The generation method adds m edges for each new vertex to the graph, one at a time. As long as $m > 1$, the graph will have a single connected component. The case of $m = 0$ is not considered as it would produce a completely disconnected graph; an uninteresting proposition. To avoid this, the NetworkX generator requires $m > 1$.

Average degree

The average degree is defined as

$$\bar{d} = \frac{2 * |E|}{|V|}$$

The experimental results show this to hold true for all graphs and both generator implementations.

Clustering coefficient

There does not appear to be an analytically derived formula for the average degree of a Barabási-Albert model [3]. Empirically, it has been seen that the clustering coefficient decreases with graph size and follows a power law of the form:

$$C \sim N^{-0.75}$$

Graphs generated from both packages follow a power law distribution, but the exponent is a bit lower than the cited 0.75, with values closer to $N^{-0.55} - N^{-0.6}$. The higher the input value for m, the lower the exponent. While it has been suggested that the preferential attachment model produces graphs with homogeneous clustering [23], the work by Fronczak et al. [18] finds this not to be true. Using mean field theory, Fronczak et al. derived the formula for the average local clustering coefficient given by equation 4.

$$C = \frac{6m[(m+1)^2(\ln N)^2 - 8m \ln N + 8m]}{8(m-1)(6m^2 + 8m + 3)N} \tag{4}$$

Fronczak et al. show that for $m \gg 1$ and large N, this formula tends toward that of a random graph with a power-law degree distribution. This provides a very close match to the experimental results (Figure 13).

Average path length

In [3], the model authors show that the empirical average path length l follows the form:

$$l = A \ln(N - B) + C$$

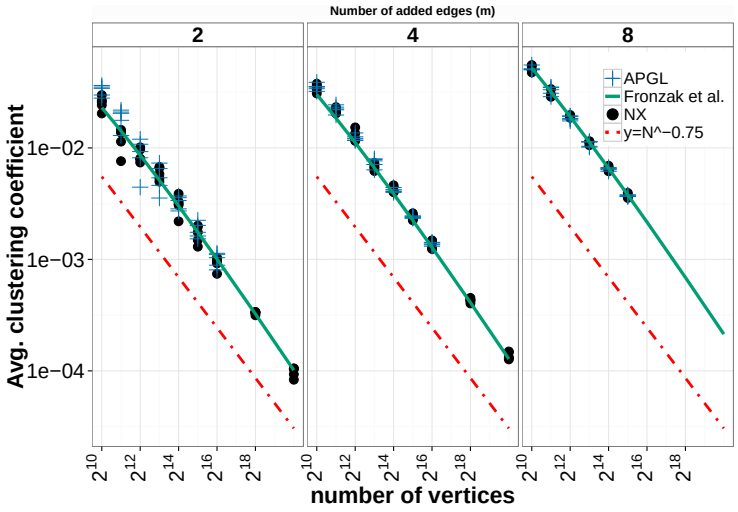


Fig. 13. Clustering coefficient scaling of Barabási-Albert models.

More rigorous analytical results by Bollobás and Riordan as well as Cohen and Havlin indicate that the scaling is closer to

$$l \sim \frac{\ln N}{\ln \ln N} \tag{5}$$

for $\ln \ln N \gg 1$ and $m \geq 2$ [8, 13]. The formula was derived specifically for the diameter of a graph, but the average path length is expected to behave similarly.
The experimental results are on the order of Equation 5 for both APGL (Figure 14) and NetworkX. While this equation does not depend on the input value m, the plots indicate that higher values of m are closer to the "ultra small world" scaling.

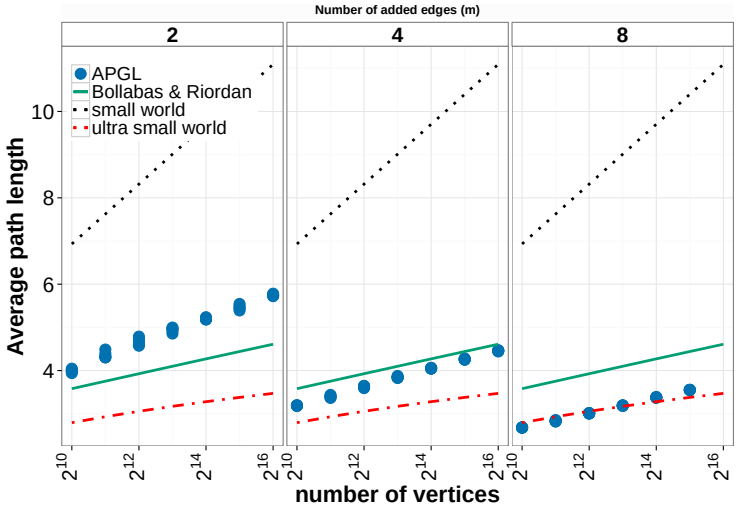


Fig. 14. Average path length scaling of the preferential attachment model.

Assortativity

Assortativity indicates a preference (or not) for nodes to connect to similar nodes. Newman [33] indicates that for the Barabási-Albert preferential attachment network, the assortativity is 0. This is shown to be true in the limit for large values of N. These theoretical findings hold true for both the APGL and NetworkX models. The plot is omitted as it is uninteresting.

Degeneracy

The Barabási-Albert preferential attachment model generates graphs with bounded degeneracy [15]. This can be shown by noting that each node is added to the graph with m edges (input parameter). The last added vertex therefore has at most m edges. It follows that the graph is at most m-degenerate. The experimental results align exactly with this analysis.

Degree distribution

One of the primary objectives in developing the small world model was to have a model that created graphs with more realistic degree distributions. The resulting Barabási-Albert network degree distributions are independent of scale (scale-free) with a power law distribution of the form:

$$P(k) \sim k^{-3}$$

The exponent value was determined experimentally, then subsequently using mean field arguments.
For the APGL and NetworkX generated graphs, plots of the degree distributions coupled with the expected power law distribution indicate that they indeed follow a power law distribution modulo a constant. The results are independent of the size of the graph as expected.

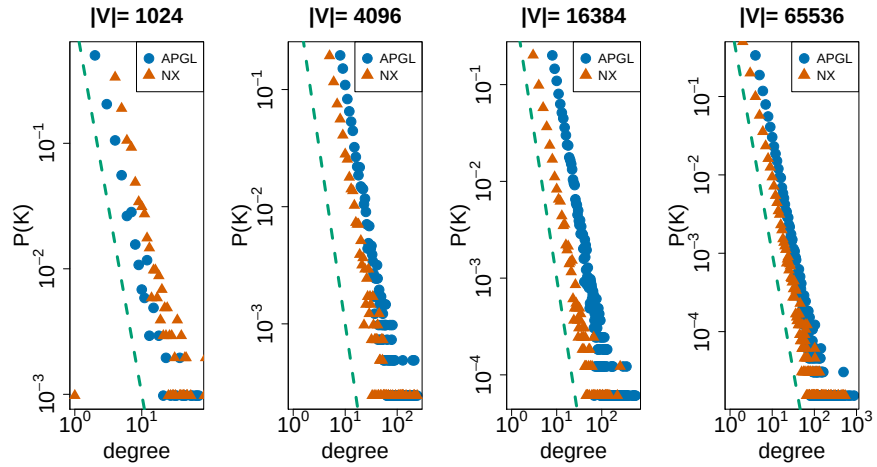
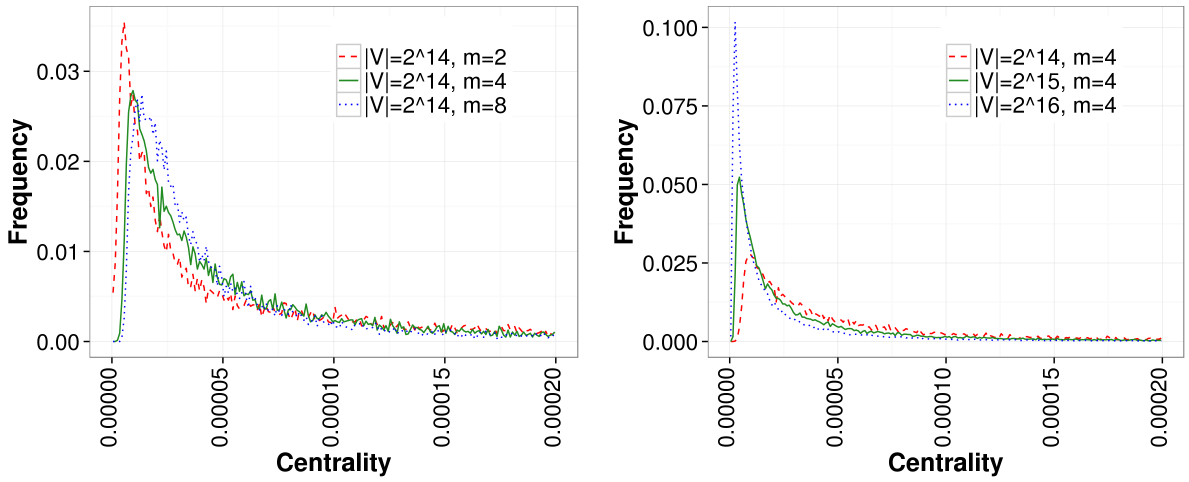


Fig. 15. Log-log plots of the degree distributions for the preferential attachment model. The dashed line shows the expected theoretical distribution.

Centrality

The centrality distributions of the synthetically generated preferential attachment graphs have a skewed bell-shape. For constant graph sizes, an increase in the number of added edges (m) results in a decrease of the maximum centrality value. The bell of the curve decreases in height but increases in width indicating a greater number of low centrality values. By increasing m, more paths are created, reducing the high centrality of some nodes.
Similar observations can be made when holding m constant and increasing the graph sizes. Figure 16 shows a shift toward zero. The plots have been truncated to show the detail at lower centrality values. The preferential attachment model created graphs where most vertices have very similar degree. As a result, the centrality values are in the same range for a majority of nodes in the graph.



(a) $|V| = 2^{14}$**, m=2,4,8** **(b)** $|V| = 2^{14}, 2^{15}, 2^{16}$**, m=4**

Fig. 16. Distribution of the centrality values for different preferential attachment parameters.

Core

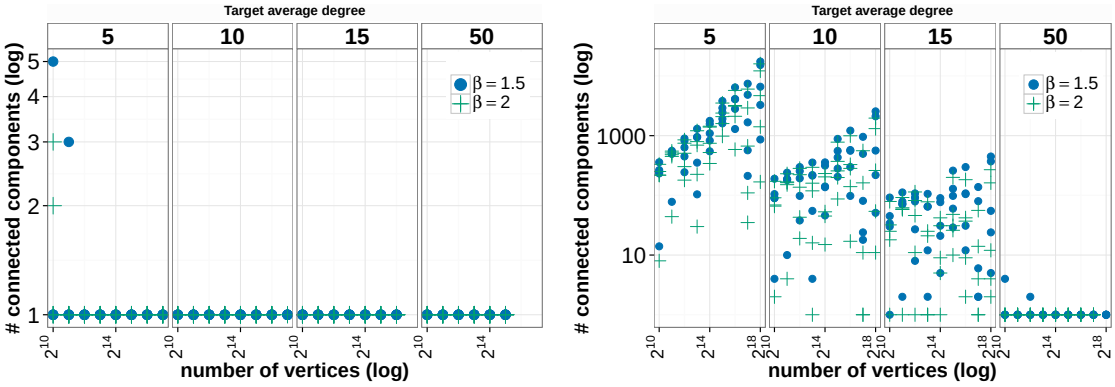
Similar to the small world model, all vertices in each of the generated graphs belong to same core. The plots have been omitted.

2.4 RANDOM HYPERBOLIC GRAPHS

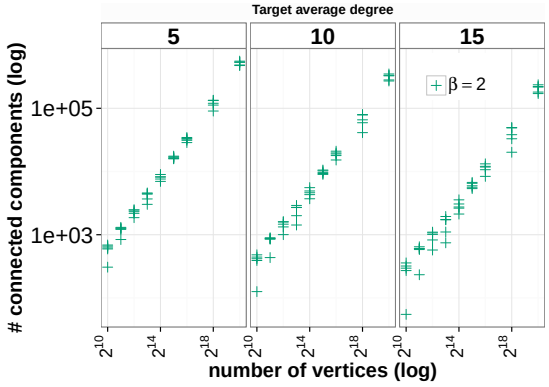
Random geometric graphs have been studied for many years (see [19]), but recently a generative model for the family of random geometric graphs on hyperbolic spaces was introduced by Krioukov et al. [25]. Note that this is a key difference between this model and others which use a Euclidean space.
Briefly, the graph is created by randomly selecting N points (vertices) within a hyperbolic space, using a Poisson point process. Edges exist between vertices if the distance between them is less than some specified distance R (e.g., if the space is a circle, R is the radius). The initial developers of the model indicate that the resulting graphs have desirable real world properties such as heavy tail degree distributions (heterogeneous) and strong clustering resulting from the hyperbolic geometry. The authors also claim that several of the original random graph models (e.g., configuration and classic random graphs) are contained within this model as limiting cases. To the best of our knowledge, the only generator package of this type to date is by Krioukov, and was obtained from the author directly.
Gugelmann et al. [21] and Bode et al. [17] provide a thorough review of the model from a mathematical standpoint. Their work will be used as a comparison point for the experimental results produced using the Krioukov generator. Of note are the similarities these graphs appear to have with large scale Internet graphs in particular. The authors claim that this is the first model that satisfies both power law degree distribution and a large clustering coefficient. Through the analysis in the next sections, these claims appear to be possible, but may be a bit overstated. A thorough knowledge of the model would be required along with very specific model parameterization.

Connected components

The model has two primary formulations, H2 and S1, requiring input parameters α, β, γ and d. The parameter β is the inverse of the temperature with $\beta > 1$ representing the cold regime and $\beta < 1$ representing the hot regime. d is the expected average degree and γ is the power law exponent. Note that the cases $\beta = 1$ and $\gamma = 2.0$ are degenerate resulting in empty graphs.
Bode et al. [6] indicate that the α parameter influences the connectivity. Specifically they claim that if $\alpha > 1$, the largest component grows sublinearly and for $1/2 < \alpha < 1$, the graph is disconnected with high probability. The authors show that the threshold is $1/2$ such that for $\alpha < 1/2$, the graph will be connected with high probability. Note that α is defined primarily for the H2 model, while the experimental graphs were generated under the S1 model. Statistically, according to Krioukov, the results are the same, but S1 runs faster, thus why it was selected for evaluation. One can interpolate between the 2 models using $\gamma = \frac{2\alpha}{\zeta} + 1$ where $\zeta = 1$ when operating on a plane such as in the generated graphs. The implication is that setting $\gamma < 2$ should result in connected graphs.
The experimental results are in line with the claims in [6]. Though, we see in Figure 17b for values close to the limit (α just under 1/2), the average input degree must be high in order for the graph to be fully connected.



(a) $\gamma = 1.8$ **(b)** $\gamma = 1.9$



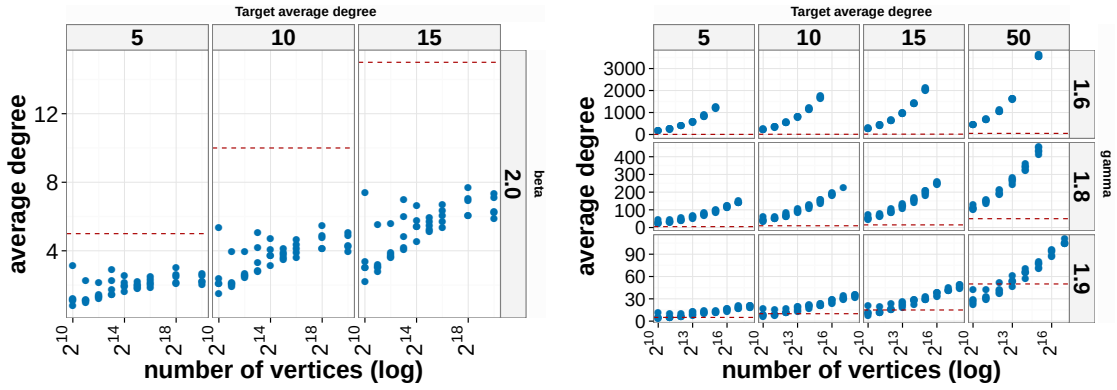
(c) $\gamma = 2.1$

Fig. 17. Comparison of the number of connected components at different γ levels.

Average degree

The experimental results indicate that the average degree is about half of the input value d. The model authors indicate that this is due to "finite size effects" [25]. Personal correspondence with the code author indicated that a change in the equations used in the implementation may rectify this and remains a path for future work [§]. The number of zero degree (isolated) nodes may also need to be accounted for and resolved. When $\alpha < 1/2$ (i.e., $\gamma < 2$), one can consider this a "dense" case and the average degree grows with N, otherwise it is almost constant [6]. Experimentally this is seen to be true (Figs. 18a and 18b) with little variation for $\gamma = \{1.9, 2.1\}$ and an exponential increase in average degree for graphs with $\gamma < 1.9$.

[§]We thank Blair Sullivan for sharing her correspondance with the model author.

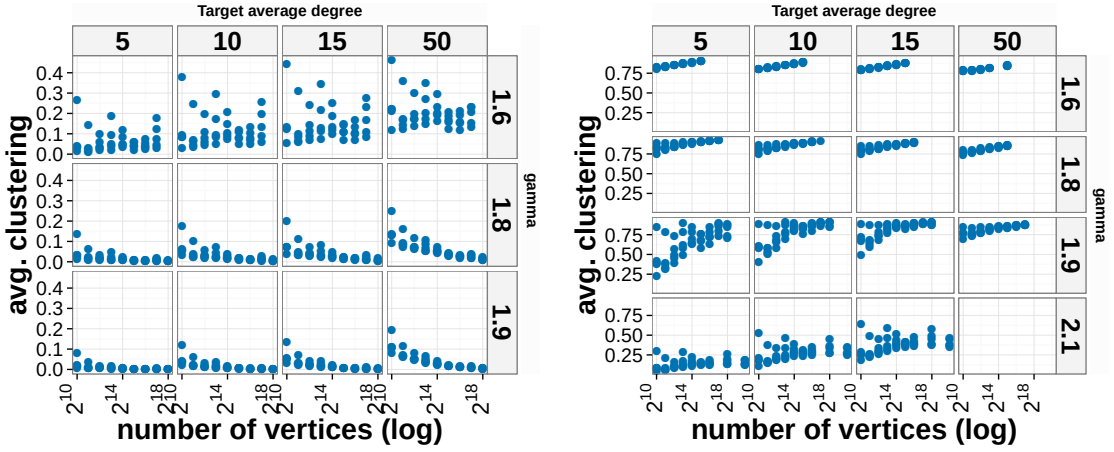


(a) $\gamma = 2.1$ **(b)** $\gamma < 2$

Fig. 18. Average degree variations for the Krioukov model cold regime ($\beta > 1$). The dotted line indicates the expected average degree d.

Clustering coefficient

The average local clustering coefficient (i.e., density of the neighborhood of a vertex) is shown by Gugelmann et al. [21] to be asymptotically bounded away from zero with high probability when $\beta > 1$. Experimentally, these values appear more constant across increasing graph sizes with lower values for $\gamma > 2$. The obtained values are only slightly below those shown by Krioukov et al. [25] for $\gamma = 2.2$.



(a) $\beta < 1$ **(b)** $\beta > 1$

Fig. 19. Average clustering coefficient for the hot and cold Krioukov regimes.

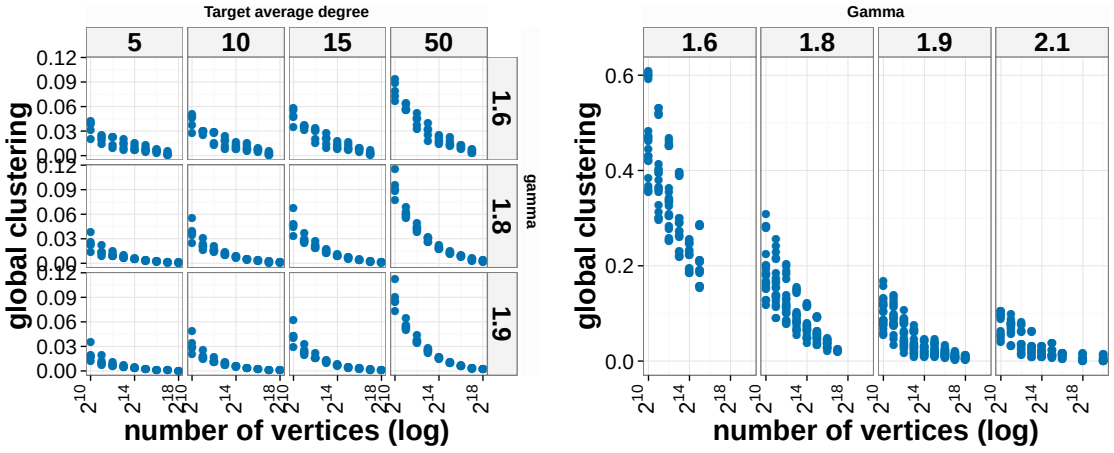
The global clustering coefficient (i.e., the number of triangles formed) is studied in depth for the cold regime with $\beta > 1$ by Candellero et al. [9]. The authors show that the value is bounded away from zero with an abrupt change when β crosses 1. Rigorous proofs are given indicating that this feature can be parameterized by β, α, and ζ alone. The authors note that several studies prove that this model follows a power law with an exponent between 2 and 3 and is sparse with a few high degree nodes mimicking real world graphs.

For both the hot and cold regime, if $1 \leq \zeta/\alpha < 2$ (equivalent here to $2 < \gamma \leq 3$), Candellero et al. [9] shows that the global clustering coefficient converges to 0 as seen in Fig. 20a.

The case for $\gamma < 2$ or $\zeta/\alpha > 2$ is not explicitly studied by any of the previous authors. Experimentally, the values are converging, though in the cold regime this may be bounded above 0. Further rigorous analysis is needed to determine the lower bound.

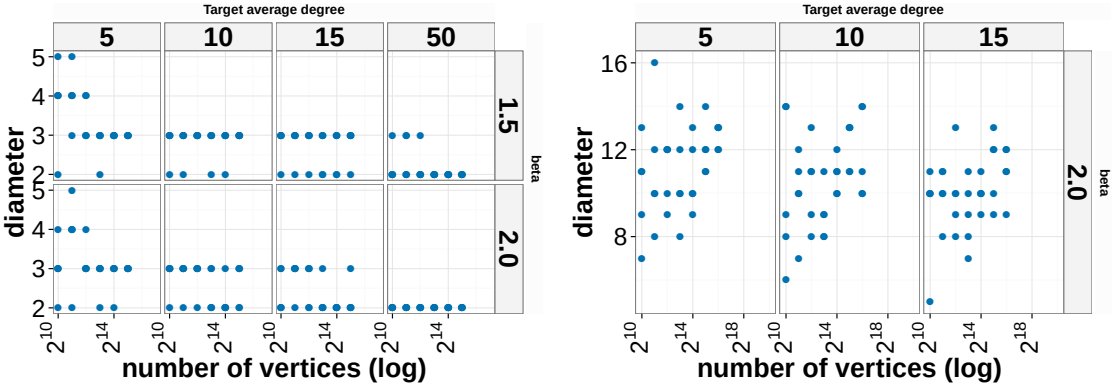
Diameter/path length

Fountoulakis [17] notes that analysis of the expected behavior of the diameter and path length for this model is still an area of open research. The simulation results indicate that values are constant and independent of increasing graph sizes, but are slightly affected by increased target average degree. This is intuitive as the latter increases the number of edges in the graph, thereby shortening paths and reducing the diameter of the graph.



(a) $\beta < 1$ **(b)** $\beta > 1$

Fig. 20. Global clustering coefficient for the different Krioukov model regimes.



(a) $\gamma = 1.8$ **(b)** $\gamma = 2.1$

Fig. 21. Krioukov diameter variations for $\beta > 1$.

Assortativity

Analytical results for the assortativity do not appear to be documented in the literature. The synthetically generated graphs show highly dissaortative patterns. In the cold regime with $\gamma > 2$, the assortativity is negative and fairly invariant with respect to the graph size. A small increasing trend is visible for values of $\gamma < 2$. The most dissassortative graphs are obtained for the lower values of γ.
In the hot regime, the assortativity is again mostly invariant to graph size, but the graphs tend to be much less dissassortative.
This matches results seen in empirical social networks with negative (or close to 0) assortative coefficients.

Degeneracy

For both the hot and cold regimes, the degeneracy of the graphs increases with graph size. For the first part of the size spectrum ($n = 2^{10} \sim 2^{12}$), the increase is linear with a slope varying based on the value of γ.

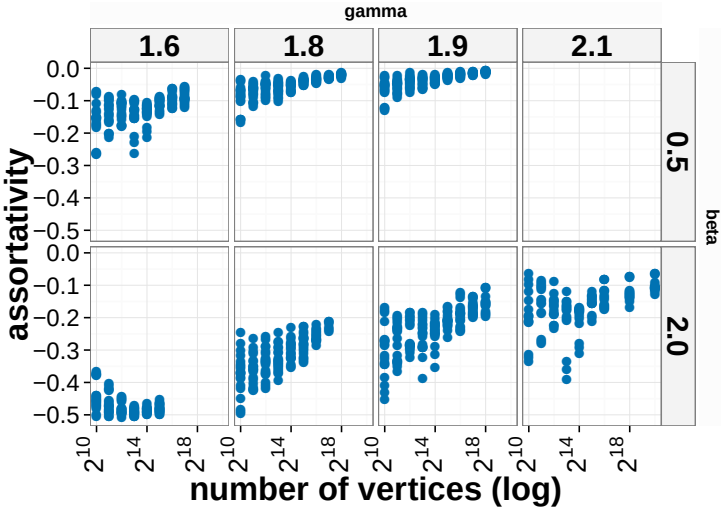
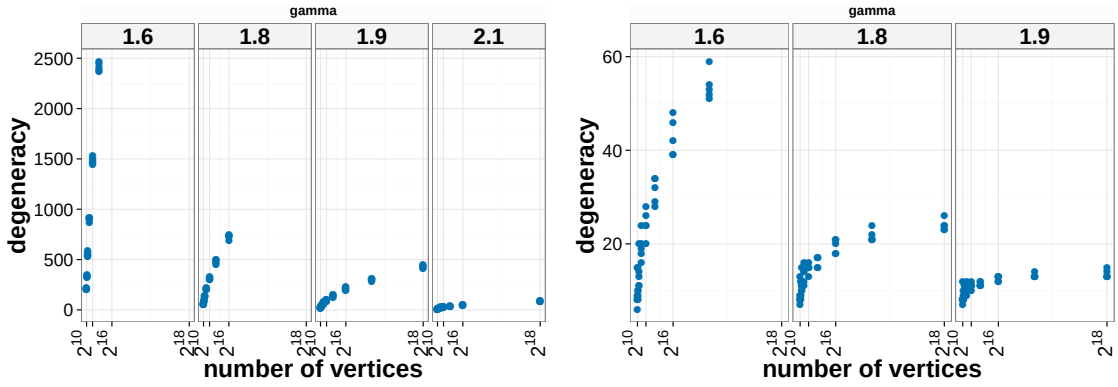


Fig. 22. Assortativity for both cold and hot Krioukov regimes.

Beyond $n \simeq 2^{13}$, the growth is still linear, but the slope is drastically reduced, appearing to converge. Note that the convergence is faster for larger γ. For $\beta > 1$, the magnitude of the degeneracy values is much higher.



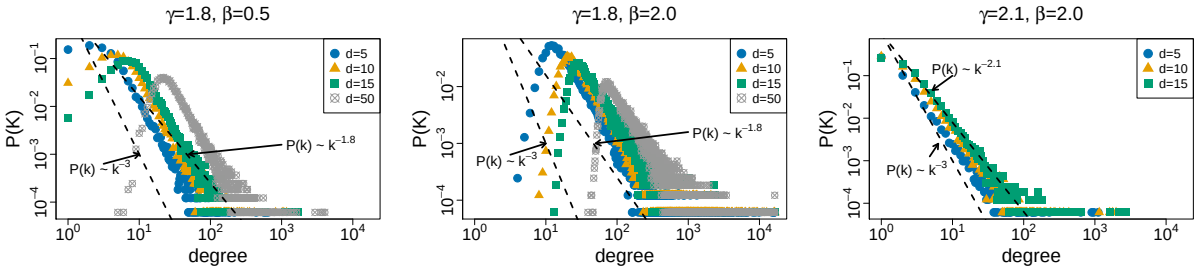
(a) $\beta > 1$, target degree = 15 **(b) $\beta < 1$, target degree = 15**

Fig. 23. Degeneracy under hot and cold Krioukov regimes.

Degree distribution

One of the objectives of this model is to provide "realistic" power-law degree distributions. Under the hyperbolic model, the degree distributions are scale free. Fountoulakis [17] proved that for $\beta > 1$, the degree distribution follows a power law. Figure 24 illustrates this to be mostly true with respect to the experimental results. For the hot regime, we observe that only the tail of the distribution is power-law, with the overall curve resembling more of a log-normal distribution. For the cold regime, lower values of the γ parameter create distributions that resemble the hot regime. In addition, the tail of the distribution, while it

does look like a power-law, it does not match the expected value of the power-law exponent. The higher the expected average degree d, the more this hold true.



(a) Hot regime **(b) Cold regime** **(c) Cold regime**

Fig. 24. Degree distribution variations under different Krioukov model regimes.

Centrality

A vertex has a centrality score ranging from 0 to 1. If a vertex has no edges, it has a value of 0. In computing the betweeness centrality, the shortest-path determination may or may not include the end points (definition dependent). In this analysis, endpoints were not included. There are several ways in which a node might have zero centrality other than by being isolated. Figure 25 illustrates a few of these cases.

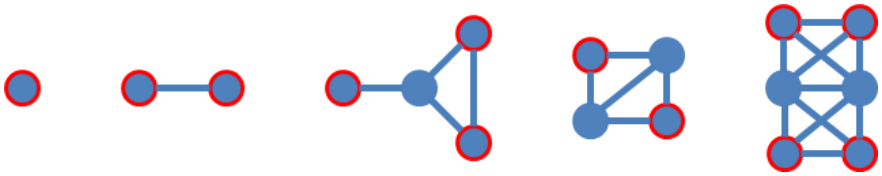


Fig. 25. Illustrative examples of vertices with zero centrality (red outline).

The centrality scores obtained for graphs generated with the Krioukov model display a high frequency of zero values. In particular, with higher γ values, there are more zero values. This shift is visible in Figure 26, where under the cold regime ($\beta > 1$), as the value of γ increases, there are more non-central vertices. Conversely, under the hot regime ($\beta < 1$), more vertices have non-zero centrality values as γ decreases. With the high frequency zero values removed, Figure 27 illustrates the change in centrality distributions as γ and β are varied. The majority of vertices still have very low centrality values. As γ decreases, the maximum centrality values decrease indicating a more even distribution of network connections.

Core:

As reflected in the degeneracy, the core sizes increase when lowering γ. Significantly lower values are also obtained for $\beta < 1$. While not explicitly shown, from the degeneracy plots we expect that these values will also increase with increasing graph size.
Figure 28 shows plots with constant graph size. If one increases the size under the cold regime (top line of plots in Fig. 28), the resulting figures are very similar with a longer tail to the right (not shown).

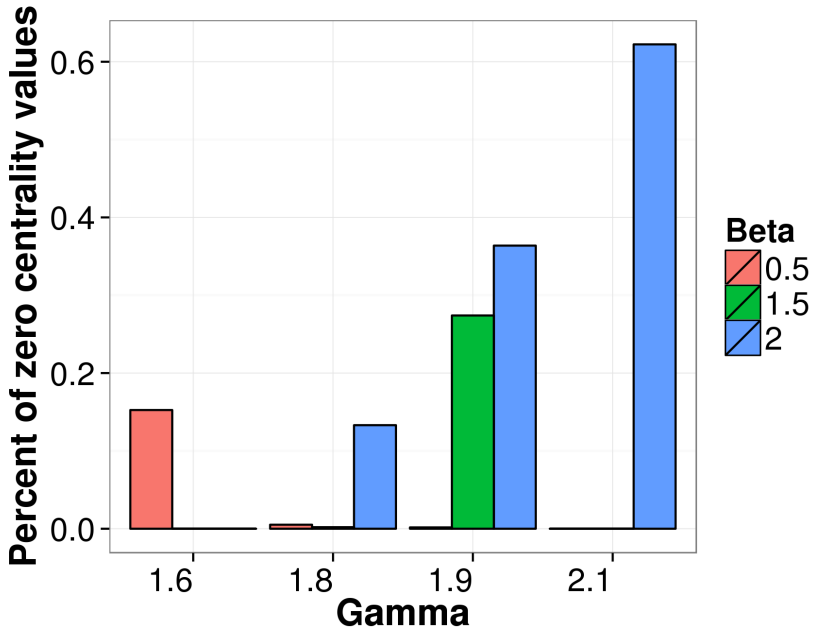


Fig. 26. Percentage of zero centrality values for different parameters settings using the Krioukov model ($|V| = 2^{14}$ and $d = 15$).

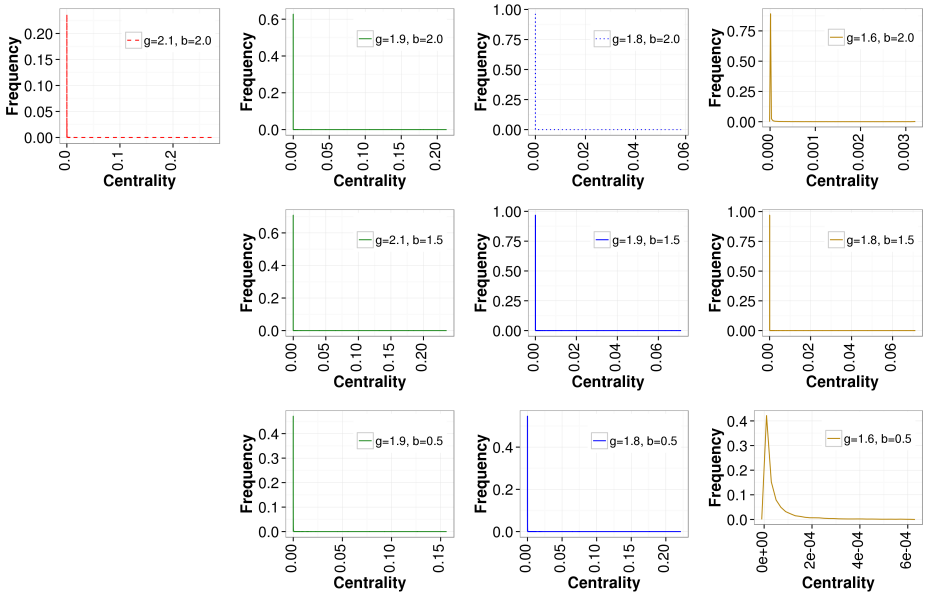


Fig. 27. Distribution of the centrality values for the Krioukov model with $|V| = 2^{14}$ and $d = 15$.

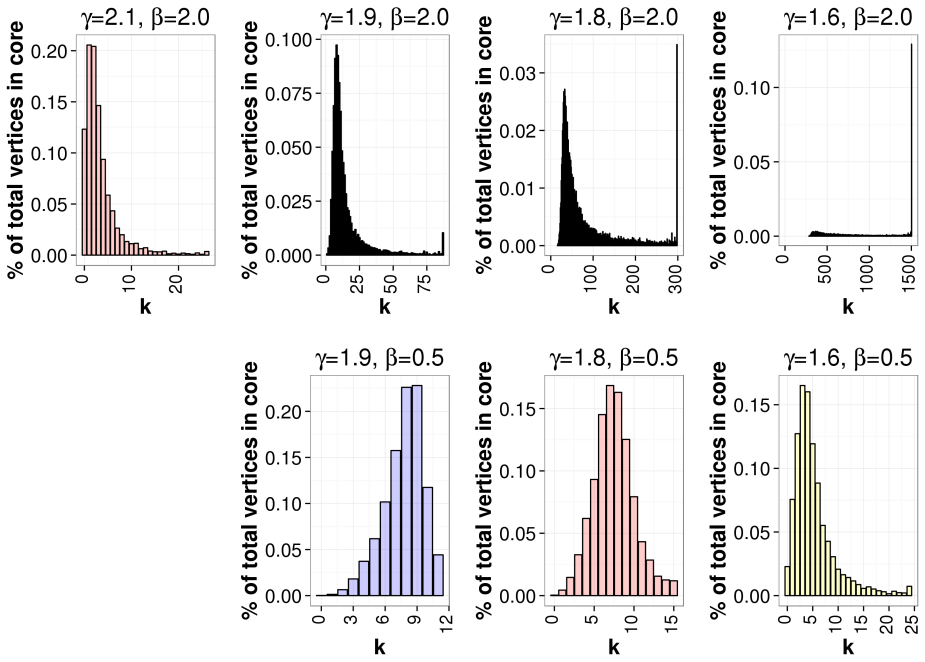


Fig. 28. Distribution of the k-core sizes with the Krioukov model for $|V| = 2^{14}$ and $d = 15$.

Observations

While several authors, specifically Fountoulakis [17], provide rigorous mathematical analysis of Krioukov's modeling approach, there remain many unanswered questions with respect to this model. In particular, the diameter of the random graph should be further investigated as well as the typical distance between two vertices belonging to the same component [17]. Additionally, further analysis regarding the distribution of components outside of the largest component or an investigation of the dependence of the clustering coefficient on β could be undertaken.

The model authors have stated that other classic models can be represented using the Krioukov model. It would be useful to determine the appropriate parameter settings which generate these cases.

2.5 R-MAT/KRONECKER

Introduced by Chakrabarti et al. [10], the R-MAT model was designed for simplicity and speed, generating a wide range of graph types. The algorithm fills an initially all zero adjacency matrix by diving it into quadrants, selecting a quadrant with probability a, b, c or d ($a + b + c + d = 1$) and repeating this process until a single 1x1 array is reached, whereby an edge (denoted by a "1" entry) is added to the graph.

The Kronecker model is a generalization of R-MAT proposed by Leskovec et al. [26] in order to provide a model that closely matches real-world networks. Under this recursive model, graphs are created by repeatedly using the Kronecker product. The authors propose both a deterministic and a stochastic version. Under the first scenario, the Kronecker product of an initiator graph G_1 adjacency matrix is taken k times, resulting in a larger graph G_1^k. Under the stochastic variant, the initiator matrix is composed of probabilities; edges exist based on the Kronecker product of this matrix. The authors claim that using a 2x2 probability matrix gives an R-MAT generator.

The Pywebgraph package (R-MAT based) and Graph500 implementation (R-MAT or stochastic Kronecker-based) were used in this analysis. After initial tests, it was found that the Pywebgraph package contained a code error, which resulted in quadrants being selected incorrectly. The code was modified appropriately.

As noted by Seshadhri et al. [38], the Graph500 implementation of the Kronecker method implements recursive quadrant selection similar to the R-MAT generator, thus the 2 methods should provide similar results. In the experiments, the quadrant probabilities $p = \{0.57, 0.19, 0.19, 0.05\}$ were used for both generators. For a true testing of the Kronecker variant, a different generator may need to be sought out.

Connected components

In the experimental results, both generators produce a large number of disconnected components. As the sizes of the graphs increase, the number of distinct connected components increases sub-linearly. At the smaller sizes, the ratio of vertices to connected components is about 4.5, while for larger graphs, this value reduces to about 2. Since edges are generated independently of one another, there is no guarantee that graphs will be fully connected. In addition, the selected quadrant probability also impacts the placement and hence connection of the graph.

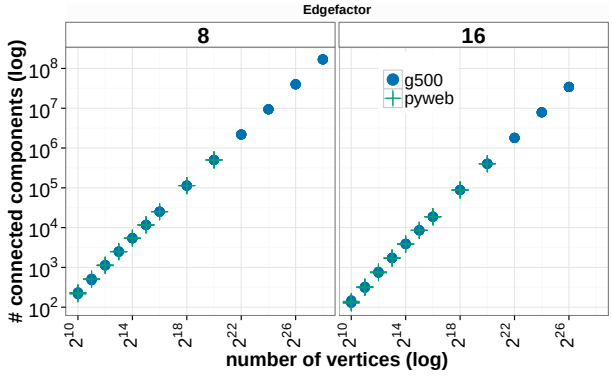


Fig. 29. Log-log plot of the number of connected components in Kronecker synthetic graphs.

Average degree

Both implemented generators require as input the "edgefactor" parameter. This indicates the average degree of the vertices in the graph and directly influences the number of edges created.
In [38], Seshadhri et al. observe that the output from the Graph500 benchmark contains a high number of isolated vertices which leads to a higher than desired average degree in the largest component. In the synthetic graphs generated, the average degree increases sub-linearly as the size of the graphs increase and is well above the desired value (Figure 30). The impact on the average degree increases as the graph sizes increase. Seshadhri et al. demonstrate that this is due in part to the lack of noise incorporation into the benchmark. The same graphs re-run with noise produce fewer isolated vertices resulting in lower average degrees for the largest connected components.

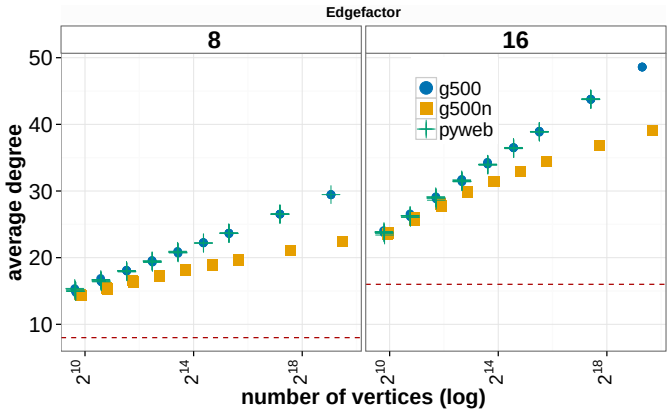


Fig. 30. Comparison of the average degree for the largest connected component in Kronecker graphs. The dotted line indicates the expected average degree.

Clustering coefficient

As the graph sizes increase, the average local clustering coefficients decrease sub-linearly with the size of the graph. The largest generated graphs have very small average clustering coefficient values.

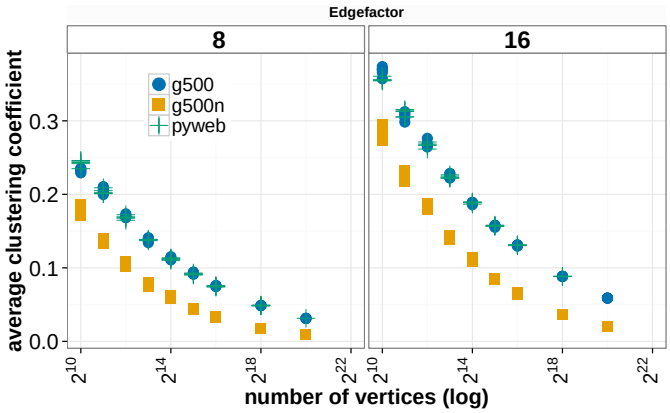
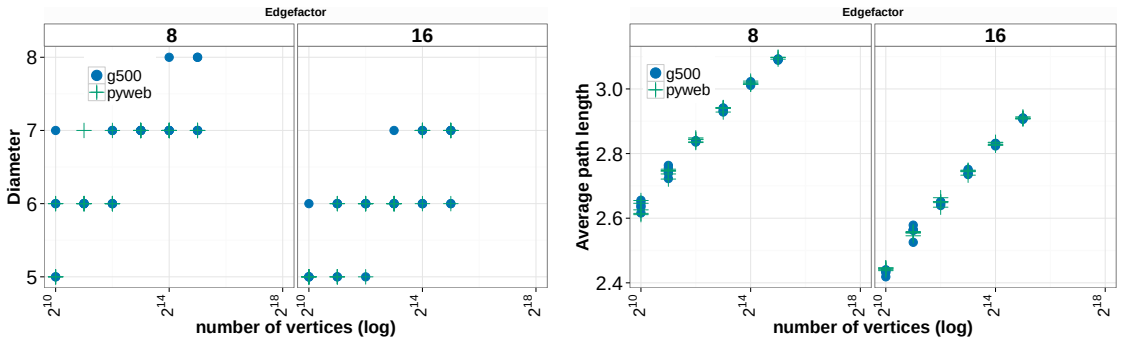


Fig. 31. Average clustering coefficient for Kronecker models.

Diameter/path length

For the synthetically generated graphs, the diameter increases very slightly as the graph sizes increase for both generators.

The average path length is a computationally intensive metric requiring the calculation of shortest paths between all pairs of vertices. As a result, several values were unable to be computed at the higher scales, but the trend is visible from the obtained results, whereby the average path length appears to increase approximately linearly.



(a) diameter **(b) average path length**

Fig. 32. Semi-log plots of the changes in diameter and path length for Kronecker models.

Assortativity

For both generator packages, the assortativity is disassortative, but trends towards 0 as the graph sizes increase.

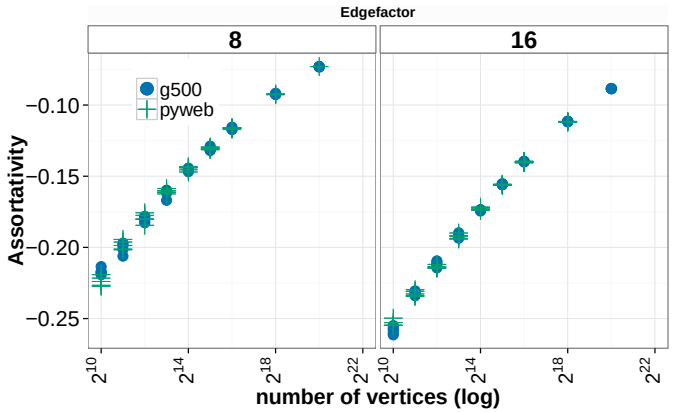


Fig. 33. Changes in Kronecker model assortativity at increasing scales.

Degeneracy

The degeneracy (or max k-core) of the graphs increases with the number of vertices. Seshadhri et al. [38] point out that in general this model generates values that are much lower than reality. By selecting input parameters (number of levels and skew) appropriately, this is less apparent, but still increases at best linearly with the expected average degree. The same authors also note that, as a result, the generated graphs will not have the same localized density as real web graphs or social networks. By adding in noise, the degeneracy of the graphs is lower, with a slower overall increase with graph size.
Note that this observation does not hold true under the experimental results. We are seeing max k-core on the order of 54-130 for 2^{14} where the authors report around 16. The analysis code has been verified and validated, we can only surmise a difference in input parameters or a different implementation of the method.

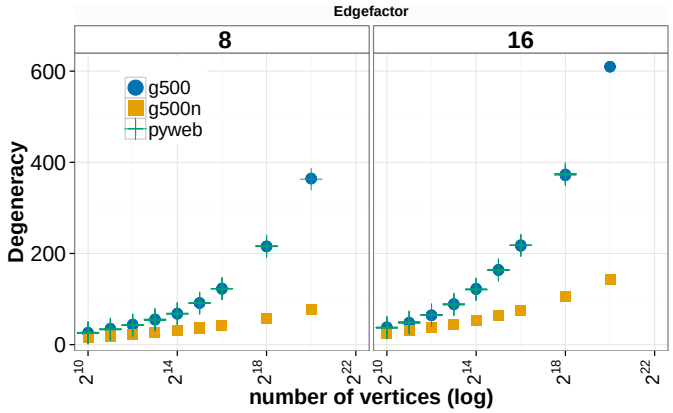


Fig. 34. Changes in Kronecker model degeneracy at increasing scales.

The skew is defined as $\sigma = a + b - 0.5$. In our experiments $\sigma = 0.26$ which is towards the higher end of the range and should produce larger max core numbers. As shown in the paper [38], higher powers of twos (i.e., larger graphs) produce larger max core values. Based on [38] however, it would appear that real graphs remain steady with increasing parameters.

Degree distribution

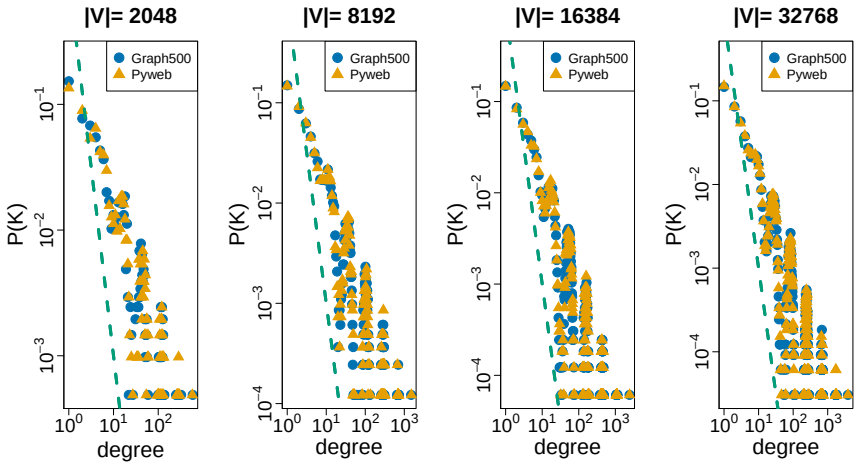


Fig. 35. Degree distribution compared to a power law with $p \sim k^{-3}$.

While many have pointed out that real-world graphs tend to follow a power law degree distribution, the output from both generator packages in this category display some critical oscillations while decaying in a semi-power law fashion. Groër et al. [20] prove that the distribution for the R-MAT model is in fact a multinomial distribution. Seshadhri et al. [38] show that by adding noise, the model redistributes the edges among the vertices, thus smoothing out the output, resulting in lognormal behavior. By including noise, the output is then closer to the expected distribution (Fig. 36). Noise reduces the number of isolated vertices and smooths out the function.

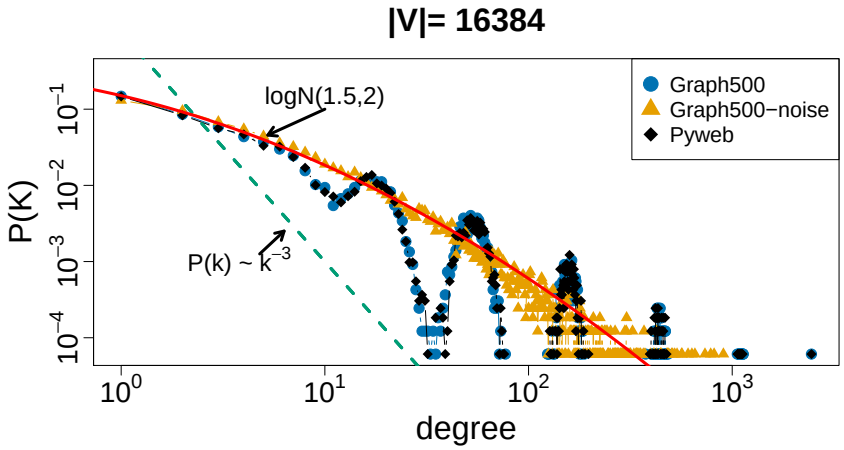


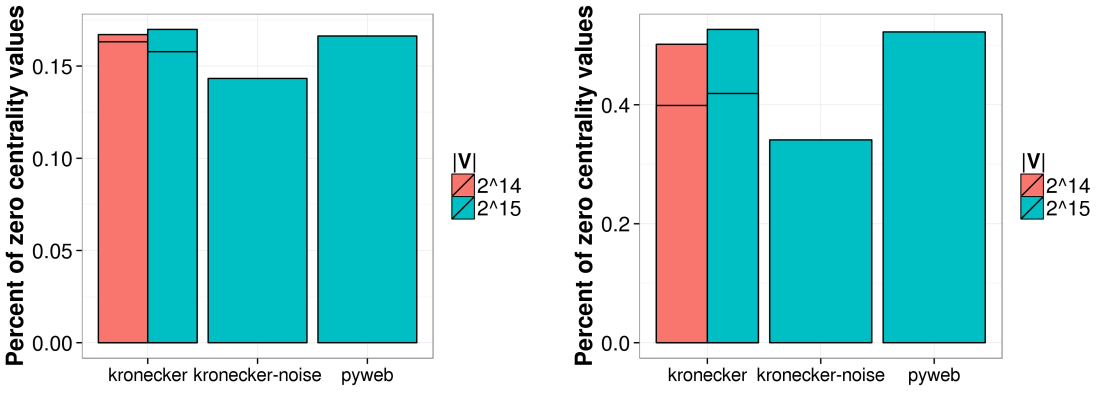
Fig. 36. Degree distribution with & without noise compared to power law and lognormal fits.

While others have reported results on much larger graphs for this model (e.g., Seshadri et al.), our analysis is limited by choices made at the onset: synthetic graphs must be generated in less than a day as well as any statistics.

Centrality

The frequency of zero centrality scores is close across the 2 different model types (and parameter settings) with respect to the connected vertices. If the disconnected nodes are included, the percentages rise significantly (as discussed in the connected components section) and the inclusion of noise attenuates the total number of zero values.

Removing the zero-values reveals similar distribution trends with most values very close to 0 and the maximum values around 0.1 - 0.12 (edgefactor dependent) with the exception of the noise-included case, which has a much lower maximal value, indicating the absence of nodes with as high of centrality.



(a) Connected vertices **(b) Isolated and connected vertices**

Fig. 37. Percentage of zero centrality values for graphs generated using the Kronecker model.

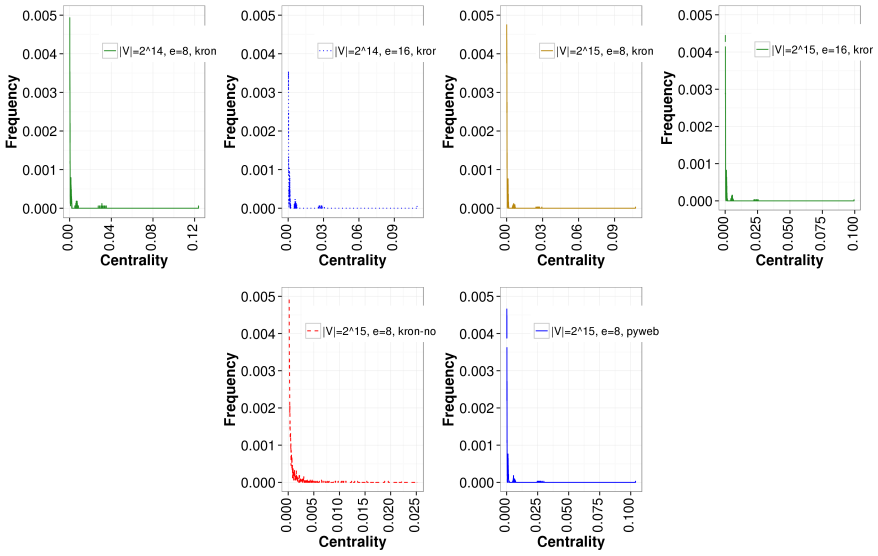


Fig. 38. Distribution of the centrality values for the Kronecker model(s).

Core

The k-core data resulting from both Graph500 and Pywebgraph packages have nearly identical results. Figure 39 diplays the Graph500 results, where the large number of isolated vertices is reflected in the core of size $k = 0$.
By increasing the edgefactor e, the number of isolates is reduced and thus the size of the 0-core, with an increase in the two largest cores. Overall, the general trend is in the form of a dying exponential, with 2 small peaks at the end.
The cycling evident in the degree distributions is also present in the core distribution. The inclusion of noise in the model significantly smooths out any variations.

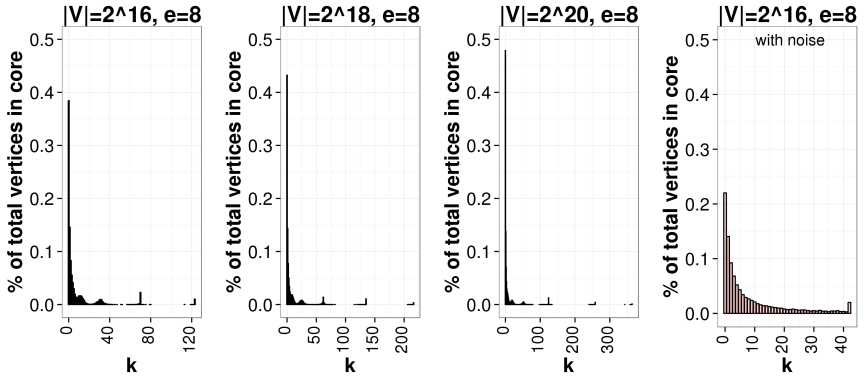


Fig. 39. k-core distribution for synthetic graphs using the Graph500 model.

2.6 INTERNET MODELS

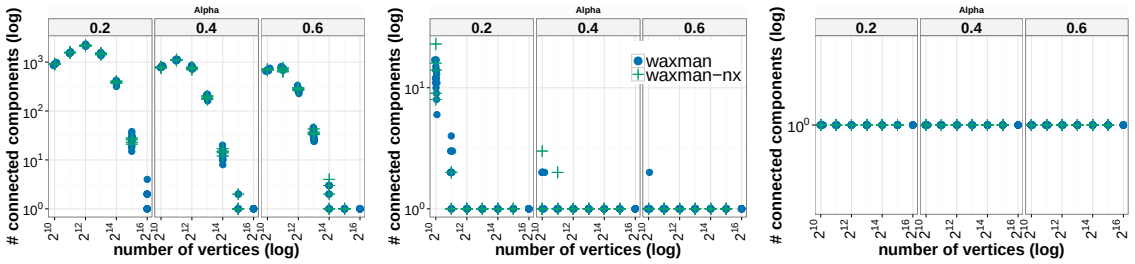
In 1999, Faloutsos et al. made the claim that the Internet follows a power law distribution. This resulted in a fair amount of research related to networks (graphs) and degree distributions.
In this section, we use "Internet" to refer to Internet topologies. There are many different types of generators with different foci or goals. Of the existing models, this report considers the Waxman (2 generator versions), Inet-3.0, Tiers and GT-ITM (transit stub) models. PLRG is also put into this category, but was discarded once initial observations revealed that every node except one had a degree of one. The BRITE package was also not analyzed as it has not been maintained since 2001.
The Waxman model was the first Internet topology generator and a precursor to other models. It places vertices randomly on the plane, then connects them according to probability $p = \alpha e^{\frac{-d(i,j)}{L\beta}}$ where α controls the density of the graph, β governs the ratio of short to long edges and L is the longest edge distance[¶]. A variation of this method selects L at random (resulting in vastly different results).
Tiers [14] and GT-ITM are hierarchical level topology models designed to integrate the tiered structure of the Internet, improving upon the Waxman model. Unfortunately, as noted in [36], while this may be a better model of the domain interconnections, it may also result in inaccurate representations of the large-scale behavior of the degrees for topologies with more than 100 nodes. It also appears that the Tiers model is no longer maintained.

[¶]The original definition by Waxman (1998) inverts the parameters α and β with $p = \beta e^{\frac{-d(i,j)}{L\alpha}}$.

The Inet-3.0 generator (the most recent version) is designed to incorporate the power law degree distribution, seeking to mimic random networks similar in nature to the status of the AS-level of the Internet from November 1997 to Jun 2000 (and beyond). There does not seem to be an updated version of the count/model since this time.
A useful area of future research would be to try to develop a model based on a more recent mapping of the AS-level of the Internet, such as those made available by CAIDA [2]. The change in Internet structure since 2000 will heavily influence the comparisons between current generator simulation results and real datasets. An additional downside to the cited models is that most look at networks ranging in size from 6000 - 8000 nodes, which may no longer be applicable for today's larger networks.
Models that are more recent than 2010 for Internet topologies do not appear to exist. Since existing models focus on a different aspect of the topology, they cannot be ranked. A model integrating all of the aforementioned properties would be very useful.
Given the current models, Inet and PLRG are best for capturing the degree distributions. If the hierarachical properties are the focus, then Tiers or GT-ITM would be a better choice [31] though the power law structure pointed out by Faloutsos et al. will be lacking.

Connected components

For the data generated from the Inet-3.0, Tiers, GT-ITM models, all components are fully connected. This is intuitive based on the model designs. Van Mieghem [39] and Naldi [32] have presented analytical findings for the Waxman model, in particular for the node distances, link probabilities and connectedness with respect to input parameters.
The 2 generators implemented for the Waxman model create both connected and disconnected graphs depending on the selected values of α and β. In general, as β increases, the probability of link connection increases and the graphs become less disconnected. Full connectivity is seen for $\beta \geq 1$. Increasing the α parameter value increases connectivity, as well as increasing the graph sizes while keeping parameter values constant. These last two observations stem from an increase in graph density but appear to have a smaller effect than β. These findings align with the link probability curves given by Naldi [32] for the rectangular grid (of which the square plane used in the presented models is a simple subset).



(a) β=0.01 **(b) β=0.05** **(c) β=0.1**

Fig. 40. Change in the number of connected components using the Waxman model.

Average degree

The Inet-3.0, Tiers and Waxman models deviate significantly from one another with respect to the average degrees. Figure 41 shows the case where the average degrees have the closest match. This, however, uses

low values of α and β for the Waxman model, from which a departure results in rapid increases in the average degree (Figure 42). In general, the graphs from the Waxman model have very high average degree.

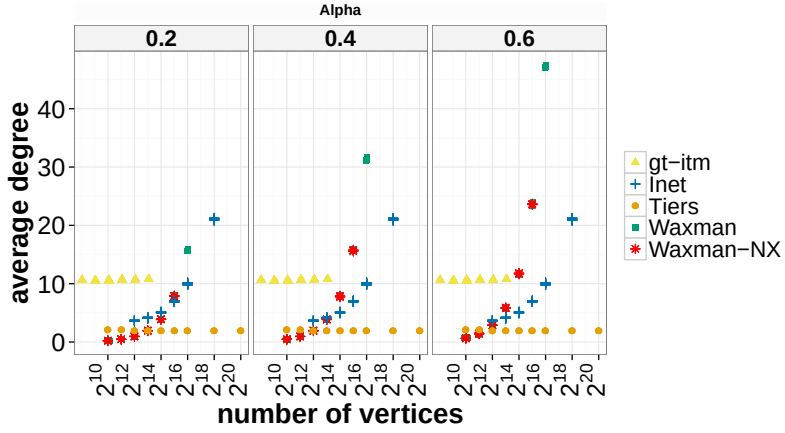


Fig. 41. Comparison of the average degrees for all internet topology models.

Inet-3.0, Tiers and GT-ITM do not follow similar average degree patterns. The origin of this stems from the fact that the models create different number of edges (Figure 43a) for the same size graph. Tiers creates anywhere from 2-3 times as many edges as the other two models, growing as the graph scales increase. This difference will affect many of the features.
The average degrees of the Inet-3.0 graphs increase exponentially as the graph sizes increase. The number of edges of the Tiers graphs is roughly equivalent to the number of vertices in the graph which results in fairly constant values of the average degree.
In the existing Tiers model implementation, only 1 WAN can be modeled, resulting in fewer edges and thus very low average degree.

Average clustering coefficient

Analytical formulas are not readily available for the average local clustering coefficient of any model in this category. Experimentally, while the input parameters produce some variations, the Waxman model is seen to create graphs with low values regardless of graph size (Figure 44).
The Tiers model produces very low average clustering coefficient values which decay almost to zero as the graph size increases. On the other hand, the Inet-3.0 model shows an increase in values converging around 0.82 (Figure 45). The Inet-3.0 authors note that, in general, the model produces values that are lower than true Internet values. We surmise that this statement was made for smaller graphs and that when $n > 2^{14}$, the clustering values are too high. Since, at smaller scales, the Inet-3.0 results are considered too low, it stands to reason that all of the values produced by the Tiers model are significantly too low. The GT-ITM model produces the highest clustering of the 3 models, with high values that continue to increase slightly as the graphs increase in size. Based on the data and comments from the Inet-3.0 authors, this model is also producing unrealistic clustering in the synthetic graphs.

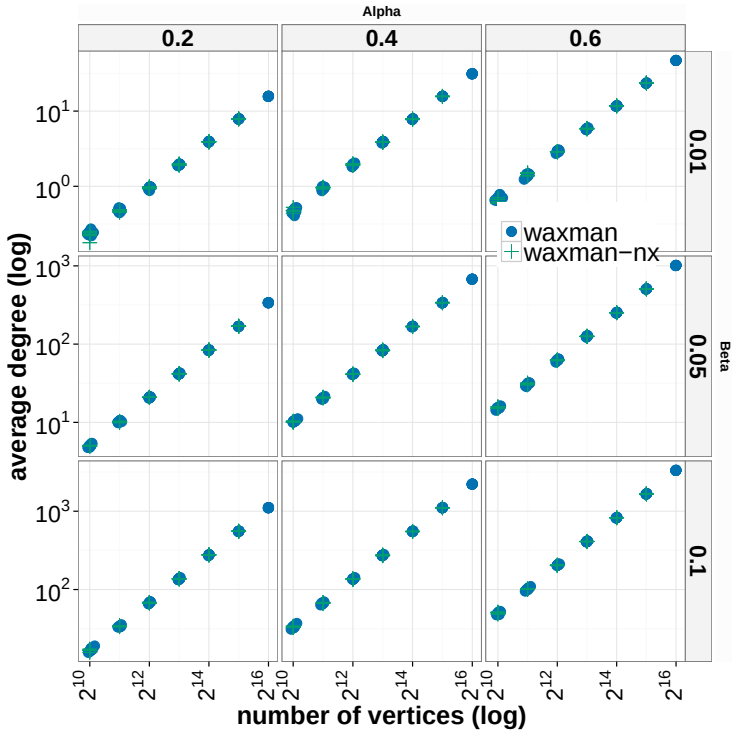
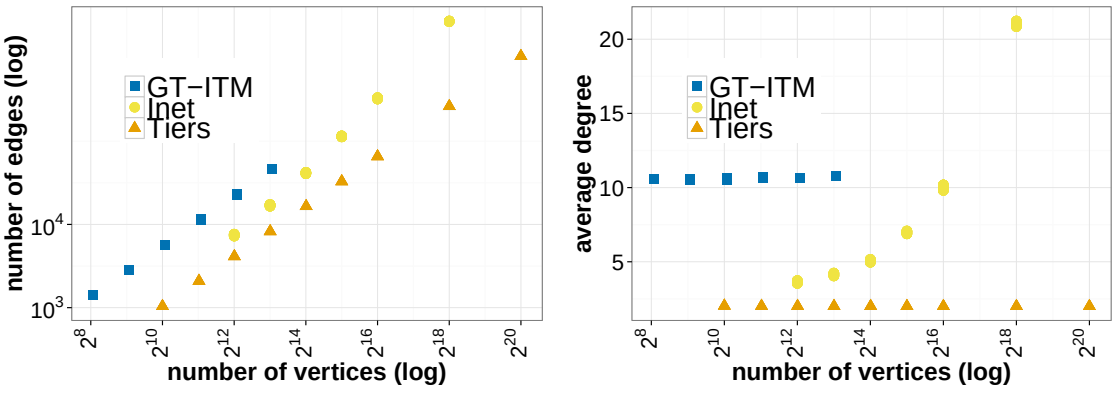


Fig. 42. Comparison of average degree growth using the Waxman model.



(a) edges **(b) average degree**

Fig. 43. Comparison of edges and average degree variations for 3 Internet topology models.

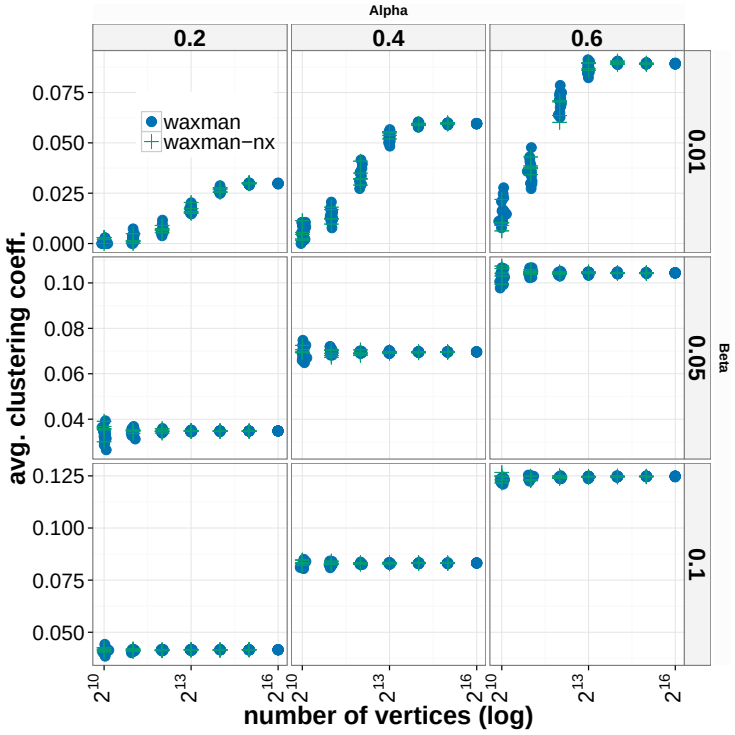


Fig. 44. Comparison of the average clustering coefficient for the Waxman models.

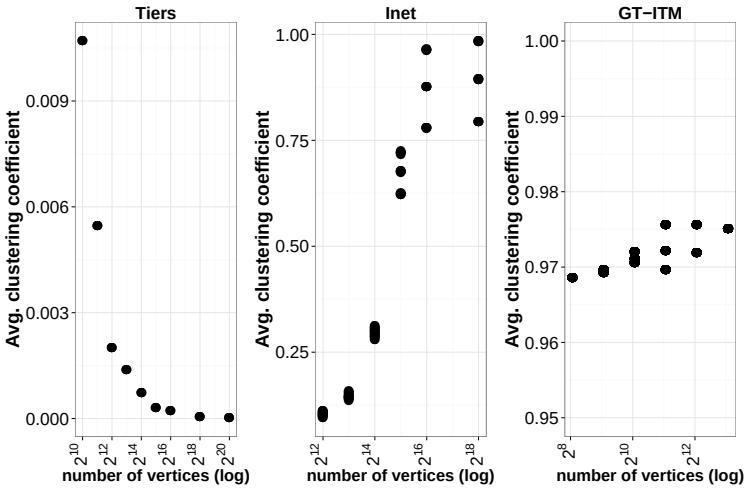


Fig. 45. Average clustering coefficient change for the Inet-3.0, Tiers and GT-ITM models.

Diameter

The variations in the diameters produced by the Waxman models depend on the input parameters. For low α and high β, the diameters vary significantly with graph size. Otherwise, the diameters are almost invariant across scale and the rest of the parameter space.
The Inet-3.0 model produces small world type graphs, meaning that the diameters are small. In fact, the diameters decrease ever so slightly as the sizes of the graphs increase. This is in line with previous observations about network diameters from Leskovec et al. The GT-ITM model also displays small world properties with constant diameter across scales. The Tiers model on the other hand has larger diameters that increase up to $|V| = 2^{14}$, then decrease. This follows from the observation regarding the number of edges produced, but does provide insight as to whether this is an accurate representation of real world networks.

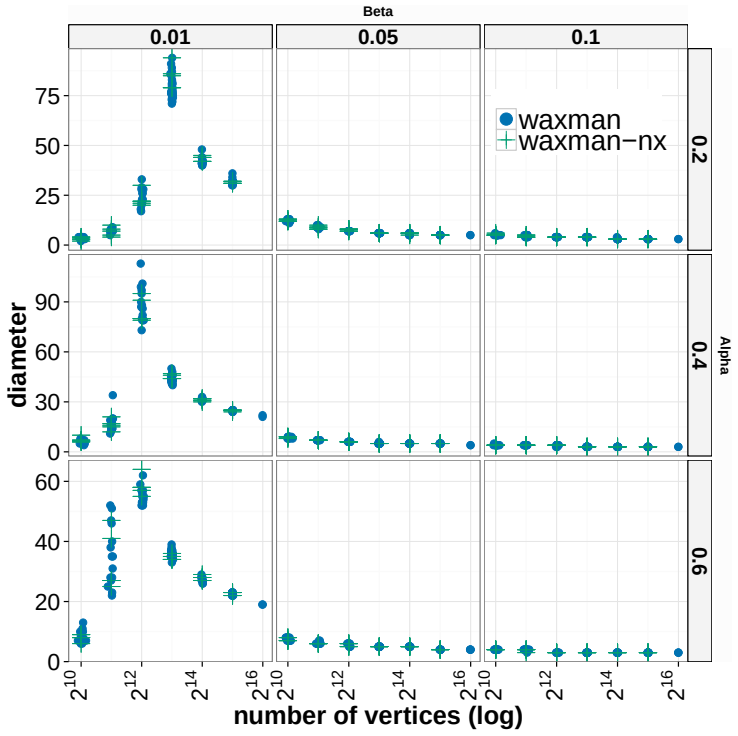


Fig. 46. Comparison of the diameters from the Waxman models.

Assortativity

Both Waxman models produce the same trends with respect to the assortativity. The α parameter has little effect on the values, while changes in β as well as the graph scale produce dissassortative and assortative networks. At constant β, the graph size has an impact on the assortativity. In general, larger graphs tend to be more assortative.
The Inet-3.0, Tiers and GT-ITM models produce very different values of this metric. In each case, the values are almost invariant across scale. Inet-3.0 and Tiers produce disassortative graphs, with the Tiers graphs becoming completely dissassorative at the highest tested graph sizes. The GT-ITM model, on the other hand, results in almost completely assortative networks. These values decline slightly as the scales

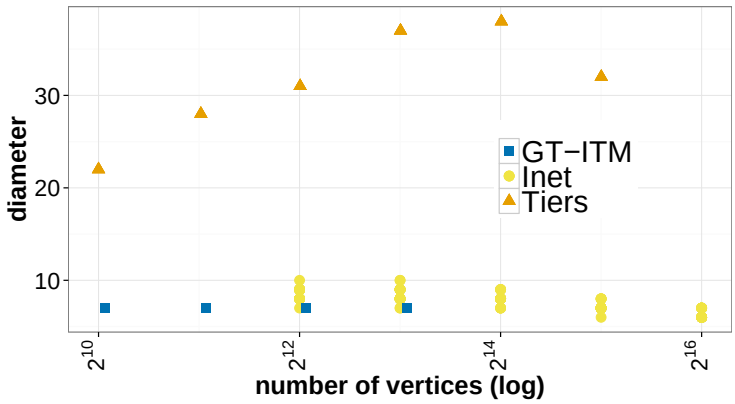


Fig. 47. Comparison of the diameters from Inet-3.0, Tiers and GT-ITM topology models.

increase, but still remain quite high. Given the general structure of Internet Topology with hubs and different levels of hierarchy, it would seem that the Inet-3.0 model, which is somewhat dissassortative, would perhaps be the closest to modeling real-world networks (i.e., reflecting the hubs-spokes connections).

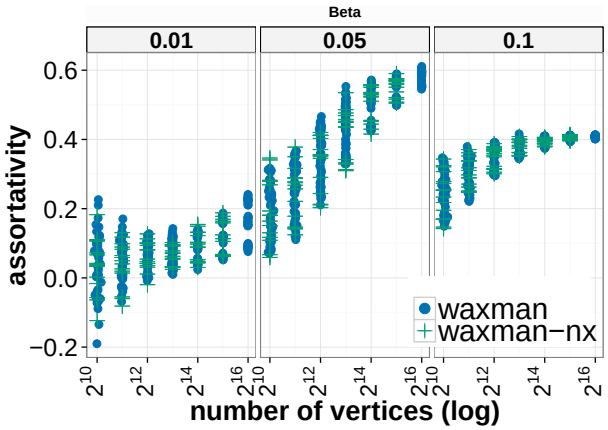


Fig. 48. Change in assortativity for Waxman graph models with increasing sizes.

Degeneracy

For this metric, the values obtained from the Waxman models vary with changes in the β parameter, but are not impacted by α. From Figure 50, it can be seen that as β increases, the degeneracy starts to increase rapidly and at smaller graph sizes. While the degeneracy appears low and almost invariant for $\beta = 0.01$, it is possible that the graph size at which this increase will be seen has simply not yet been reached.
The values computed using Inet-3.0 synthetic graphs have exponential growth as the graph sizes increase and stand out as being large overall. The Tiers model produces very small values which are scale invariant, while the GT-ITM model produces slightly larger values with small increases as the graph sizes increase.
In [38], the reported core sizes for the peer-to-peer networks *p2p-Gnutella25* and *p2p-Gnutella30* are 5 and

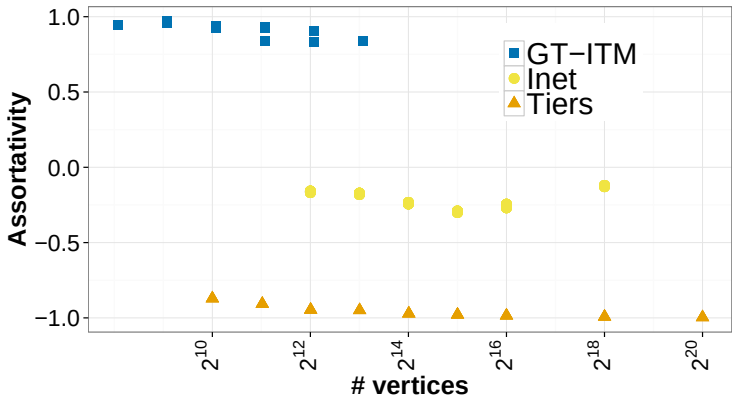


Fig. 49. Comparison of the assortativity of Inet-3.0, Tiers and GT-ITM topology models.

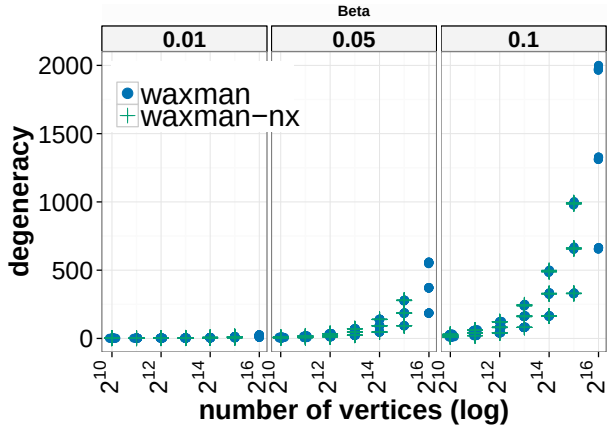


Fig. 50. Comparison of the degeneracy of graphs created using the Waxman models.

7 respectively. Alvarez-Hamlin et al. [4] report degeneracy values for the CAIDA (2005) and DIMES (2005) AS maps as 26 and 39 respectively (with graph sizes of $n = 8542$ and 20455).
While the Tiers model might be appropriate the for the p2p networks, the Waxman, Inet-3.0 ($d \sim 51$) and GT-ITM ($d \sim 18$) models produce values that are too high. On the other hand, for the AS maps reported in [4], the Inet-3.0 model and Waxman model (properly parameterized) are appropriate for the larger network.
In general, it does not appear that any of the models do a good job of creating synthetic graphs with appropriate (core) structure.

Degree distribution

In [36], the authors are fairly emphatic, stating that models that do not incorporate a power law representation of the degree distribution are "doomed," almost specifically placing the Tiers and Transit-stub models into this category, while promoting the Inet model (version 1.0). Output from the synthetic graphs in the current experiments indicate that only the Inet-3.0 model produces graphs with power law degree distributions $P(k) \sim k^c$ where $c \simeq 2$.

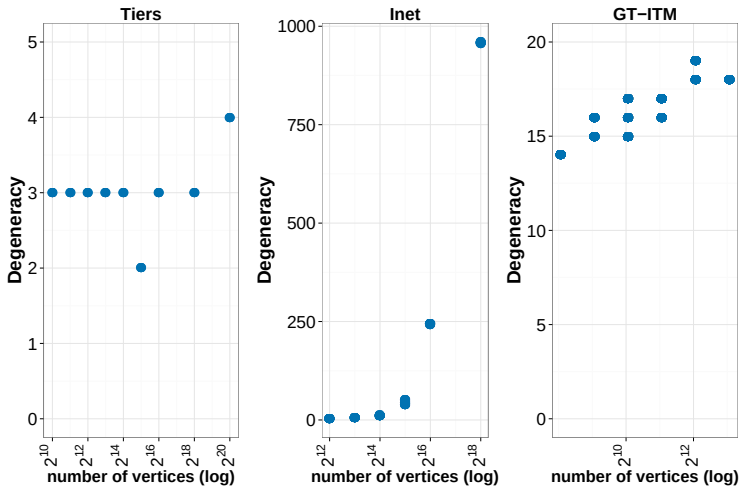


Fig. 51. Comparison of the degeneracy of Inet-3.0, Tiers and GT-ITM topology models.

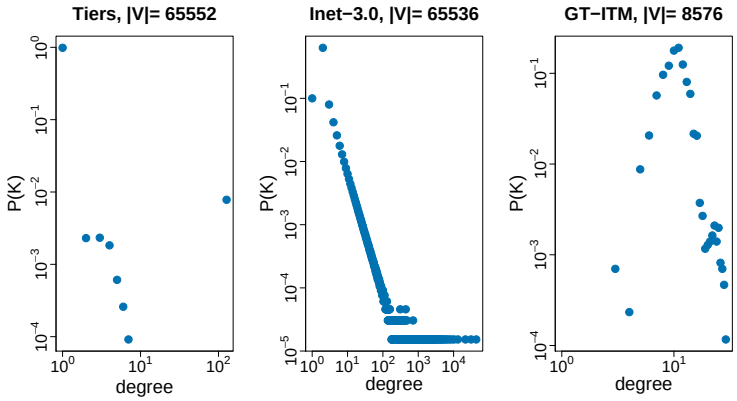


Fig. 52. Degree distributions from Inet-3.0, Tiers and GT-ITM topology models.

The synthetic Waxman graphs also do not follow power law degree distributions (Figure 53). At smaller sizes and low β values, parts of the output resemble this type of distribution, but the larger the graph and the higher the value of β, the distributions look normally distributed with no tails. The "bell" also becomes more compressed at these higher parameter levels and sizes.

Centrality

The distribution of centrality values for the synthetically generated Waxman graphs have similar patterns at increasing sizes. The observed variations are due to the changes in input parameters. The α value shows no significant impact, while increases in β shift the bell of the curve to the right towards higher centrality values. In Figure 54, the centrality axis has been truncated in order to show more details. One should note, however that only the graphs having $\beta = 0.01$ as input exceed the maximally displayed value. These graphs have a few nodes with values around 0.05 to 0.06.

The Inet-3.0 model distributions contain a high percentage of 0-values (Figure 55a). The remaining values are also very close to 0, with a few higher values between 0.2 - 0.5. This indicates the presence of hub

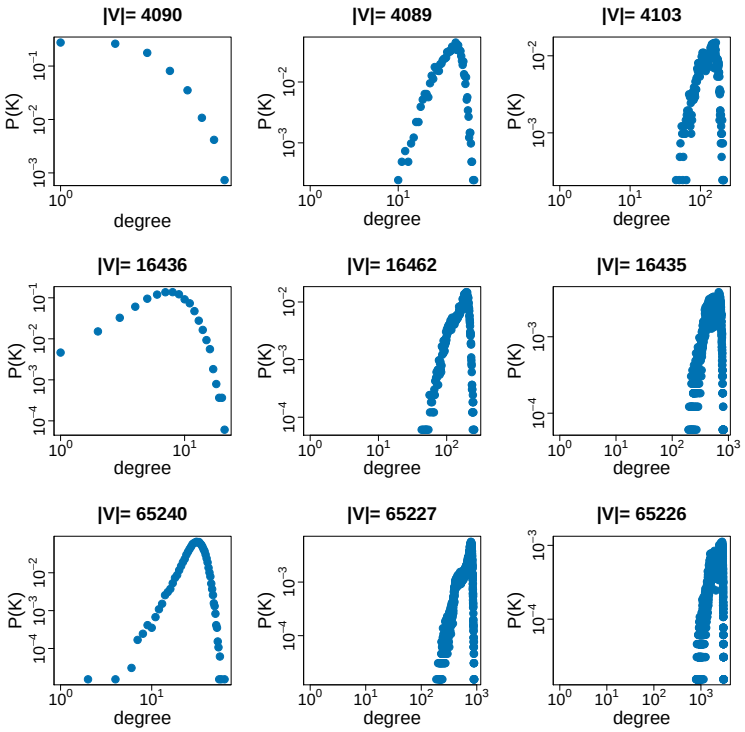


Fig. 53. Degree distributions from the Waxman model at various scales.

vertices.

The Tiers model generates distributions where greater than 98% of the values are zero. Given the limitations of the model specified previously, this result is intuitive since nodes are connected to a central hub. The maximal centrality values are about 0.6.

The GT-ITM model implements a transit-stub approach resulting in close to 90% of the nodes having zero centrality. The remaining "transit" nodes are more central with the maximum centrality values around 0.07.

Core

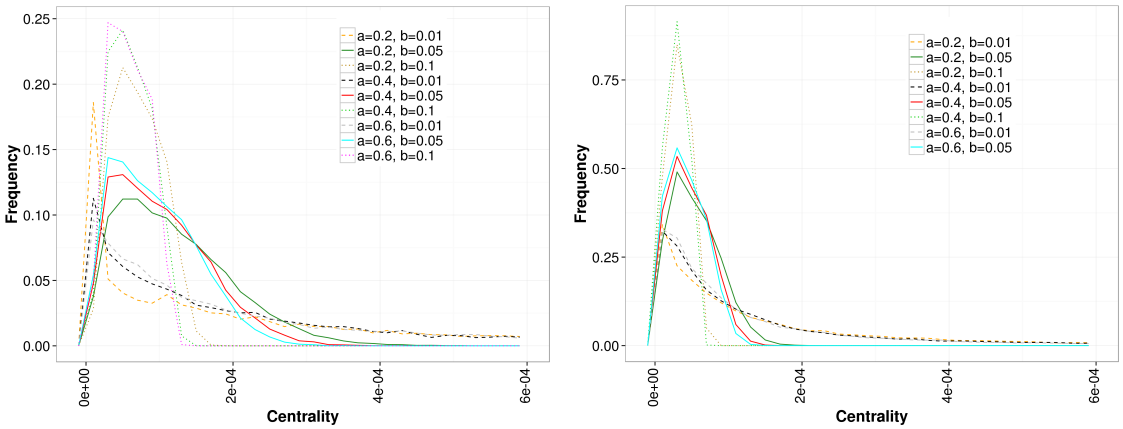
The distributions of k-core values for the synthetically generated Waxman graphs have similar patterns at different sizes, but the pattern is shifted with higher core numbers (Figure 56). As seen in the degeneracy analysis, there are higher values of k for larger graphs. The percentage of total nodes in the largest core is higher at larger graph sizes resulting in the cores being smaller.

The α parameter has little effect on the distribution of k-cores. An increase in this parameter slightly increases the size of the cores and quantity.

An increase in β greatly increases the size and number of cores. However, the largest core contains the same percentage of vertices, indicating the others are simply more "smeared" across the remaining core values, since the number of nodes is constant.

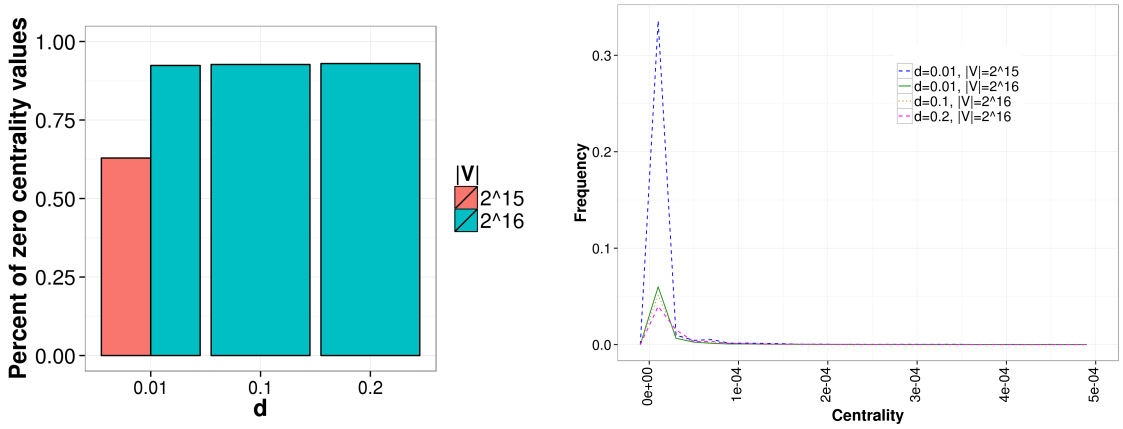
The Inet-3.0 model displays a shift from 2-core to 1-core as the value of the parameter d increases. The maximum core is about 245 for each of the d values tested. The tails of the distributions have been omitted in Figure 57 as they are almost identical.

Due to code limitations, the synthetic graphs using the Tiers model were generated with only 1 parameter



(a) $|V| = 2^{14}$ **(b)** $|V| = 2^{15}$

Fig. 54. Distribution of the centrality values for the Waxman model.



(a) Proportion of 0-values **(b) Centrality distribution**

Fig. 55. Distributions of the centrality values for the Inet-3.0 model.

combination for each graph size. Even at $|V| = 2^{20}$, there are only four distinct core sizes. Each power of two increase in graph size adds a single additional core value, while the majority of vertices (>99.9 %) still belong to the 1-core.

The largest graphs obtained using the GT-ITM model were of size $|V| = 8576$, despite the requested input. The distribution of k-core values follows a "bell-shape".

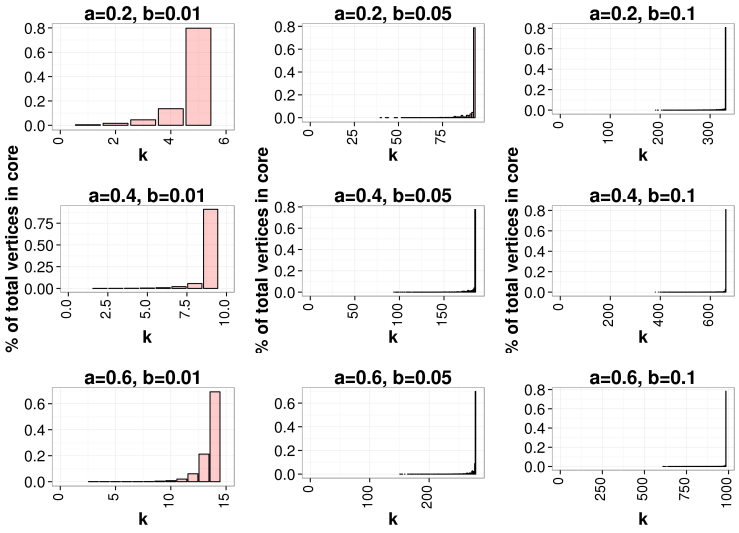


Fig. 56. k**-core distributions from the Waxman model at various parameter settings,** $|V| = 2^{15}$**.**

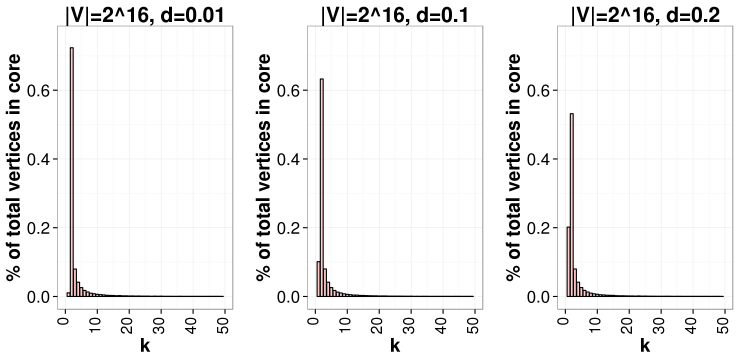


Fig. 57. k**-core distributions from the Inet-3.0 model at various parameter settings,** $|V| = 2^{15}$**.**

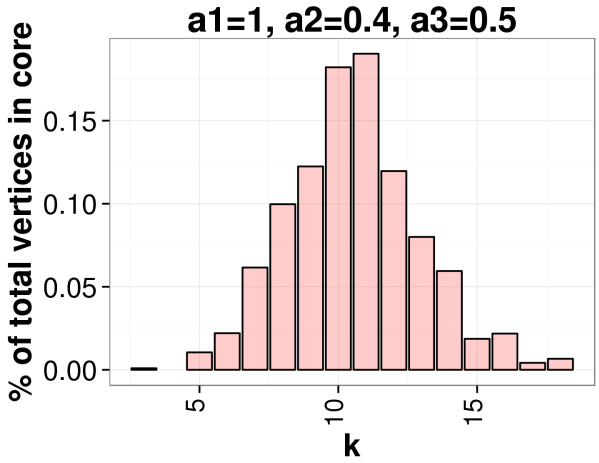


Fig. 58. Distribution of k-core values from the GT-ITM model, $|V|_{input} = 2^{15}$, $|V|_{actual} = 8576$.

2.7 BTER

Seshadhri et al. [24, 37] introduced the Block Two-Level Ërdos-Rényi model (BTER) as a scalable alternative to previous models, with an emphasis on the ability to match "real-world" community structure. This is a two step model requiring an input degree distribution sequence which is used to form groups with uniform vertex degrees in the pre-processing phase. In the first phase, small independent communities are generated following an Ërdos-Rényi model. Edges are created within each community with probability p, a function of the smallest degree in the community. In the second phase, the communities are connected using a Chung-Lu model.

Connected components

In the two-phase model set-up, the first phase results in many separate groups which are then connected in the second phase. The authors note that isolated vertices may be created in the process and thus provide a mechanism for manually connecting a certain portion of them.
Investigating the resulting number of disconnected components in the graph indicates that as the graph sizes get larger, there are more and more disjoint elements. In Figure 59, we consider the size of the largest singly connected component relative to the overall graph. As the graph sizes increase, this ratio tends toward one indicating that the graph as a whole is more fully connected. Smaller graphs though tend to be composed of many smaller subgraphs.

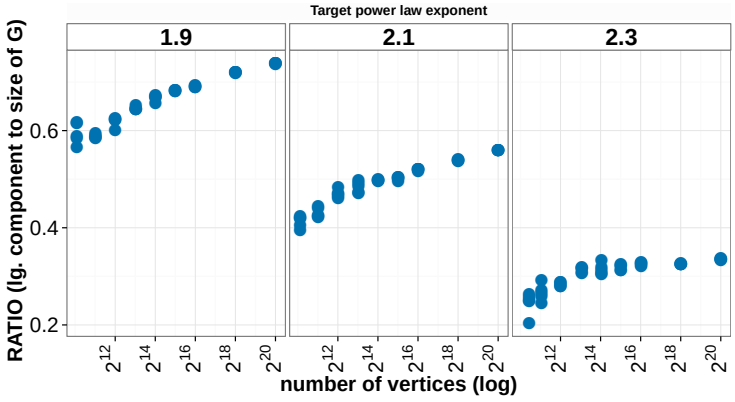


Fig. 59. Change in the ratio of the size of the largest singly connected component to the overall graph size for graphs generated using the BTER model.

Average degree

The model requires a degree distribution as input which is used to create the initial degrees of nodes and communities and replicate the input degree distribution. This directly influences the number of edges created and hence the average degree of the graph. In the created synthetic graphs, as the scales are increased, under the same power law exponent, the number of edges in the graph increases, thus the observed increase in the average degree. Replicating a real dataset's average degree will depend on choosing an appropriate input degree distribution.

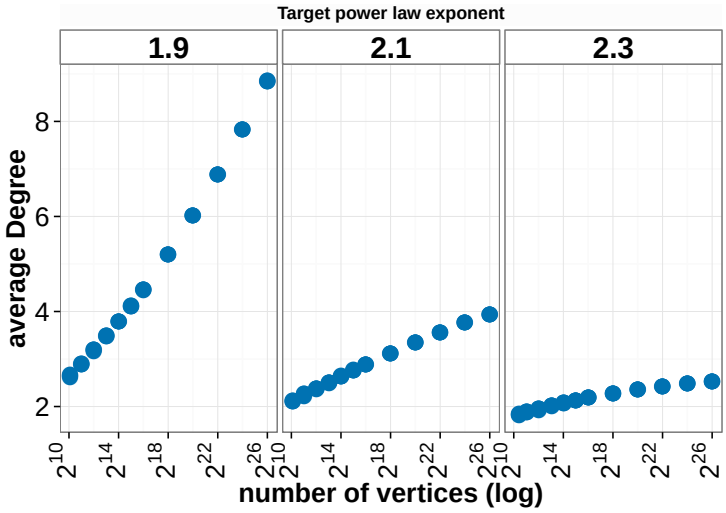


Fig. 60. Average degrees from synthetic BTER graphs at different scales.

Average clustering coefficient

In the design of the model, the authors criticize previous models for creating graphs with low clustering. Particularly, they note the tendency of vertices with low degree to have high average clustering coefficients. In [37], the synthetically generated BTER graphs do indeed provide a good graphical match to the tested datasets, though the authors do not provide actual numerical values. This is perhaps due to the fact that the averaging could mask variations between nodes with high and low degree and undercut the value-add of the model.

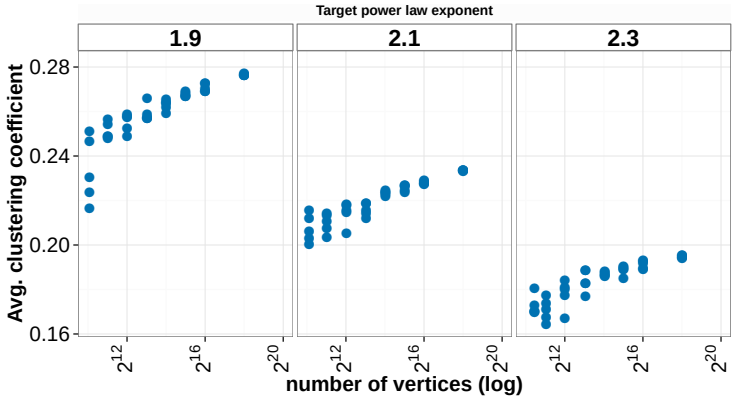


Fig. 61. Average clustering coefficient of synthetic BTER graphs at different scales.

The 4 graphs tested in the authors' initial paper (data available at `http://snap.stanford.edu/data/`): *ca-AstroPh, ca-ContMat, cit-HepPh, soc-Epinions1* have average clustering coefficients of 0.6306, 0.6334, 0.2848 and 0.1378 respectively. Note that especially for the lower values, several of the other models introduced produce comparable values, though we cannot say anything about individual vertex values. The values obtained in our experiments do not actually fit these data points. This is most likely due to the

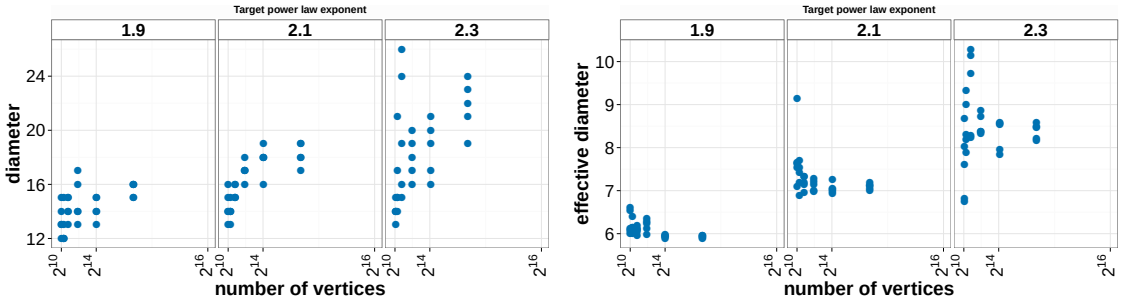
input degree distribution used or the parameters for the edge probabilities within each community (left up to the model user to select). As evidenced by the results in Table 4 and the original paper, the BTER model can produce comparable results for the average clustering coefficient with correct parameterization. The tabulated results were obtained using the parameters proposed by BTER's authors.

Table 4. Comparison of BTER output using *cit-HepPh* as a baseline.

	cit-HepPh	BTER	pure Chung-Lu
Average degree	24.36	25.19	25.49
Avg. clustering coefficient	0.2847	0.268	0.00475
Diameter	12	11	8
Effective diameter	4.986	4.332	3.80
Assortativity	-0.00628	0.2738	-0.002065
Degeneracy	30	83	23
Avg. path length	4.326	3.727	3.776

Diameter

In the original paper describing the model [37], 4 datasets are examined which have diameters 14, 14, 12, 14 respectively. With proper parameter selection, the BTER model appears to be able to replicate these points.



(a) diameter **(b) effective diameter**

Fig. 62. Comparison of diameter behavior for synthetic BTER graphs at different scales.

As pointed out in a study on dataset diameters, Leskovec et al. [28] note the shrinking diameter effect over time in real networks. While there is no time component to BTER, the diameters do appear to increase slightly with size. It has also been pointed out that the diameter can be highly susceptible to outlying points. As seen in Figure 62b, the effective diameter values are mostly invariant with scale.

Assortativity

The model authors are clear in their initial description [37] that the current design will not adequately represent highly dissassortative datasets. Based on the simulation experiments, the synthetic graphs all have positive assortativity with moderately increasing values as the graph sizes increase.

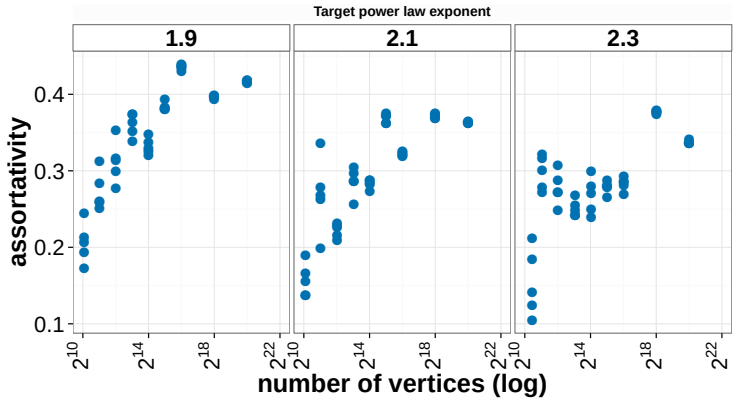


Fig. 63. Assortativity values from synthetic BTER graphs at different scales.

In Table 4, the BTER model produces positive assortativity values despite the initial graph being slightly dissassortative. As noted by the authors, this is a current shortcoming of the model.

Degeneracy

The degeneracy values obtained through simulation increase at larger graph sizes. Using the *cit-HepPh* data as a comparison point, the obtained values (Table 4) indicate that the BTER model produces values that are inconsistent with the true dataset. While the BTER authors indicate that values obtained via the Kronecker method are too low, the values obtained here are too high. This may indicate that the produced subgraphs are actually too dense, though the obtained clustering coefficients would negate this.

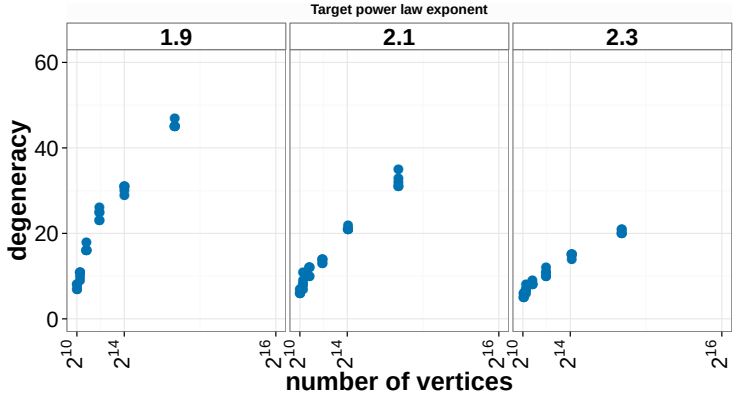


Fig. 64. Degeneracy of synthetic BTER graphs at different scales.

Degree distribution

The BTER model requires a distribution as input to the pre-processing stage. In our experiments, we used the default power law distribution. The results shown in Figure 67 use an exponent value of $\gamma = 1.9$ which is reflected in the output. The model can take a variety of distributions as input.

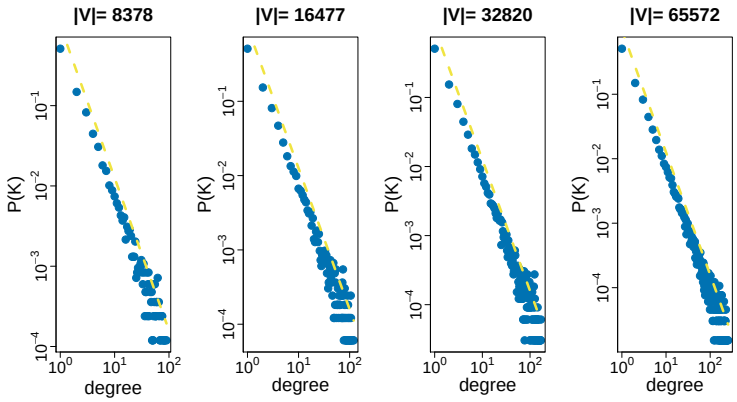
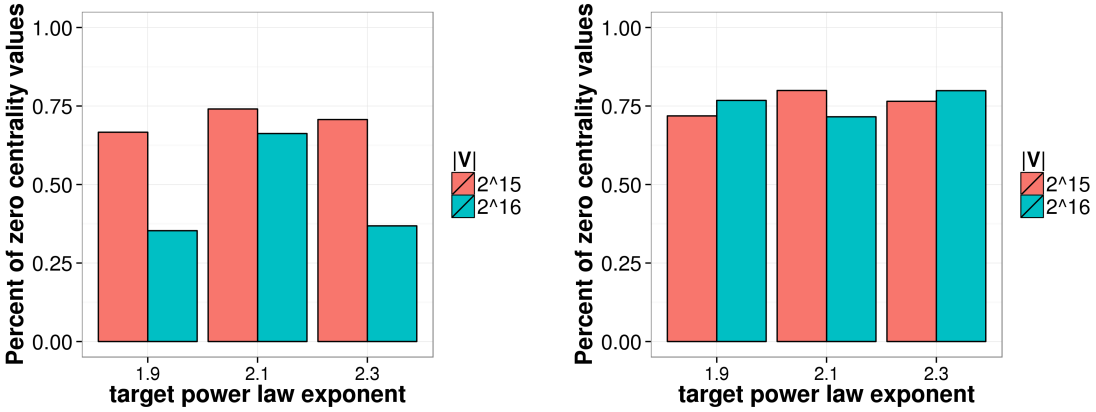


Fig. 65. Degree distribution of synthetic BTER graphs at different scales.

Centrality

The overall frequency of zero centrality scores is comparable as graph sizes increase and across target power law exponent values. If the disconnected nodes are removed, this percentages shows a significant decrease as graph sizes increase. The increase in isolated vertices is visible through this analysis (Figure 66).

Removing the zero-values reveals similar distribution trends. As the target exponent is increased, the slope of the curve increases forcing the peak upwards with a greater number of values closer to 0. The maximum values are similar ranging from 0.0046 to 0.0086.



(a) connected vertices **(b) connected and isolated vertices**

Fig. 66. Percentage of zero centrality values for graphs generated using the BTER model.

Core

An increase in graph size results in larger maximum k-core values, otherwise the shape of the distribution remains the same with a slightly more elongated tail.

The γ parameter corresponds to the target power law exponent. An increase in γ reduces the maximum

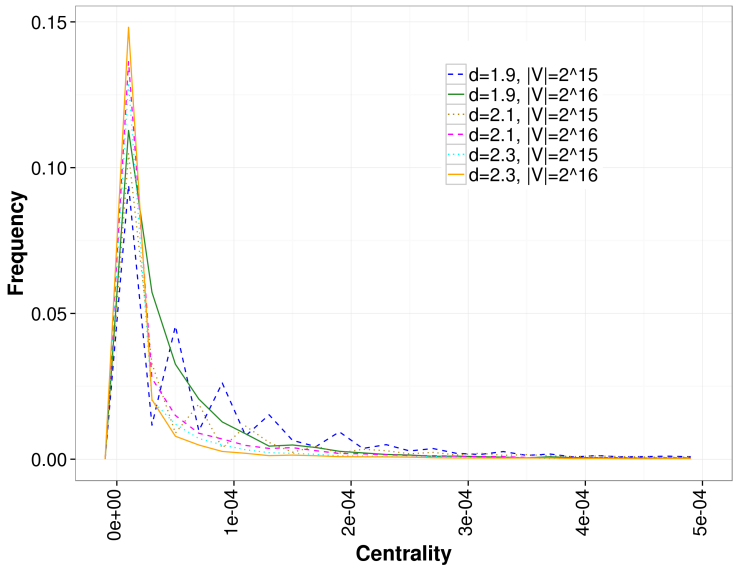


Fig. 67. Centrality distributions of synthetic BTER graphs at different scales.

k-core and shrinks the distribution tail, thereby forcing the same number of nodes into fewer bins. This can be seen in the increase in the percentage of nodes present in the 1-core. Since the BTER generated graphs have disconnected components, there are also vertices which belong to the 0-core.

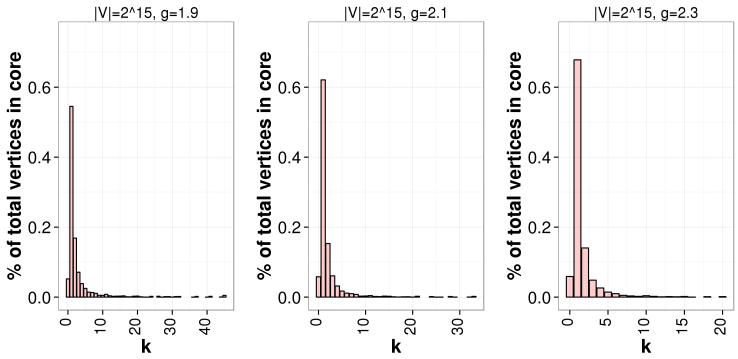


Fig. 68. k-core distributions from the BTER model at various parameter settings, $|V| = 2^{15}$.

Observations

For this model, the input requirements do present a downside. While the authors contend that this model is easy to fit to real-world data and replicates it well, the connectivity parameters in phase 2 require some fitting by trial and error. For many scientists, this is not ideal and may be a drawback to this model. Additionally, while the authors demonstrate the ability of this model to replicate features previously overlooked in other generators, some metrics still vary with respect to reality such as the degeneracy and the assortativity.

2.8 RDPG

The Random Dot Product Graph (RDPG) model was proposed by Young and Scheinerman. We found only one implementation as part of the 'mfr' package in R. Unfortunately, since the initial implementation, it has been removed due to non-maintenance by the authors. Additionally, the implementation was not very scalable, thus few graphs were able to be generated. Our initial results display some rather odd behavior that we can no longer test. They are presented here for completeness, however we note that a new implementation should be undertaken and tested. We are not the only ones to make mention of this, as the authors in [37] also point out that the model is scalable, but comparisons with real world graphs have yet to be studied. We propose this as a future avenue of research.

The model operates by creating a vector v for each vertex from $\mathbb{R}^d$ where d is the space dimension. An edge exists between vertices i and j if the dot product $v_i.v_j > p$ for some probability p. While not explicitly documented, it appears that p is set to 1 in the mfr package. We chose to operate in the 2 dimensional plane with vector elements selected i.i.d from a Uniform(0,1) distribution.

Connected components

Under the current parameter settings, all graph were full connected.

Average degree

The obtained synthetic graphs have very high average degrees which increase linearly and rapidly with scale. This behavior is not close to real world datasets.

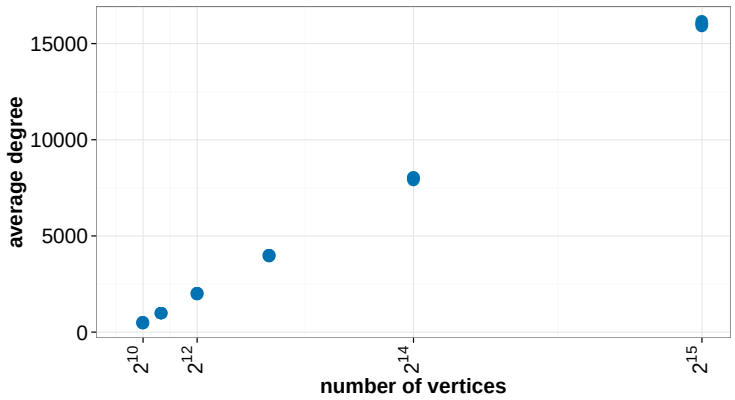


Fig. 69. Average degree of synthetic RDPG graphs at different scales.

Average clustering coefficient

The average clustering coefficients obtained via the RDPG model are fairly high, but invariant of scale. Young and Scheinerman [44] provide a proof indicating positive clustering, which is observed in the test data output.

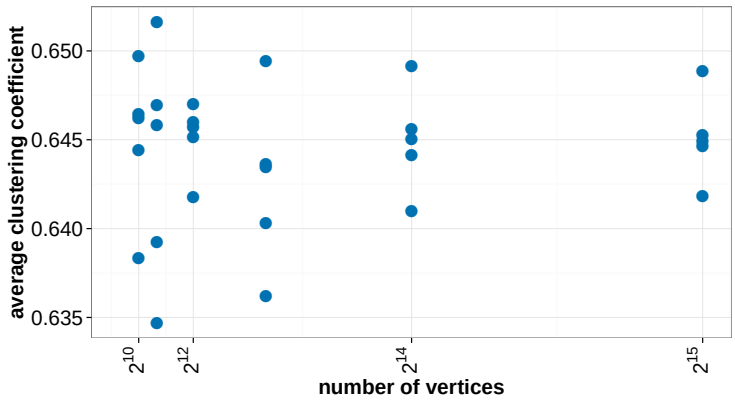


Fig. 70. Average clustering coefficient of synthetic RDPG graphs at different scales.

Diameter

Young and Scheinerman [44] highlight that for the simplifying case of a $d = 1$ model, Kraetzl et al. showed that the diameter of the giant component is no more than 6 as $n \to \infty$. The first set of authors further proved asymptotic constant diameter. For the synthetically generated graphs, all diameters are 3, which is in line with this finding given that the graph sizes are all less than $|V| = 2^{15}$.

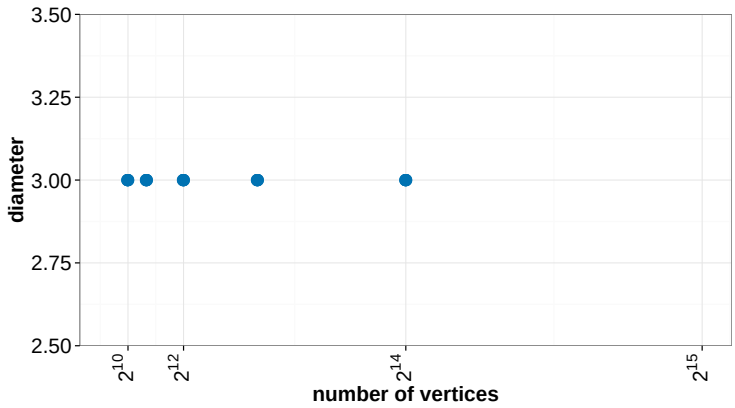


Fig. 71. Diameter of synthetic RDPG graphs at different scales.

Assortativity

In his dissertation [43], Young proves that the model has slight positive assortativity tending toward 0 as $n \to \infty$. The synthetically generated graphs, however, are slightly dissassortative, with a convergence at increasing scales around -0.0325. Given that we cannot further test the generator, it is difficult to say whether the implementation is incorrect or these properties would become more apparent at higher scales.

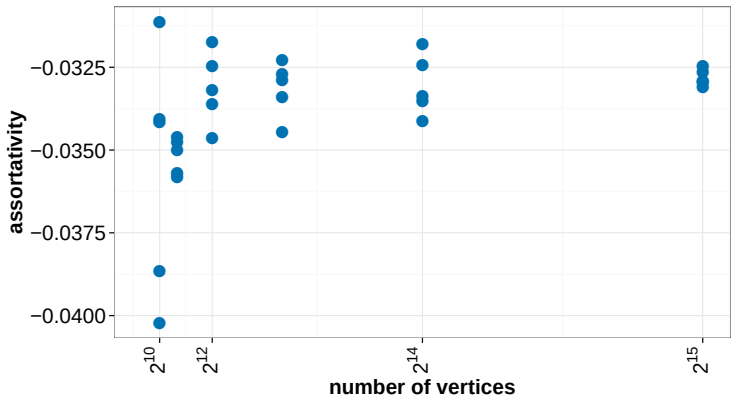


Fig. 72. Assortativity of synthetic RDPG graphs at different scales.

Degeneracy

The degeneracy trends resemble the average degrees, increasing rapidly with high values. While other models produce values that are too low, the results here seem abnormally high especially compared to documented real-world data, which do not generally exceed a maximum k-core of more than approximately 250.

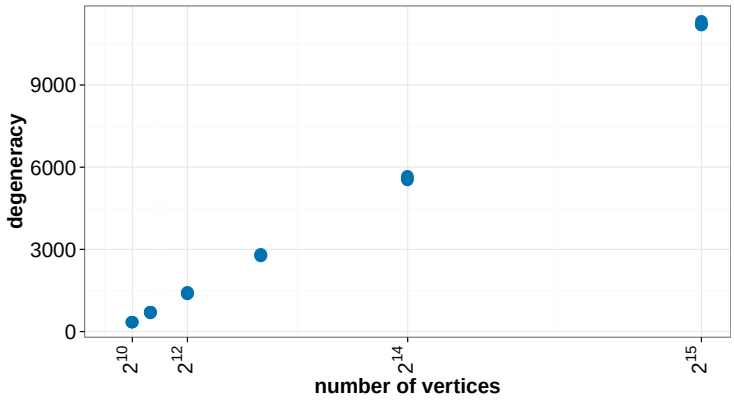


Fig. 73. Degeneracy of synthetic RDPG graphs at different scales.

Degree distribution

The graphs generated via this method contain a higher number of above average degree vertices than other models. This results in degree distributions with a positive slope, almost the mirror image of a power law distribution.

Centrality

The graphs generated using the RDPG model tend to be quite dense, resulting in extensive computation time for the centrality. The distribution (Figure 75) shows a propensity for the frequency of the centrality

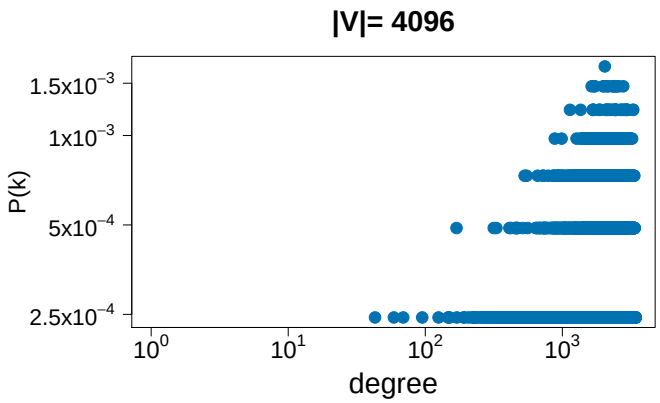


Fig. 74. Degree distribution of a synthetic RDPG graph.

values to decrease as the values increase. Close to 80% of the data points have a values less than 10^{-4}, which is half of the maximal value.

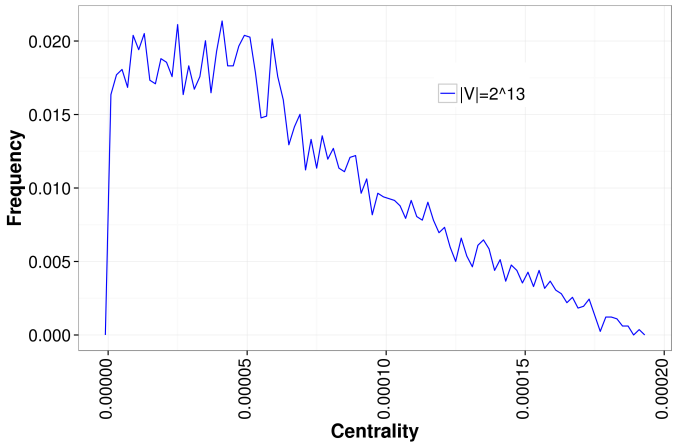


Fig. 75. Centrality distribution of a synthetic RDPG graph.

Core

Using the RDPG model, an increase in graph size produces an increase in the maximum k-core. The overall distribution of the cores remains the same at different scales with an elongated tail. A strong peak at the 2-core value persists regardless of graph size.

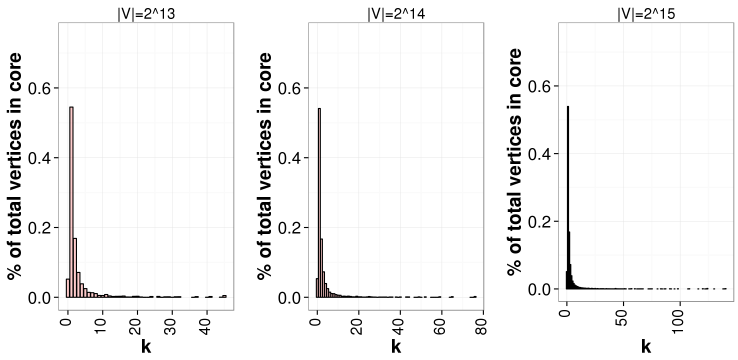


Fig. 76. *k*-**core distributions from the RDPG model at various parameter settings,** $|V| = 2^{15}$**.**

3. PROPERTIES OF "REAL" GRAPHS

3.1 GENERAL MODEL PROPERTIES

Each of the models analyzed in the previous section makes claims about how well it mimics different properties of real-world networks. Several models have analytical formulations for some of the studied metrics. Of the metrics surveyed, we highlight three that are most discussed in the graph literature, specifically clustering, graph diameter and degree distribution.
Table 5 summarizes the existing analytical formations, while Table 6 indicates whether these formulas continue to be true at scale. In general, the models related to Internet structure do not make many claims with respect to traditional graph metrics. Instead, the focus is on the structure itself or the ability to replicate the hierarchical nature of the Internet.

Table 5. Model property claims.

Model	$\bar{d}$	Avg. C(k)	D	Assort.	Deg.dist.
Ërdos-Rényi	pn	p	$\frac{\log n}{\log\langle k\rangle}$	0	Poisson
Small World	*	3/4	Eq. 2	0	Eq. 3
Pref. Attach.	$\frac{2\|E\|}{n}$	Eq. 4	Eq. 5	0	power law
Hyperbolic	*	$> 0^{\#}$	open	open	power law
R-MAT	*	open	open	open	$N(\mu,\sigma)$ mix
Kronecker		open	small	open	power law
Inet	-	-	-	-	power law
Tiers	-	-	-	-	-
Transit-stub	-	-	-	-	-
Waxman					
BTER	*	"realistic"	small	needs work	heavy tailed
RDPG	open	positive	< 6	0^{+}	open

* denotes a model input
# bounded away from 0

Table 7 provides a brief survey of known properties of real-world graphs. In addition to the properties mentioned, many networks also display an intricate community structure. This was a key focus in the development of the BTER model for example.

Table 6. Summary of graph model feature scalability.

Model	Avg. cc		Diameter		Deg. dist	
	Analytical formula	Match at scale	Analytical formula	Match at scale	Analytical formula	Match at scale
Ërdos-Rényi	✓	✓	✓	✓	✓	✓
Small world	✓	✓	~✓	✓	✓	✓
Pref. attach.	✓	✓	✓	✓	✓	✓
Waxman	✗	✗	~✓	✗	✓	✗
Inet-3.0	✗	✓	✗		✗	
Tiers	✗		✗	✗	✗	✗
Graph500	✗		✗	✓	✓	✗
pyweb	✗		✗	✓	✓	✗
Krioukov	✓	~✓	✗		~✗	✓
BTER	✗		✗	✗	~✓	✓
RDPG	✗		✗		✗	✗

Table 7. Real world network properties by network type (refer to [34] for numerical examples).

Network type	properties
Social network	high clustering, small-world, degree dist. is heavy-tailed/power law
Autonomous systems	high clustering, small diameter, power law degree dist.
Information networks	mid to high clustering, power law degree dist.
Biological networks	high and low clustering, power law degree dist.
Power grid	Exponential degree dist.

3.2 REAL WORLD NETWORKS

Many different types of networks exist, from true physical (hardware) networks, social (online) networks to geographical linkages (such as road networks). While one model may be well-suited to a given problem type, it may be a poor representation of another. In general, many existing networks are large (e.g., millions or billions of nodes) and continue to increase in size as data storage, analytics, and data science become increasingly popular. Several publicly available datasets were compared against synthetic graphs generated using the models presented in this report in order to test their ability to simulate real-world data. These include the following datasets: the *Marvel Universe* [‖], a word association dataset (*EATnew*) [**] and two yeast protein interaction networks, (*CampyYTH* [††] and *yeast-interactome* [‡‡]). Further details are given in Tables 8 and 9.

Table 8. Real world networks studied.

Dataset	$\|V\|$	$\|E\|$	Description
yeast-interactome	1278	1641	Protein interaction map of the yeast interactome network
CampyYTH	1332	11664	Protein interaction data from a large-scale yeast two-hybrid (YTH) screen
Marvel Universe	6486	168267	Marvel Comics character collaboration graph
EATnew	23219	304938	Edinburgh Associative Thesaurus (EAT) word association set

Table 9. Properties of the real world networks studied.

	edge density	Avg. degree	Assortativity	Avg. C(k)	Diameter	Eff. diameter	avg. path length
yeast-interactome	0.002	2.57	-0.12	0.045	14	7.09	5.36
CampyYTH	0.013	17.51	-0.26	0.095	6	3.48	2.91
Marvel Universe	0.0080	51.89	-0.016	0.77	5	2.87	2.64
EATnew	0.0011	26.27	-0.054	0.099	6	3.87	3.48

Using synthetic graphs of comparable size to the real datasets, the edge density, average degree, assortativity, average clustering and diameter were compared using the Root Mean Square Error (RMSE) statistic. The generator models were ranked based on their ability to replicate the true values. Tables 10, 11, 12 and 13 summarize the results. One can quickly see that no single model adequately represents all types of real data or even all graph properties. Models that perform well overall frequently have poor results for one or more of the statistics.

When selecting a generator model, it is therefore important to evaluate the type of data in question and the features that are most necessary to reproduce. During the model development process, authors tend to focus on replicating a subset of features leading to results such as those observed here. While the synthetic graphs were generated with a wide variety of parameter values, it is possible that certain model generators could fare better than the current results indicate if the parameters are tuned further. A few of the models however, seem to routinely have poor performance across the board, such as RDPG and Tiers. Subsequent

[‖] http://bioinfo.uib.es/ joemiro/marvel.html

[**] http://vlado.fmf.uni-lj.si/pub/networks/data/dic/eat/Eat.htm

[††] http://proteome.wayne.edu/CampyDescription.html, v.3.1

[‡‡] http://interactome.dfci.harvard.edu/S_cerevisiae/

work will investigate methods for selecting optimal parameters and models to best match and replicate real world behaviors. There are many additional domains than those covered by the real datasets investigated. Data from different domains (e.g., internet topology) might be best fit by generators that fare poorly for the cases selected here. Further investigations may reveal that parameter and model selection are also domain dependent.

Table 10. Ranks of best/worst generator matches to the *Marvel Universe* dataset.

	Edge density	Avg. degree	Assortativity	Avg. C(k)	Eff. diameter	Avg. path length	Rank
			top 3 (with tie)				
pyweb	5	5	3	4	3	3	3.83
Ërdos-Rényi	1	1	9	11	1	1	4
Krioukov (cold)	2	3	4	1	7	7	4
g500	4	4	1	6	5	5	4.167
			bottom 3 (with tie)				
Small-world	9	8	5	7	11	11	8.5
RDPG	12	12	7	2	9	9	8.5
BTER	10	9	10	3	10	10	8.67
Tiers	11	10	12	12	12	12	11.5

Table 11. Ranks of best/worst generator matches to the *CampyYTH* dataset.

	Edge density	Avg. degree	Assortativity	Avg. C(k)	Eff. diameter	Avg. path length	Rank
			top 3 (with tie)				
g500	1	4	2	5	2	2	2.67
Waxman	4	1	9	4	1	1	3.33
Krioukov (hot)	2	5	5	3	3	3	3.5
Pref. attachment	3	2	6	2	4	4	3.5
			bottom 3				
BTER	9	9	10	8	9	9	9
RDPG	11	11	7	11	8	8	9.33
Tiers	10	10	11	6	11	11	9.83

Table 12. Ranks of generator matches to the *yeast-interactome* dataset.

	Edge density	Avg. degree	Assortativity	Avg. C(k)	Eff. diameter	Avg. path length	Rank
			top 3				
Pref. attachment	5	5	1	2	5	5	3.83
Ërdos-Rényi	4	3	8	4	3	2	4
Small-world	6	6	2	6	1	3	4
Waxman	8	8	9	1	2	1	4.83
BTER	2	1	10	9	4	4	5
Krioukov (hot)	7	7	3	5	6	6	5.67
Krioukov (cold)	3	4	7	7	7	7	5.83
Tiers	1	2	11	3	11	11	6.5
			bottom 3				
g500	10	10	4	8	8	8	8
pyweb	9	9	6	10	9	9	8.67
RDPG	11	11	5	11	10	10	9.67

Table 13. Ranks of generator matches to the *EATnew* word association dataset.

	Edge density	Avg. degree	Assortativity	Avg. C(k)	Eff. diameter	Avg. path length	Rank
			top 3				
g500	2	3	5	1	3	3	2.83
Ërdos-Rényi	1	4	4	7	2	2	3.33
Pref. attachment	5	5	3	5	1	1	3.33
			bottom 3				
Waxman	11	11	10	4	5	6	7.83
BTER	8	8	9	9	9	8	8.5
Tiers	10	10	11	8	11	11	10.17

4. SUMMARY

In this report, synthetic graphs from a variety of models were analyzed using some of the most common graph metrics. As pointed out by many other authors, most models stem from the desire to match one or two network properties, resulting in many different models, but none adequately mimic all properties even for a single network type.
A large importance has been placed on replicating the degree distribution with models such as Krioukov (hyperbolic), BTER and the Kronecker model providing good options. The downside to these models is that they tend to require some parameter matching via trial and error. Clustering and graph diameter also appear currently to be popular metrics to study and reproduce. Several of the more "standard" models such as small world and preferential attachment do a good job matching these parameters and should not necessarily be rejected in favor of newer options.

5. FUTURE WORK

This work began as an attempt to provide a comprehensive overview of the abilities of existing graph generator packages to scale and create realistic synthetic graphs.
Many of the models scaled well "out of the box" such as BTER, Krioukov (hyperbolic) and the Graph500 Kronecker implementation. Additional improvements could be made by porting the models to more optimized code bases and/or parallelization. Efforts are currently underway in both of these areas. We are also considering further investigations of graphs at larger scales than than those presented here; this will require additional enhancements to INDDGO, or the use of a different, more scalable approach to the analysis. Several of the Internet models (i.e., Tiers, Transit-stub) and the RDPG model did not scale well. Seshadhri et al. [37] note that the RDPG model is scalable, but that no one has compared the results to real graphs. This presents an area for future work, especially since the existing open-source RDPG implementation is no longer maintained. With respect to the Internet models, new models are needed or one should pursue existing alternatives such as Inet-3.0. Caution must exercised however, since it does claim to incorporate the hierarchical structure as do others.
In many cases, one of the primary purposes of synthetic graph creation is to mimic reality. While many of the models studied do match up well compared to real data, they often require parameterization. The BTER authors, for example, note that methods could be developed to automatically select 2 of the primary inputs to their model (ρ_{init} and ρ_{decay}). A auto-selection technique exists for the Kronecker model, but the scalability remains to be seen.
This study has shown that many good graph generator models exist. Unfortunately however, there is no one model that can adequately match all graph (or network) features nor replicate any and all data sets.

ACKNOWLEDGMENTS

This work was supported by the United States Department of Defense and used resources of the Extreme Scale Systems Center at Oak Ridge National Laboratory.

This research was also supported by an allocation of advanced computing resources provided by the National Science Foundation. Computations were performed on Kraken and Nautilus at the National Institute for Computational Sciences (http://www.nics.tennessee.edu/).

INDDGO development was supported by the DARPA GRAPHS program and the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory (ORNL).

ORNL is managed by UT-Battelle, LLC for the U. S. Department of Energy under Contract No. DE-AC05-00OR22725.

We thank Blair Sullivan for allowing us to build on top of the initial INDDGO framework and for making it publicly available.

6. REFERENCES

[1] Brief introduction | graph 500, 2012.

[2] Center for applied internet data analysis, 2014.

[3] Albert, R., and Barabási, A.-L. Statistical mechanics of complex networks. *Rev. Mod. Phys. 74* (Jan 2002), 47–97.

[4] Alvarez-Hamelin, J. I., Dall'Asta, L., Barrat, A., and Vespignani, A. K-core decomposition of internet graphs: hierarchies, self-similarity and measurement biases. *NHM 3*, 2 (2008), 371–393.

[5] Barabasi, A.-L., and Albert, R. Emergence of scaling in random networks. *arXiv:cond-mat/9910332* (Oct. 1999). Science 286, 509 (1999).

[6] Bode, M., Fountoulakis, N., and MÃijller, T. On the giant component of random hyperbolic graphs. In *The Seventh European Conference on Combinatorics, Graph Theory and Applications*, J. NeÅąetÅŹil and M. Pellegrini, Eds., vol. 16 of *CRM Series*. Scuola Normale Superiore, 2013, pp. 425–429.

[7] Bollobás, B. *Random Graphs*. Cambridge University Press, 2001.

[8] Bollobás, B., and Riordan, O. The diameter of a scale-free random graph. *Combinatorica 24*, 1 (Jan. 2004), 5–34.

[9] Candellero, E., and Fountoulakes, N. Clustering in random geometric graphs on the hyperbolic plane. *pre-print*.

[10] Chakrabarti, D., Zhan, Y., and Faloutsos, C. R-MAT: a recursive model for graph mining. In *In SDM* (2004), pp. 442–446.

[11] Chung, F., and Lu, L. The diameter of sparse random graphs. *Advances in Applied Mathematics 26*, 4 (2001), 257 – 279.

[12] Ciglan, M., Laclavík, M., and Nørvåg, K. On community detection in real-world networks and the importance of degree assortativity. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2013), KDD '13, ACM, pp. 1007–1015.

[13] Cohen, R., and Havlin, S. Scale-free networks are ultrasmall. *Phys. Rev. Lett. 90* (Feb 2003), 058701.

[14] Doar, M. A better model for generating test networks. In *Global Telecommunications Conference, 1996. GLOBECOM '96. 'Communications: The Key to Global Prosperity* (1996), pp. 86–93.

[15] Eppstein, D., LÃűffler, M., and Strash, D. Listing all maximal cliques in sparse graphs in near-optimal time. In *Algorithms and Computation*, O. Cheong, K.-Y. Chwa, and K. Park, Eds., vol. 6506 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 403–414.

[16] Erdős, P., and Rényi, A. On the evolution of random graphs. In *PUBLICATION OF THE MATHEMATICAL INSTITUTE OF THE HUNGARIAN ACADEMY OF SCIENCES* (1960), p. 17–61.

[17] Fountoulakis, N. On the evolution of random graphs on spaces of negative curvature. *CoRR abs/1205.2923* (2012).

[18] Fronczak, A., Fronczak, P., and Holyst, J. A. Mean-field theory for clustering coefficients in barabási-albert networks. *Phys. Rev. E 68* (2003), 046126.

[19] Gilbert, E. N. Random Plane Networks. *Journal of the Society for Industrial and Applied Mathematics 9*, 4 (Dec. 1961), 533–543.

[20] Groër, C., Sullivan, B. D., and Poole, S. A mathematical analysis of the r-MAT random graph generator. *Networks 58*, 3 (2011), 159–170.

[21] Gugelmann, L., Panagiotou, K., and Peter, U. Random hyperbolic graphs: Degree sequence and clustering. In *Automata, Languages, and Programming*, A. Czumaj, K. Mehlhorn, A. Pitts, and R. Wattenhofer, Eds., vol. 7392 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 573–585.

[22] Jensen, T. R., and Toft, B. *Graph Coloring Problem*, vol. 39. John Wiley & Sons, Inc., 2011.

[23] Klemm, K., and Eguíluz, V. M. Growing scale-free networks with small-world behavior. *Phys. Rev. E 65* (May 2002), 057102.

[24] Kolda, T. G., Pinar, A., Plantenga, T., and Seshadhri, C. A scalable generative graph model with community structure. *arXiv:1302.6636* (Feb. 2013).

[25] Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A., and Boguñá, M. Hyperbolic geometry of complex networks. *Physical Review E 82*, 3 (Sept. 2010), 036106.

[26] Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C., and Ghahramani, Z. Kronecker graphs: An approach to modeling networks. *J. Mach. Learn. Res. 11* (Mar. 2010), 985–1042.

[27] Leskovec, J., Kleinberg, J., and Faloutsos, C. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining* (2005), p. 177–187.

[28] Leskovec, J., Kleinberg, J., and Faloutsos, C. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data 1*, 1 (Mar. 2007).

[29] Li, C., Wang, H., de Haan, W., Stam, C. J., and Van Mieghem, P. The correlation of metrics in complex networks with applications in functional brain networks. *Journal of Statistical Mechanics: Theory and Experiment 2011*, 11 (Nov. 2011), P11018.

[30] Lothian, J., Powers, S., Sullivan, B. D., Baker, M., Schrock, J., and Poole, S. Synethic graph generation for data-intensive benchmarking: Background and framework. Tech. Rep. ORNL/TM-2013/339, Oak Ridge National Laboratory, Oak Ridge, TN, October 2013.

[31] Medina, A., Lakhina, A., Matta, I., and Byers, J. Brite: Universal topology generation from a user's perspective. Tech. Rep. BUCS-TR-2001-003, Boston Univsersity, April 2001.

[32] Naldi, M. Connectivity of waxman topology models. *Comput. Commun. 29*, 1 (Dec. 2005), 24–31.

[33] Newman, M. E. Assortative mixing in networks. *Phys. Rev. Lett. 89*, 20 (2002), 208701.

[34] Newman, M. E. J. The structure and function of complex networks. *SIAM REVIEW 45* (2003), 167–256.

[35] Newman, M. E. J., Moore, C., and Watts, D. J. Mean field solution of the small-world network model. *Physical Review Letters 84* (2000), 3201–3204.

[36] Pastor-Satorras, R., and Vespignani, A. *Evolution and Structure of the Internet: A Statistical Physics Approach*. Cambridge University Press, New York, NY, USA, 2004.

[37] Seshadhri, C., Kolda, T. G., and Pinar, A. Community structure and scale-free collections of erdős-rényi graphs. *Physical Review E 85*, 5 (May 2012), 056109.

[38] Seshadhri, C., Pinar, A., and Kolda, T. G. An in-depth analysis of stochastic kronecker graphs. *arXiv:1102.5046* (Feb. 2011).

[39] Van Mieghem, P. Paths in the simple random graph and the waxman graph. *Probab. Eng. Inf. Sci. 15*, 4 (Oct. 2001), 535–555.

[40] Van Mieghem, P., Wang, H., Ge, X., Tang, S., and Kuipers, F. A. Influence of assortativity and degree-preserving rewiring on the spectra of networks. *The European Physical Journal B 76*, 4 (2010), 643–652.

[41] Wang, H., Winterbach, W., and Van Mieghem, P. Assortativity of complementary graphs. *The European Physical Journal B 83*, 2 (2011), 203–214.

[42] Watts, D. J., and Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature 393*, 6684 (June 1998), 440–442.

[43] Young, S. J. *Random Dot product Graphs: A FLEXIBLE MODEL FOR COMPLEX NETWORKS*. PhD thesis, Georgia Institute of Technology, November 2008.

[44] Young, S. J., and Scheinerman, E. Directed random dot product graphs. *Internet Mathematics 5*, 1-2 (2008), 91–112.