Assortativity in Chung Lu Random Graph Models

Stephen Mussmann[1,2,3]*, John Moore[1,2]*, Joseph J. Pfeiffer III[1], Jennifer Neville[1,2]
Departments of Computer Science[1], Statistics[2], and Mathematics[3]
Purdue University, West Lafayette, IN
{somussma,moore269,jpfeiffer,neville}@purdue.edu
*Authors contributed equally

ABSTRACT

Due to the widespread interest in networks as a representation to investigate the properties of complex systems, there has been a great deal of interest in generative models of graph structure that can capture the properties of networks observed in the real world. Recent models have focused primarily on accurate characterization of sparse networks with skewed degree distributions, short path lengths, and local clustering. While assortativity—degree correlation among linked nodes—is used as a measure to both describe and evaluate connectivity patterns in networks, there has been little effort to explicitly incorporate patterns of assortativity into model representations. This is because many graph models are edge-based (modeling whether a link should be placed between a pair of nodes i and j) and assortativity is a second-order characteristic that depends on the global properties of the graph (i.e., the final degree of i and j). As such, it is difficult to incorporate direct optimization of assortativity into edge-based generative models.

One exception is the BTER method [5], which generates graphs with positive assortativity (e.g., high degree nodes link to each other). However, BTER does not directly estimate assortativity and also is not applicable for networks with negative assortativity (e.g, high degree nodes link primarily to low degree nodes). In this work, we present a novel approach to directly model observed assortativity (both positive and negative) via accept-reject sampling. Our key observation is to use a coarse approximation of the observed *joint degree distribution* and modify the likelihood that two nodes i, j should link based on the output properties of the original model. We implement our approach as an augmentation of Chung-Lu models and refer to it as *Binning* Chung Lu (BCL). We apply our method to six network datasets and show that it captures assortativity significantly more accurately than other methods while maintaining other graph properties of the original CL models. Also, our BCL approach is efficient (linear in the number of observed edges), thus it scales easily to large networks.

KDD 2014 New York, NY, USA

http://dx.doi.org/10.1145/2659480.2659495.

1. INTRODUCTION

Due to the widespread interest in networks as a representation to investigate the properties of complex systems, there has been a great deal of interest in generative models of graph structure that can capture the properties of networks observed in the real world. More specifically, scalable models of social and information networks have facilitated the study of the structure of the Internet and World Wide Web, and provided foundational support for the development of algorithms and systems deployed in those online environments.

Research on generative models of graphs has focused primarily on accurate characterization of sparse networks with skewed degree distributions, short path lengths, and local clustering. Recent models include the Chung-Lu Graph Model (CL) [2] and Transivitive Chung Lu (TCL) [9]. CL will, in expectation, match the degree distribution in the original graph, and it has scalable methods to learn and generate graphs with hundreds of thousands of nodes and millions of edges. TCL is an extension of CL, which attempts to model clustering via triangle closures. Sometimes the TCL generation method chooses to add an edge based on the CL process, while other times it chooses to follow a transitive edge and then close a triangle. The probability of using the CL process versus the transitive process is controlled by a single parameter, which can be learned quickly on large networks with millions of edges.

While assortativity—degree correlation among linked nodes—is often used as a measure to both describe and evaluate connectivity patterns in networks (see [8, 19, 12]), there has been little effort to explicitly incorporate patterns of assortativity into model representations. This is because many graph models are edge-based (modeling whether a link should be placed between a pair of nodes i and j) and assortativity is a second-order characteristic that depends on the global properties of the graph (i.e., the final degree of i and j). As such, it is difficult to incorporate direct optimization of assortativity into edge-based generative models. Specifically, CL and TCL do not model assortativity, which is reflected in networks sampled from their distributions.

Two existing models that produce networks with assortativity are the Block Two-Level Erdos-Renyi (BTER) graph model [5], and the Joint Degree Distribution (JDD) model [16]. The BTER model [5] groups vertices with similar degrees into blocks of vertices: vertices within the same block have higher probability of linking, while vertices across blocks have significantly lower probability. Thus, the networks produced have a high amount of clustering and posi-

tive assortativity. Further, increasing the amount of assortativity impacts the amount of clustering (and vice versa). As a result, BTER cannot model networks where the assortativity is independent of the clustering. This includes negative assortativity (i.e., high degree nodes link primarily to low degree nodes), which occurs in many real-world networks. The recent JDD model [16] estimates a joint distribution over the degrees of incident nodes, and since assortativity is simply a function of the joint degree distribution, JDD can model it. However, the JDD model does not have a scalable sampling method, since its mixing time is currently quadratic in the number of nodes. Also, the JDD does not model other more global network characteristics such as clustering.

In this work, we present a novel approach to augment edge-based statistical models of graphs to direct optimize observed assortativity (both positive and negative) via accept-reject sampling. Our key observation is to use a coarse approximation of the observed *joint degree distribution* and modify the likelihood that two nodes i, j should link based on the output properties of the original model. Specifically, to generate graphs that accurately reflect a target level of assortativity, we enhance the Chung Lu model as follows. We use a binning technique to estimate the joint degree distribution, where edges from the original graph are counted in bins based on the degrees of the two incident nodes. We then compute the *output* bin frequencies that occur when a graph is sampled from a CL model. The difference between the two determines the *acceptance* rates to use in a statistical sampling process that modifies the likelihood of i, j pairs to better match the target bin frequencies. We implement this general approach to modeling assortativity for both CL and TCL models and refer to it as *Binning* Chung Lu (BCL).

We apply our method to six network datasets and show that it is able to accurately capture assortativity (significantly better than the baseline methods) while maintaining the other graph properties of the original models. Additionally, the BCL approach is efficient (linear in the number of observed edges), thus it scales to large networks. We will demonstrate how BCL was able to learn and sample from a large Patents dataset with 14 million edges.

Our contributions can be summarized as follows:

- Introduction of a coarse binning technique to capture the joint degree distribution (rather than the scalar measure of assortativity) during optimization.
- Development of an accept-reject sampling process to augment existing edge-based generative models to capture observed assortativity in the input networks.
- Implementation of our *binned* sampling approach with respect to two Chung Lu models.
- Empirical demonstration showing graphs generated with our proposed method maintain the degree and clustering properties of the original models, while at the same time accurately modeling assortativity (thus providing significant gains over previous methods).

Section 2 outlines the notation used, discusses related work, and introduces the Chung Lu model with its variants. Section 3 outlines our binning approach, which results in approximately matching the assortativity coefficient. In section 4, we analyze our experimental rsults and compare to competing models. We conclude in section 5.

2. NOTATION AND BACKGROUND

Let a graph $G = \langle \mathbf{V}, \mathbf{E} \rangle$ define a set of *vertices* or nodes $\mathbf{V}$, with a corresponding set of edges $\mathbf{E} \subset \mathbf{V} \times \mathbf{V}$. Let us denote the edges e_{ij} where $e_{ij} = (v_i, v_j)$. Note that since the graph is undirected, $e_{ij} \in \mathbf{E}$ if and only if $e_{ji} \in \mathbf{E}$.

Let D_i be the degree of node v_i in the graph. Let M be the total number of edges in the graph. Note that $|\mathbf{E}| = 2M$ since each undirected edge corresponds to two elements in $\mathbf{E}$. We can refer to the number of nodes of each degree as the degree distribution of the graph.

We will refer to three nodes mutually connected as a triangle. Further we will define a path of length two in the graph as a wedge. The local clustering coefficient is the ratio of the number of triangles incident to a node compared to the number of wedges centered on that node [18]. The distribution of clustering coefficients of nodes is a structural characteristic of a graph. Intuitively, high clustering coefficients are indicative of a graph where connected nodes have many mutual neighbors.

Finally, let us define "closing a triangle" as the process of adding an edge to a wedge to create a triangle.

Assortativity is a graph metric that has been used in other work [8]. Let us define $K = \{(D_i, D_j) | e_{ij} \in \mathbf{E}\}$. The assortativity is the sample Pearson correlation coefficient for the data K:

$$\mathcal{A} = \frac{\sum_{e_{ij} \in \mathbf{E}} (D_i - \overline{D})(D_j - \overline{D})}{\sum_{e_{ij} \in \mathbf{E}} (D_i - \overline{D})^2}$$

where $\overline{D} = \frac{1}{|\mathbf{E}|} \sum_{e_{ij} \in \mathbf{E}} D_i$.

2.1 Generative models of graphs

A primary concern within the social network community is how to define a process to generate edges in a network. In their seminal 1960 paper, Erdos and Renyi proposed the Erdos-Renyi Generative Graph Model, which is the first random graph model [3]. This model treats every pair of edges in the network as an independent Bernoulli trial, each existing with the same probability.

However, despite advantages provided by the Erdos-Renyi graph model, it also has shortcomings. In particular, commonly observed statistics in social networks such as the degree distribution, degree assortativity and clustering coefficients are not modeled [18, 8]. The Chung Lu family of models [2] theoretically and empirically correct the degree distribution statistic. In addition, extensions such as Transitive Chung Lu (TCL) [9] and Block Two-level Erdos Renyi (BTER) [5], model the clustering coefficients as well as the degree distribution. BTER also creates assortativity in networks, but does not learn the amount of assortativity and doesn't generate negative assortativity. This is a limitation to BTER as networks have varying amounts of assortativity that are commonly negative[8]. In contrast, we will explictly model both the amount of assortativity in the network and allow for modeling of negative assortativity.

A central requirement for generative graph models of social networks is their scalability. Modern networks can have tens of millions (e.g. Patents) to billions of vertices and edges. To be of practical use, models must learn and sample from the graph distribution in subquadratic time. Exponential Random Graph Models (ERGMs) [15] can theoretically model the degree distribution, clustering and assortativity, but their learning and sampling is (at best) quadratic in run-

time. Sampling from the Joint Degree Distribution (JDD) model is quadratic over the number of nodes [16]. In contrast, the CL family of models (and our proposed extension) is subquadratic, making them scale to large data.

Lastly, the Attributed Graph Model (AGM) [10] influences parts of our work. AGM extends scalable generative graph models to sample from the joint distribution of edges and attributes. By using Accept-Reject sampling [7], AGM proposes edges using a generative graph model and accepts them if they are from the joint distribution of attributes and edges. In contrast, we utilize Accept-Reject sampling to define a new distribution of structural networks which incorporate assortativity.

2.2 Chung-Lu Models

Chung Lu models are characterized by the marginal probability of edges being proportional to the degrees of the endpoints. One example of a simple Chung Lu model is the Fast Chung Lu (FCL) algorithm [11]. Another example is the Transitive Chung Lu (TCL) [9], a model that extends the Fast Chung Lu (FCL) algorithm. FCL is a generative graph model that generates graphs with the same degree distribution. TCL also preserves this property but additionally, TCL preserves the clustering coefficient distribution of the original graph. In the next section, we will propose a method that can be augment Chung Lu models to additionally preserve the assortativity of the input graph.

FCL generates a graph by repeatedly sampling edges and adding them to the graph. Let G be an observed graph. Let π be a sampling distribution for nodes such that $\pi(i) = \frac{D_i}{2M}$. On each iteration, the FCL algorithm samples twice from π to select an i, j pair and then adds the edge e_{ij} to the output graph G', connecting the two sampled nodes.

Let D_i^{FCL} be the random variable of the degree of node v_i in the graph generated by FCL. Since the algorithm samples from the degree distribution twice for each of the M edges,

$$\mathbb{E}\left[D_i^{FCL}\right] = M * [\pi(i) + \pi(i)] = M \cdot 2\frac{D_i}{2M} = D_i$$

Thus FCL preserves the degree distribution of the input graph in expectation. In practice, it is possible to sample an edge that is already in the graph. In these cases, FCL places the two endpoints on a queue instead of adding the edge between them. Then, in the future, as long as the queue is non-empty, instead of sampling from π for the first node, the first node is selected by removing the node at the front of the queue. With this addition, FCL will produce a graph without edge collisions that still preserves the degree distribution in general.

However, for most graphs, the observed clustering coefficient is much higher than what is generated by FCL. The objective of generative graph models is to generate a graph with similar structural characteristics. Thus, FCL has a shortcoming by not matching the clustering coefficients; TCL [9] fixes this.

TCL still samples nodes from π to form edges, but introduces clustering by sometimes choosing (with probability ρ) the target node j from the two-hop neighbors of the sampled node i. This is equivalent to closing a triangle. These extra triangles will increase the clustering coefficient. With a judicious choice of ρ, the clustering coefficient distribution of the generated graph will approximately match that of the input graph. This single parameter can be learned by an Expectation-Maximization method after defining hidden variables that indicate whether an edge was added by the original FCL process or by closing a triangle. See [9] for more detail.

2.3 Accept Reject Sampling

Accept reject sampling is a method used in statistics [7] that we will employ in our BCL method. Suppose you have two computable distributions Q and Q', where Q' is a *proposal* distribution that is easy to sample from, but we wish to sample from the *target* distribution Q that is more difficult to sample from directly. Using accept reject sampling, we can repeatedly sample from Q' but only *accept* the sample some of the time. In particular, for each value x that we sample from Q', we flip a Bernoulli coin with probability proportional to $\frac{Q(x)}{Q'(x)}$ and only accept the sample x if the Bernoulli trial is a success. We repeat this process until success. In particular the probability that x is sampled is:

$$A(x) = \frac{Q(x)}{Q'(x)M}$$

where M is a constant such that $A(x) \leq 1$. Thus, the probability of a success is

$$\sum_x A(x)Q'(x) = \sum_x \frac{Q(x)}{M} = \frac{1}{M}$$

Note that the distribution of trials necessary for success is a geometric distribution with success probability $\frac{1}{M}$ so the probability of choosing x using the accept reject sampling is

$$\sum_t (1 - \frac{1}{M})^t A(x)Q'(x) = M\frac{Q(x)}{M} = Q(x)$$

which is what we desired.

3. ASSORTATIVITY IN GRAPH MODELS

Intuitively, assortativity measures the tendency for edges to be placed between nodes of similar degree. If a graph has negative assortativity, nodes tend to connect to nodes with dissimilar degree. Conversely, if a graph has positive assortativity, nodes tend to connect to similar degree nodes.

It is possible that there would be no correlation between the degrees of nodes across an edge in a graph. However, for many real-world datasets, there is such a correlation and in some cases, it is a rather strong correlation. See Table 1 for the observed assortativity ($\mathcal{A}$) of such graphs. Facebook wall has nodes for users and edges between people who post on each other's walls. Purdue Email is a graph where nodes are email users and edges are between those who have exchanged emails. The Gnutella dataset represents a Peer2Peer network. Epinions is a graph where nodes are Epinion users and edges are between users who 'trust' each other. The Rovira dataset is similar to Purdue email but consists of emails at the University Rovira i Virgili in Tarragona. Lastly, Patents is a citation network of US patents. These datasets will be discussed more in depth later.

It is important to note that the assortativity doesn't fully capture the distribution of the degrees of nodes across edges. It is well known that linear correlation misses a lot of information in most two-dimensional distributions. Assortativity is not an exception. To illustrate this, we give examples of two graphs with the same assortativity but very different distributions of degrees across edges. Each graph is composed

Table 1: Network Statistics

Graph	Nodes	Edges	$\mathcal{A}$	$\hat{A}_{TCL}$
Facebook Wall	444,829	1,014,542	-0.297	-0.0021
Purdue Email	54,076	880,693	-0.1161	-0.0092
Gnutella	36,682	88,328	-0.1034	0.0006
Epinions	75,865	385,418	0.0226	-0.0363
Rovira Email	1,133	5,451	0.0782	-0.0200
Patents	2,745,762	13,965,410	0.1813	0.0004

Table 2: Graph A

Number	Subgraph
1782	5-star
16	11-clique

Table 3: Graph B

Number	Subgraph
8910	2-clique
1782	6-clique
176	10-star

Table 4: Degree Distributions

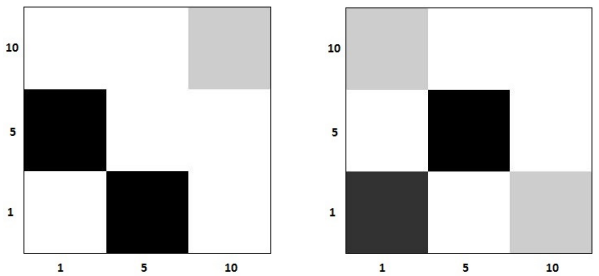
Degree	Graph A	Graph B
1	8910	8910
5	1782	1782
10	176	176

of disconnected subgraphs that are either stars or cliques. An n-star is composed of one node connected to n one-degree nodes. An n-clique is composed of n nodes where each node is connected to every other node. The compositions of these graphs can be seen in Tables 2 and 3. Both of these graphs have the exact same degree distributions as shown in Table 4 and have the same assortativity of $\mathcal{A} = \frac{9}{187}$.

We now present a method to (coarsely) visualize the joint degree distribution in Figures 1a and 1b. In this method, we divide the degrees of the graph into k sets or bins that we refer to as $\mathbf{B}_k = [B_1, B_2, ..., B_k]$. Let $b(D)$ be a function that returns the set membership for a given degree D. For the two graph examples, we can use $k = 3$ and let $B_1 = \{1\}, B_2 = \{5\}, B_3 = \{10\}$. Thus the nodes of degree 1 are in one set, the degree 5 nodes in another, and those of degree 10 in the final set.

Now we can construct a $k \times k$ matrix $\mathcal{B}$ to represent the joint degree distribution, where each cell i, j counts the number of edges between nodes with degrees in B_i and those with degree B_j. In other words, an edge e_{ij} would be counted in cell $b(D_i), b(D_j)$. So for instance, in our case, all edges where both endpoints have degree 5 would be placed in $\mathcal{B}_{2,2}$. When visualizing $\mathcal{B}$, we use a gray scale intensity plot to indicate the number of edges in each cell (i.e., a cell without any edges will be colored white and a cell with the largest amount of edges will be close to black). Figures 1a and 1b visualize $\mathcal{B}$ for the example graphs A and B.

Assuming the sets B_n are ordered by degree of vertices, as we do above, these plots are closely related to assortativity. If the dark boxes lie in a line with positive slope, the graph will have positive assortativity whereas if the dark boxes lie in a line with negative slope, the graph will have negative assortativity. In fact, these *binned* plots are a histogram



(a) Graph A (b) Graph B

Figure 1: Joint degree distributions

approximation of the full joint distribution of the degrees of endpoints for edges. The assortativity is then merely the correlation coefficient of this distribution.

It can be seen in Figures 1a and 1b that the distribution is very different for the two graphs despite their having the same assortativity and degree distributions. This example illustrates and demonstrates the claim that the single dimensional measure of assortativity does not fully capture the joint distribution of the degrees of the endpoints of edges.

Thus far, generative graph models do not explicitly model positive and negative assortativity. The various Chung Lu models, CL, FCL, and TCL do not create nonzero assortativity and BTER generates positive assortativity as a byproduct, but does not attempt to maintin the assortativity value from the original graph. There are also generative models that attempt to model joint degree distributions, which is the probability that an edge randomly selected will be between nodes of certain degrees. Assortativity can be used as a sufficient statistic for modeling joint degree distributions, and Stanton and Pinar [17] are able to construct simple graphs that match a given joint degree distribution. However, their MCMC approach to sampling is quadratic in the number of nodes, thus is it not practical to apply for large networks. In contrast, our binning approach to modeling assortativity with Chung Lu models is linear in the number of edges, thus BCL makes it feasible to generate large-scale networks with assortativity.

3.1 Binning Approach

Most generative graph models are not able to reproduce assortativity, and even fewer model negative assortativity. As this is a structural characteristic of a graph, it is an advantage for generative graph models to preserve this metric in the resulting graph. While the Chung Lu models presented preserve other metrics, the processes produce no correlation between the degrees of endpoints of edges and therefore the generated graphs will have trivial assortativity (see Figure 1, last column). We propose the Binning Chung Lu (BCL) model to capture assortativity. BCL uses an existing edge-by-edge generative graph model (CL) and augments it with accept-reject sampling based upon ij degree combinations to better match assortativity in the network. In fact, BCL is general enough to augment any CL model.

We use the binning idea introduced in the previous section in BCL. We define the sets of vertices $\mathbf{B}_k$ in a particular way. Let us create a "degree vector" as an ordered set of the vertices where each vertex v_i is repeated D_i times. So a given vertex with degree 3 would appear 3 times in the degree vector. As such, the size of the degree vector is $|\mathbf{E}|$.

Finally let us sort this vector by the degree of the vertices so the vertices with low degree appear on one end and the vertices with high degree appear on the other end.

Let us now divide the degree vector into k consecutive parts. We define these parts B_p as the set of degrees that fall between the $(p-1)^{th}$ quantile and the p^{th} quantile of the degree vector. Note that these sets $\mathbf{B}_k$ are ordered by degree. Finally, we can use these sets to count the edges e_{ij} in the cell $b(D_i), b(D_j)$ of $\mathcal{B}$ as discussed earlier.

Given this coarse, *binned* representation of the joint degree distribution $\mathcal{B}$, we can outline an accept-reject sampling method as discussed in the background section to optimize it during graph generation. The original CL model is the proposal distribution Q' that is easy to sample from and we define the target distribution Q to be network distribution with the same binned joint degree distribution as the input graph.

To implement accept-reject sampling, we first compute $\mathcal{B}$ from the observed input graph G. Then we generate a graph G' from a Chung Lu model CL and compute, using the same bins $\mathbf{B}$, the joint degree distribution $\mathcal{B}'$ from G'. From these, we can calculate the acceptance probabilities A are calculated in the following way.

$$R(m,n) = \frac{\#\{e_{ij} \in \mathbf{E} \mid b(D(i)) = m \wedge b(D(j)) = n\}}{\#\{e_{ij} \in \mathbf{E}' \mid b(D(i)) = m \wedge b(D(j)) = n\}} = \frac{\mathcal{B}_{mn}}{\mathcal{B}'_{mn}}$$

$$A(m,n) = \frac{R(m,n)}{\max_{mn} R(m,n)}$$

We refer to the sampling process of the CL model as $\pi(\Theta)$. Thus, we sample two nodes i, j from π and after each sample, determine whether to accept or reject the edge e_{ij} depending on the acceptance probability for the bin that the sampled edge falls into. We repeat this process until we have sampled as many edges as the original graph.

This algorithm can be seen in 1. G is the original graph, π is the CL model, and Θ is the parameters for the CL model. Note that we first must generate a preliminary graph using the CL model so that we can compute the acceptance probabilities. In algorithm 1, $\pi(\Theta)$ refers to sampling a target node from a general CL model. Additionally, $b(D_i)$ returns the bin membership of a node i with degree D_i in the input graph G. Thus $A(b(D_i), b(D_j))$ returns the acceptance probability from the appropriate cell of $\mathcal{B}$ for the proposed edge e_{ij}.

Because of the properties of accept-reject sampling, the resulting graph will approximately have the same bin frequencies as $\mathcal{B}$. If two graphs have the same bin frequencies, the joint distribution of endpoint degrees will be approximately the same. This will imply that the assortativity of the generated graph will approximately match that of the input graph. This will be shown empirically in the experiments section. Additionally, it can be shown that our BCL method preserves the degrees of nodes. In particular, when the binning method is used on top of a Chung Lu method, the marginal probability of an edge under the resulting model is proportional to the degrees of the endpoints.

Algorithm 1 Binning Chung Lu Models(G, π, Θ, k)

```
Compute k × k bin frequencies B from G
Generate G' from π, using G and Θ
Compute k × k bin frequencies B' from G'
Compute A(m, n) from B and B'
Create empty graph G^BCL
while |E^BCL| ≤ |E| do
    < i, j >= edge sampled using π(Θ)
    a = A(b(D_i), b(D_j))
    r = bernoulli_sample(a)
    if r = 1 then
        E^BCL = E^BCL ∪ e_ij
    end if
end while
return(G^BCL)
```

4. EXPERIMENTS

Experiments were performed to assess the algorithms. To empirically evaluate the models, we learned model parameters from real-world graphs and then generated new graphs using those parameters. We then compared the network statistics of the generated graphs with those of the original networks.

4.1 Datasets

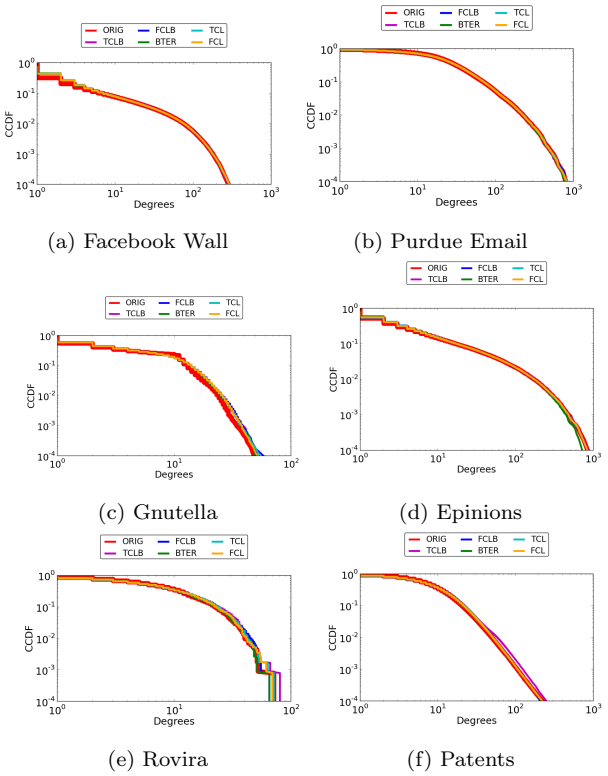
We used six different datasets to evaluate our experimental results. Five of them are all social networking datasets, while the last, patents, is a citation network. Their node and edge counts can be found in Figure 1.

The first dataset we study is a collection of Facebook wall postings from the period 03/01/07–03/01/08, among the set of 56,061 publicly visible users the Purdue University Facebook network. In this network, the users can add each other to their lists of friends and the edge set represents a wall posting between friends.

Next, we study a collection of emails gathered from the SMTP logs of Purdue University [1]. This dataset has an edge between users who sent e-mail to each other. The mailing network has a small set of nodes which sent out mail at a vastly greater rate than normal nodes; these nodes were most likely mailing lists or automatic mailing systems. In order to correct for these 'spammer' nodes, we remove nodes with a degree greater than $1,000$ as these nodes did not represent participants in any kind of social interaction. The network has over two hundred thousand nodes, and nearly two million edges.

The Gnutella30 network is a different type than the other networks presented. Gnutella is a Peer2Peer network where users are attempting to find seeds for file sharing [14]. The user reaches out to its current peers, querying if they have a file. If not, the friend refers them to other users who might have a file, repeating this process until a seed user can be found. Because this network represents the structure of a file sharing program rather than true social interactions, it has significantly less clustering than the other networks.

The next dataset we analyze is the Epinions dataset [13]. This network represents the users of Epinions, a website which encourages users to indicate other users whose consumer product reviews they 'trust'. The reviews of all users on a product are then weighted to incorporate both the reviewer ratings and the amount of trust received from other

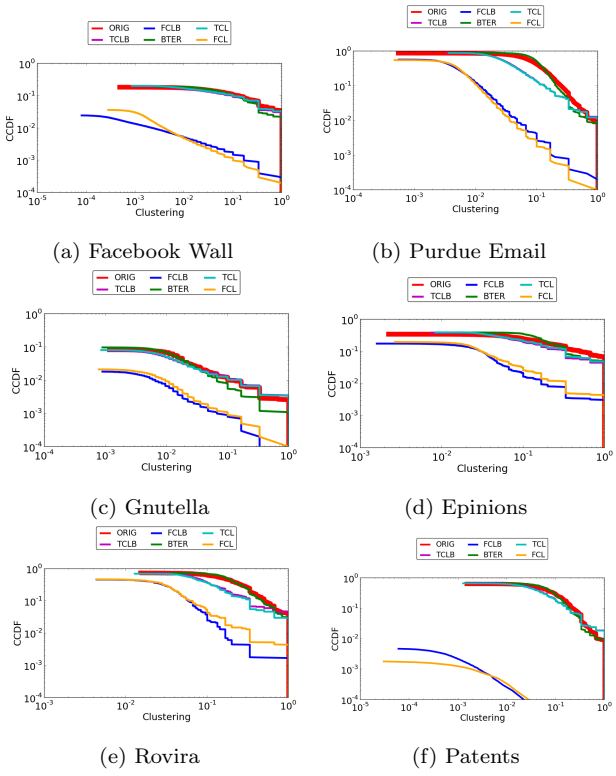


(a) Facebook Wall (b) Purdue Email

(c) Gnutella (d) Epinions

(e) Rovira (f) Patents

Figure 2: Degree Distributions



(a) Facebook Wall (b) Purdue Email

(c) Gnutella (d) Epinions

(e) Rovira (f) Patents

Figure 3: Clustering Coefficients

users. The edge set of this network represents nominations of trustworthy individuals between the users.

In addition to the Purdue email dataset mentioned earlier, we also study the Rovira email dataset [4]. This smaller network consists of a network of students at University Rovira i Virgili in Tarragona. Nodes are users and edges represent that at least one email was sent.

Lastly, we study a citation network of US Patents [6]. Nodes in this network are published patents, while edges indicate one patent cited the other. This is a large network, with over 10 million citations between 2 million edges and demonstrates the scalability of our proposed methods.

4.2 Methods Compared

We compare BCL (in conjunction with TCL and FCL) against the Block Two-Level Erdos-Renyi (BTER) model[1] [5], TCL, and FCL. The BTER model [5] groups vertices with similar degrees into blocks with high probability, resulting in networks with a high amount of clustering and positive assortativity. As a result, BTER cannot model networks where the assortativity is independent of the clustering, meaning augmenting BTER with BCL would interfere with the clustering that BTER models. In contrast, the degree and clustering statistics of FCL and TCL are independent from the assortativity. Thus, we compare FCLB (BCL with FCL proposal distribution) and TCLB (BCL with TCL proposal distribution) against FCL, TCL and BTER. We will demonstrate how TCLB can jointly model degree, assortativity and clustering, in contrast to any of the baseline models.

[1]Downloaded from *www.sandia.gov/ tgkolda/feastpack*

4.3 Methodology

We ran experiments on six datasets using five different algorithms. The five algorithms were Fast Chung Lu (FCL), Transitive Chung Lu (TCL), Fast Chung Lu with Binning (FCLB), Transivitive Chung Lu with Binning (TCLB), and BTER. For evaluation, we compared the graphs generated by the algorithms using different metrics. The metrics used were the complementary cumulative distribution function for both the degree distribution and the distribution of local clustering coefficients. Additionally, the assortativity coefficient and distribution of the degree of nodes across edges were compared. To compare these, we will use a binning number of 10 bins, and split the degree vector of the original graph based upon 9 quantiles (since we had 10 bins). We will plot the edges between nodes by graphing the total number of connections between quantiles. Low degree bins are in the lower left corner while high are to the right and at the top. Thus, the edges in the lower left bins are low degree vertices connected to low degree vertices. Each generated graph will have the same quantile cutoff points by using the original graphs.

4.4 Results

In figure 2, the degree distributions can be seen. It is apparent that all the Chung Lu models and BTER closely match the degree distribution of the original graph.

In figure 3, the local clustering coefficient distributions can be seen. The Fast Chung Lu models consistently miss the distribution of the original graph. However, the transitive version works significantly better along with BTER. Additionally, the binning method doesn't significantly change the

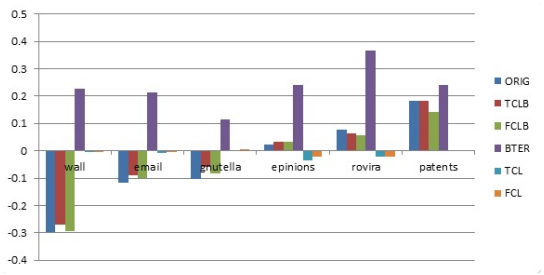
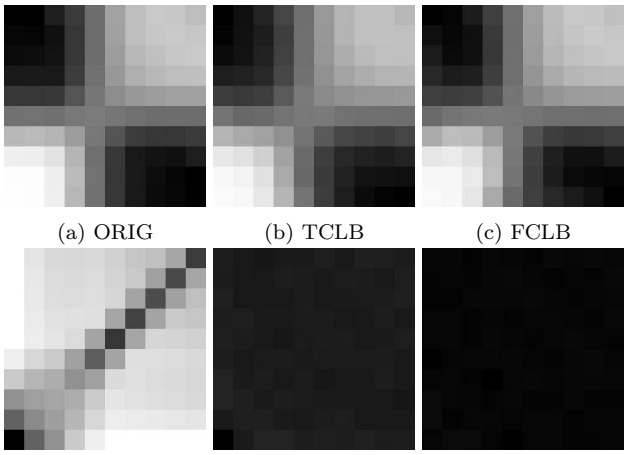


Figure 4: Assortativity

clustering even if the clustering is different from the original graph as in FCL.

Finally, the assortativity can be seen in figure 4. It can be seen that the non-binning Chung Lu models have assortativity very close to zero. However, the Chung Lu models with binning match the assortativity well. BTER has positive assortativity that doesn't match that of the original graph in general.

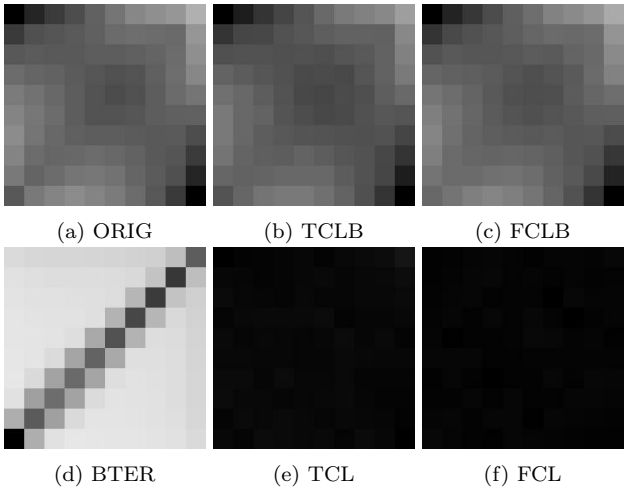
However, as mentioned earlier, assortativity doesn't completely describe the "assortativity distribution". For this reason, we have shown the distributions. The graphs are set up so that the axes are quantile scales rather than absolute scales just as the bins are defined. This is important to note because the assortativity is measured on the absolute scale which is linear while the quantile scales turn out to be more logarithmic in most cases because of the non-uniform degree distribution. However, note that since the Chung Lu methods roughly match the degree distributions, if the distributions align on the quantile scales, they will also align on the absolute scales. The darker regions correspond to a higher density of edges. White always corresponds to zero density while black refers to the maximum density for a given graph. The graphs can be seen in figures 5, 6, 7, 8, 9, and 10. It can be seen that the binning method creates graphs that have very similar assortativity distributions to the original graph while BTER creates very different assortativity graphs that look like a simple linear correlation.



(a) ORIG (b) TCLB (c) FCLB

(d) BTER (e) TCL (f) FCL

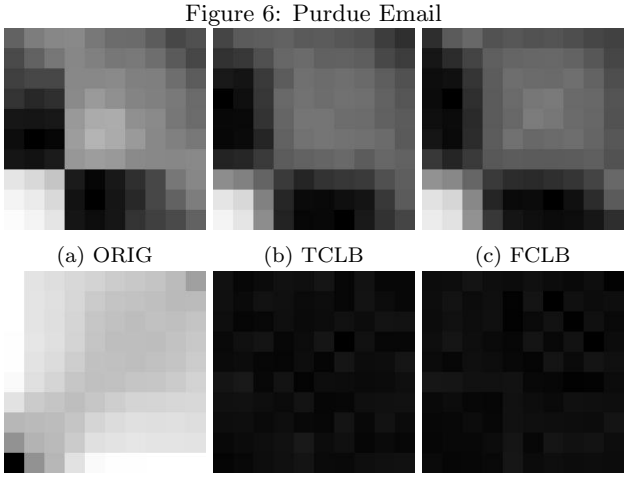
Figure 5: Facebook Wall



(a) ORIG (b) TCLB (c) FCLB

(d) BTER (e) TCL (f) FCL

Figure 6: Purdue Email



(a) ORIG (b) TCLB (c) FCLB

(d) BTER (e) TCL (f) FCL

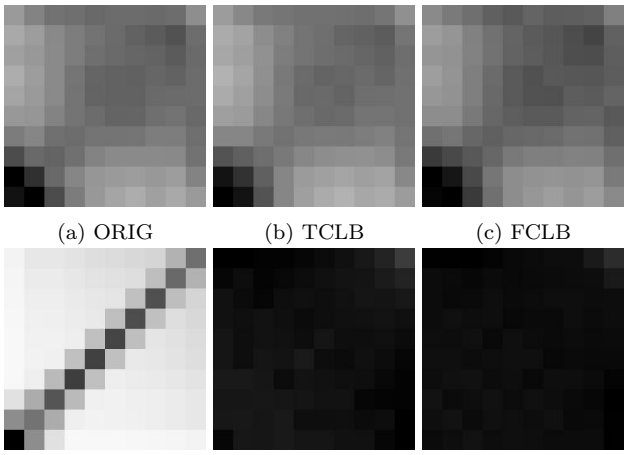
Figure 7: Gnutella

5. CONCLUSIONS

Chung Lu models have shown to be effective in preserving some of the structural attributes of graphs. However, there has not been one developed that preserves the assortativity of graphs. Further, assortativity is a limited structural characteristic and the assortativity distribution better captures the structural nature of the graph. In this paper, we proposed a course binning technique to capture the joint degree distribution of endpoints of edges. Using accept reject sampling with these bins, a method was presented that can be placed on top of a Chung Lu method to preserve the assortativity without increasing the computational complexity by more than a constant factor.

This method was used with two different Chung Lu models, Fast Chung Lu and Transitive Chung Lu. Empirically, it was shown that this method was effective in preserving assortativity of the original graph and not changing degree distribution or clustering coefficient of the non-binning version. This is in contrast to BTER and the non-binning versions which do not match the assortativity.

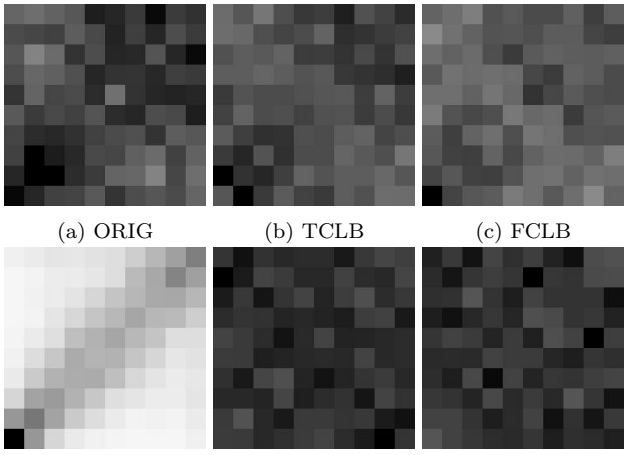
The effects of the binning method could be tested on other Chung Lu models perhaps such as those that additionally model graph attributes such as AGM. Additionally, the binning method could perhaps be generalized to be used on



(a) ORIG (b) TCLB (c) FCLB

(d) BTER (e) TCL (f) FCL

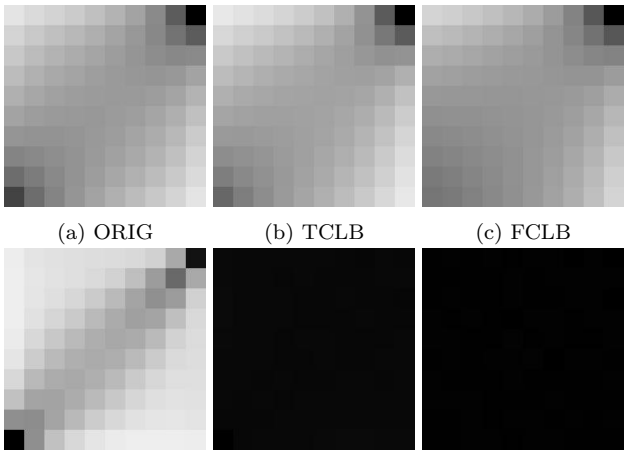
Figure 8: Epinions



(a) ORIG (b) TCLB (c) FCLB

(d) BTER (e) TCL (f) FCL

Figure 9: Rovira



(a) ORIG (b) TCLB (c) FCLB

(d) BTER (e) TCL (f) FCL

Figure 10: Patents

top of models that are not Chung Lu models such as BTER so that the assortativity of the original graph could be preserved.

6. REFERENCES

[1] N. Ahmed, J. Neville, and R. Kompella. Network sampling: From static to streaming graphs. *ACM Transactions on Knowledge Discovery from Data*, 2014.

[2] F. Chung and L. Lu. The average distances in random graphs with given expected degrees. *Internet Mathematics*, 1, 2002.

[3] P. Erdos and A. Renyi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5:17–61, 1960.

[4] R. GuimerÌ, L. Danon, A. DÌ?az-Guilera, F. Giralt, and A. Arenas. Self-similar community structure in a network of human interactions. *Phys. Rev. E*, 68(6):065103, 2003.

[5] T. G. Kolda, A. Pinar, T. Plantenga, and C. Seshadhri. A scalable generative graph model with community structure. arXiv:1302.6636, February 2013. revised March 2013.

[6] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *In KDD*, pages 177–187. ACM Press, 2005.

[7] F. Liang, C. Liu, and R. J. Carrol. *Advanced Markov chain Monte Carlo methods: learning from past samples*. Wiley Series in Computational Statistics. Wiley, New York, NY, 2010.

[8] M. E. J. Newman. Assortative mixing in networks. *Physical Review Letters*, 89(20):208701, Oct. 2002.

[9] J. J. Pfeiffer III, T. La Fond, S. Moreno, and J. Neville. Fast generation of large scale social networks while incorporating transitive closures. In *(SocialCom)*, 2012.

[10] J. J. Pfeiffer III, S. Moreno, T. La Fond, J. Neville, and B. Gallagher. Attributed graph models: Modeling network structure with correlated attributes. In *Proceedings of the 23rd International World Wide Web Conference (WWW 2014)*, 2014.

[11] A. Pinar, C. Seshadhri, and T. G. Kolda. The similarity between stochastic kronecker and chung-lu graph models. *CoRR*, abs/1110.4925, 2011.

[12] M. Piraveenan, M. Prokopenko, and A. Y. Zomaya. Local assortativity and growth of internet. *The European Physical Journal B*, 70(2):275–285, 2009.

[13] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. In *In Proceedings of the Second International Semantic Web Conference*, pages 351–368, 2003.

[14] M. Ripeanu, A. Iamnitchi, and I. Foster. Mapping the gnutella network. *IEEE Internet Computing*, 6(1):50–57, Jan. 2002.

[15] G. Robins, P. Pattison, Y. Kalish, and D. Lusher. An introduction to exponential random graph (p*) models for social networks. *Social Networks*, May 2007.

[16] I. Stanton and A. Pinar. Constructing and sampling graphs with a prescribed joint degree distribution. *CoRR*, abs/1103.4875, 2011.

[17] I. Stanton and A. Pinar. Constructing and sampling graphs with a prescribed joint degree distribution. arXiv:1103.4875, August 2011.

[18] D. J. Watts and S. H. Strogatz. Collective dynamics of'small-world'networks. *Nature*, 393(6684):409–10, 1998.

[19] D. Zhou, H. E. Stanley, G. D'Agostino, and A. Scala. Assortativity decreases the robustness of interdependent networks. *Phys. Rev. E*, 86:066103, Dec 2012.