Tuning Topology Generators Using Spectral Distributions

Hamed Haddadi[1], Damien Fay[2], Steve Uhlig[3], Andrew Moore[2],
Richard Mortier[4], Almerima Jamakovic[3], Miguel Rio[1]

[1] University College London
[2] University of Cambridge
[3] Delft University of Technology
[4] Vipadia Ltd

Abstract. An increasing number of synthetic topology generators are available, each claiming to produce representative Internet topologies. Every generator has its own parameters, allowing the user to generate topologies with different characteristics. However, there exist no clear guidelines on tuning the value of these parameters in order to obtain a topology with specific characteristics.

In this paper we optimize the parameters of several topology generators to match a given Internet topology. The optimization is performed either with respect to the link density, or to the spectrum of the normalized Laplacian matrix. Contrary to approaches in the literature that rely only on the largest eigenvalues, we take into account the set of all eigenvalues. However, we show that on their own the eigenvalues cannot be used to construct a metric for optimizing parameters. Instead we present a weighted spectral method which simultaneously takes into account all the properties of the graph.

Key words: Internet Topology, Graph Spectrum

1 Introduction

Today's Internet is formed from more than 25,000 Autonomous Systems (ASes), each of which can contain tens or hundreds of routers. Constant evolution and change in the Internet, due to failures and bugs in the short term, and growth and death of networks in the long term, has made it difficult for scientists to produce representative Internet topologies at either AS or router level. However, such maps are essential for the simulation and analysis of ideas including new and improved routing protocols, and peer-to-peer, media-streaming applications. Since obtaining accurate, timely maps of the Internet topology is difficult, and development of new protocols and systems requires understanding their performance over a range of scenarios, researchers use synthetic topology generators.

There are many such generators, each of which is parameterized, often with multiple parameters, giving rise to a plethora of potential synthetic graphs. Understanding and generating those graphs, useful because they accurately represent features of the true underlying Internet graph, is difficult. Existing approaches to tuning the generator parameters range from selection of particular metrics of interest, e.g., link count, and tuning to match that particular metric, to simply using the default parameters encoded in the particular release of the generator package in use!

The core problem is to select an appropriate cost function which reflects those aspects of the graph that are important to the user and weights those aspects accordingly. Such a selection process is inherently subjective: there is no "best" cost function in general. Once a suitable cost function is selected, it is a simple matter to tune the available parameters of the topology generator to produce output that optimally matches said cost function.

In the light of this, our contributions in this paper are as follows:

- We propose a new cost function, the *weighted spectrum*, constructed from the eigenvalues of the normalized Laplacian matrix, or graph spectrum;
- We demonstrate that the graph spectrum alone is unsatisfactory as a cost function;
- We provide an efficient approximation of the weighted spectrum;
- We use this approximation to tune parameters for a set of Internet topology generators, enabling us to use these generators to effectively match a particular measured Internet topology.

The graph spectrum is a useful starting point for such a cost function as it yields a set of invariants about a graph that encode all the properties of that graph [8]. Our proposed cost function improves on the simple graph spectrum because it incorporates the knowledge that not all eigenvalues are equally important, and weights toward those that are considered to encode more significant aspects of the graph's structure. The basis of our algorithm is to provide a way to measure the difference between two graphs with respect to a common reference, a suitable regular graph.[5]

After reviewing related work in Section 2, we outline background theory in Section 3 before introducing the topology generators we use in Section 4. In Section 5 we present the results of our analysis and in Section 6 we compare topologies generated at optimal values of the parameters with an observed dataset. Finally, we conclude the paper in Section 7 and discuss future work.

2 Related Work

Zegura *et al.* [27] analyze topologies of 100 nodes generated using pure random, Waxman [25], exponential and several locality based models of topology such as Transit-Stub [6]. They use metrics such as average node degree, network diameter, and number of paths between nodes, and use the number of edges as the metric of choice for optimization of the tuning parameter. However as we show

[5] A regular graph is one where all nodes have the same degree.

in this paper, the number of links is not an ideal choice particularly in random networks, due to the network structure only resembling the observed Internet topology at link counts much higher than those suggested by the optimization process.

Tangmunarunkit *et al.* [23] provide a first point of comparison of the underlying characteristics of degree-based models against structural models. A major conclusion is that the degree-based model in its simplest form performs better than random or structural models at representing all the studied parameters. They compare three categories of model generators: the Waxman model of random graphs, the TIERS [10] and Transit-Stub structural models, and the simplest degree based generator, called the Power-Law Random Graph [1]. They compare under three metrics: expansion, resilience and distortion and conclude that the hierarchy present in the measured networks is stricter than in degree-based generators. However, they leave many questions unanswered about the accuracy of degree-based generators and their choice of metrics and parameter values.

Heckmann *et al.* [15] discuss different types of topologies and present a collection of real-world topologies that can be used for simulation. They then define several similarity metrics, such as the shortest path distributions, node degree distributions and node rank exponents, to compare artificially generated topologies with real world topologies from AT&T's network. They use these to determine the input parameter range of the topology generators of BRITE [19], TIERS and GT-ITM [6] to create realistic topologies.

Gkantsidis *et al.* [13] perform a comparison of clustering coefficients using the eigenvectors of the k largest eigenvalues of the adjacency matrices of BGP topology graphs. However, the choice of k is somewhat arbitrary, and further, the selected eigenvectors are all given equal importance. They consider the rest of the spectrum as noise, although it has been shown that the eigenvalues of either the adjacency matrix or the normalized Laplacian matrix can be used to accurately represent a topology and some specific eigenvalues provide a measure of properties such as robustness of a network to failures [5, 16].

Vukadinovic *et al.* [24] used the normalized Laplacian spectrum for analysis of AS graphs. They propose that the normalized Laplacian spectrum can be used as a fingerprint for Internet-like graphs. Using the Inet [26] generator and AS graphs from BGP data, they obtain eigenvalues of the normalized Laplacian matrix. The differences between synthetic and observed topologies indicate that the structural properties of the Internet should be included in an Internet AS model alongside power law relationships. They believe that the graph spectrum should be considered an essential metric when comparing graphs. We expand on this work by demonstrating how an appropriate weighting of the eigenvalues can be used to reveal structural differences between two topologies.

Use of spectrum for graph comparison is not limited to Internet research. Hanna [14] uses graph spectra for numerical comparison of architectural space in large building plans. By defining space as a graph, he shows that the spectra of two plan types can be used effectively to judge the effects of global vs. local changes to, and hence the edit distances between, the plans. Hanna believes spectra give a reliable metric for capturing the local relationships and can be used to guide optimization algorithms for reproducing plans.

3 Graph Spectra

In this section we introduce a brief overview of graph and establish the techniques used later in the paper. Here we define the spectrum, the associated normalized Laplacian matrix, and several relevant facts relating to this matrix. Given an undirected graph $G = (V, E)$, V is the set of vertices (nodes), E is the set of edges (links) and d_v is the degree of node v.

Definition 1. *For a connected graph the normalized Laplacian of the graph G is the matrix $L(G)$ defined as:*

$$L(G)(u,v) = \begin{cases} 1, & \textit{if } u = v \textit{ and } d_v \neq 0 \\ -\dfrac{1}{\sqrt{d_u d_v}}, & \textit{if } u \textit{ and } v \textit{ are adjacent} \\ 0, & \textit{otherwise} \end{cases} \tag{1}$$

The associated spectrum is the set of ordered eigenvalues of L denoted by $\lambda_1, \lambda_2, \ldots, \lambda_{N-1}$ where N is the number of vertices and the eigenvalues are ordered such that $0 \leq \lambda_1 \leq \cdots \leq \lambda_i \leq \lambda_{i+1} \leq \cdots \leq \lambda_{N-1} \leq 2$. The normalized Laplacian has some very interesting properties, the relevant ones of which we list here:

1. For a connected graph the spectrum is symmetrical around 1 i.e., $\lambda_i = \lambda_{N-i-1}$;
2. If D is the diameter of the graph (the maximum number of steps between all pairs of nodes) and $vol(G)$ denotes the volume of G which is the sum of the node degrees d_v:

$$\lambda_1 \geq \frac{1}{Dvol(G)} = \frac{1}{D\sum\limits_v (d_v)} \tag{2}$$

   Thus, the first eigenvalue is bounded by the node degrees of the vertices.
3. For a connected graph

$$2h_G \geq \lambda_1 \geq \frac{h_g^2}{2} \tag{3}$$

   where h_G is the Cheeger constant and is a measure of the minimum cut-set of a graph, see e.g. [8] for a full explanation. The Cheeger constant is closely related to flow problems in graphs and is thus of obvious importance to network designers.

For these and other reasons, e.g. as presented in [8, 5, 24, 13], the spectrum of a graph is often called the *footprint* of a graph. More specifically, in this paper we evaluate the use of the spectrum as a measure of the deviation of a graph, explained below. A random graph is defined as one for which all but $o(N)$ vertices almost certainly have degree [7]:

$$d_v = \frac{N}{2} + o(N) \tag{4}$$

where $o(N)$ denotes of the order of N. For random graphs there exists a large set of properties which form an equivalence class of properties such that if one of the properties is proven then all are proven, see e.g. [7] for an initial list. There also exist non-random graphs which satisfy the equivalence class of properties. These are known as quasi-random graphs. One of the most tractable properties of the equivalence class of properties is the 4-cycle. A 4-cycle is a route starting and ending at one vertex which passes through 4 points in total, where these may be repeated points:

$$N_G(C_4) \leq (1 + o(1))(\frac{N}{2})^4 \tag{5}$$

where C_4 denotes a 4 cycle and $N_G(C_4)$ denotes the number of such cycles. However, in this paper our interest does not lie in random graphs (those examined here are not random but structured) but in a measure called the deviation of a graph, $dev(G)$, which is a measure of a graph's deviation from pseudo-randomness. For a regular graph, in which each vertex has the same degree, this is defined as the number of 4-cycles. However, this can also be related to the spectrum: in a given graph G with N eigenvalues $\lambda_1, \ldots, \lambda_N$, the deviation is calculated as follows. For a regular graph:

$$dev(G) = \sum_i (1 - \lambda_i)^4 \tag{6}$$

and for a general graph:

$$dev(G) = \sum_i (1 - \lambda_i)^4 + 20\sqrt{Irr(G)} \tag{7}$$

where $Irr(G)$ is the irregularity of the graph [8]. The deviation of a graph may be used as a measure of the structure in a graph, i.e., its distance away from randomness. It is the first term on the right hand side of the bound above which forms the metric proposed in this paper. This term expresses the appropriate weighting, i.e., a power of 4, of the eigenvalues that sum to form the bound on the deviation of a graph.

Next we consider the interpretation of the eigenvalues of the normalized Laplacian matrix. In the following only eigenvalues less than or equal to 1 are considered, as the spectrum is symmetrical for connected graphs. Spectral clustering is a technique which uses the eigenvalues of the normalised Laplacian matrix to perform clustering of a dataset [20]. The first (smallest) eigenvalue and associated eigenvector are associated with the main clusters of data. Subsequent eigenvalues and eigenvectors can be associated with cluster splitting and also identification of smaller clusters [22]. Typically, there exists what is called a spectral gap in which for some $k, \lambda_k << \lambda_{k+1} \approx 1$. That is, eigenvalues $\lambda_{k+1}, \ldots, \lambda_N$ are approximately equal to one and are likely to represent noise in the original dataset. It is then typical to reduce the dimensionality of the data using an approximation based on the spectral decomposition. It is interesting to note that while the normalized Laplacian has well behaved convergence properties with regards to clustering, this is not true for other matrices derived from the adjacency matrix [17]. However, with regards to topological graphs, while the first eigenvalue may be associated, as above, with the optimal cut, which can be considered the optimal cluster, interpretation of subsequent eigenvalues cannot be associated with specific graph properties other than the distribution of cluster information within a graph.

Having established the background material necessary for our method we now examine the construction of a metric for graph comparison. Given two graphs, G_1 and G_2 say we wish to determine at what points their structure vary. As a first attempt one might try to construct a metric based on the differences between the eigenvalues as:

$$C = \sum_i \lambda_{i,G_1} - \lambda_{i,G_2} \tag{8}$$

However, pairwise comparison of the eigenvalues as above leads to comparing eigenvalues which represent different structures in the graph, i.e., it is more appropriate to compare eigenvalues of similar size. In order to achieve this, the distribution of eigenvalues is used to construct our metric as:

$$C = \int_i (1-i)^4 (P(\lambda_{i,G_1} = i) - P(\lambda_{i,G_2} = i)) d_i \tag{9}$$

In this paper the distribution of eigenvalues $P(\lambda_i = i)$ is estimated by using pivoting and Sylvester's Law of Inertia to compute the number of eigenvalues that fall in a given interval.

While the primary motivation for using a power of four in the equation above is the number of 4-cycles, and thus the deviation from random behaviour of a graph as discussed above, an interesting link can also be made with the well known clustering coefficient, as will now be shown. First however, some background must be established. Consider the adjacency matrix for a graph, A, in which:

$$A_{i,j} = 1 \text{ if } i \rightarrow j$$

where $A_{i,j}$ is the ith and jth entry of A. The number of paths of length 2 between nodes i and j, t, can easily be found by squaring the adjacency matrix as:

$$A^2_{i,j} = t \text{ if } i \rightarrow k \rightarrow j$$

for some intermediate node(s) k. In general the t paths of length N between nodes i and j can be found by taking the Nth power of A as:

$$A^N_{i,j} = t \text{ if } i \rightarrow j \text{ via } N \text{ steps.}$$

noting that for a cycle a path must start and finish at the same point gives:

$$A^N_{i,i} = t \text{ if } i \rightarrow i \text{ via an } N \text{ cycle.}$$

Now consider the spectral decomposition of the matrix A:

$$A = \sum_i \gamma_i \epsilon_i {\epsilon_i}^T \tag{10}$$

where γ_i and ϵ_i are the ith eigenpair of A. These form an orthonormal basis for A (i.e. ortogonal $\epsilon_i \epsilon_j^T = 0$ and normal $\epsilon_i \epsilon_i^T = \mathbf{1}$), and so:

$$A^N = (\sum_i \gamma_i \epsilon_i \epsilon_i^T)^N \tag{11}$$

Here we are interested in the number of N-cycles which is the trace of A^N:

$$tr(A^N) = \sum_i \gamma_i^N \tag{12}$$

Thus, for an adjacency matrix the number of N-cycles in the graphs is the sum of the eigenvalues. Next consider the normalised Laplacian which can be related to the adjacency matrix as:

$$L(G) = I - D^{-1/2} A D^{-1/2} \tag{13}$$

where D is a diagonal matrix whose ith entry is the degree of node i. Taking the identity matrix to the left and taking the trace gives:

$$tr(I - L(G)) = tr(D^{-1/2} A D^{-1/2}) \tag{14}$$

However, $tr(I - L(G))$ is also related to the eigenvalues of $L(G)$ as:

$$tr(I - L(G)) = \sum_i 1 - \lambda_i \tag{15}$$

Putting the two results together and taking a power of N results in:

$$tr((I - L(G))^N) = tr((D^{-1/2} A D^{-1/2})^N) = \sum_i (1 - \lambda_i)^N \tag{16}$$

The right hand side of this equation is the weighted spectrum but it is the terms on the left hand side we will now examine. Noting that the i,jth entry of $D^{-1/2} A D^{-1/2}$ is:

$$(D^{-1/2} A D^{-1/2}))_{i,j} = \frac{A_{i,j}}{\sqrt{d_i}\sqrt{d_j}} \tag{17}$$

then an N-path passing through a set of nodes, S say, will consist of a product of $\#S$ such terms:

$$\prod_S \frac{A_{i,j}}{\sqrt{d_i}\sqrt{d_j}} \tag{18}$$

If node i has K N-*cycles*, consisting of the sets $S_{1,\ldots,K}$ then the ith diagonal element of $(I - L(G))^N$ is:

$$(D^{-1/2}AD^{-1/2}))^N_{i,i} = \sum_{k=1}^{K} \prod_{i,j \in S_k} \frac{1}{d_j} \tag{19}$$

Next we consider the clustering coefficient of a graph, G. The cluster coefficient, $\gamma(G)$, is defined as the average number of 3-cycles divided by the total number of possible 3-cycles:

$$\gamma(G) = 1/N \sum_i \frac{T_i}{d_i(d_i - 1)/2}, d_i \geq 2 \tag{20}$$

where T_i is the number of 3-cycles for node i, d_i is the degree of node i. Now consider a specific 3-cycle between nodes a, b and c. For the cluster coefficient the contribution to the average is (noting that the 3-cycle will be considered three times, once from each node):

$$\frac{1}{d_a(d_a - 1)/2} + \frac{1}{d_b(d_b - 1)/2} + \frac{1}{d_c(d_c - 1)/2} \tag{21}$$

However, for the weighted spectrum and taking the number of 3-cycles (Note: 4-cycles are the main focus of this research for reasons explained above), this particular 3-cycle makes the following contribution to the overall sum (i.e. using K=1, $S_k = a, b, c$ for node a then likewise for nodes b and c):

$$\frac{3}{d_a d_b d_c} \tag{22}$$

So it can be seen that the clustering coefficient normalises each 3-cycle according to the total number of possible 3-cycles while the 3-cycle weighted spectrum instead normalises using a product of the degrees. Thus the two metrics can be considered to be similar but not equal. Note also that in contrast to the clustering coefficient (one number) the weighted spectrum results in many terms which represent sucessively finer and finer clusters.

4 Available Topologies

4.1 Synthetic Topologies

There are many models available that claim to describe the Internet AS topology. Several of these are embodied in tools built by the community for generating simulated topologies. In this section we describe the particular models whose output we compare in this paper. The first are produced from the Waxman model [25], derived from the Erdös-Rényi random graphs [11], where the probability of two nodes being connected is proportional to the Euclidean distance between them. The second come from the Barabasi and Albert (BA) [3] model, following measurements of various power laws in degree distributions and rank exponents by Faloutsos *et al.* [12]. These incorporate common beliefs about preferential attachment and incremental growth. The third are from the Generalized Linear Preference model [4] which additionally model clustering coefficients. Finally, Inet [26] and PFP [28] focus on alternative characteristics of AS topology: the existence of a meshed core, and the phenomenon of preferential attachment respectively. Each model focuses only on particular metrics and parameters, and has only been compared with selected AS topology observations.

4.2 Waxman

The Waxman model of random graphs is based on a probability model for interconnecting nodes of the topology given by:

$$P(u,v) = \alpha e^{-d/(\beta L)} \tag{23}$$

where $0 < \alpha, \beta \leq 1$, d is the Euclidean distance between two nodes u and v, and L is the network diameter, i.e., the largest distance between two nodes. Note that d and L are not parameters for the Waxman model. The Internet is known not to be a random network but we include the Waxman model as a baseline for comparison purposes.

4.3 BA

The BA [2] model was inspired by the idea of preferentially attaching new nodes to existing well-connected nodes, leading to the incremental growth of nodes and the links between them. Starting with a network of m_0 isolated nodes, $m \leq m_0$ new links are added with probability p. One end of each link is attached to a random node, while the other end is attached to a node selected by preferring the more popular, i.e., well-connected, nodes with probability

$$\Pi(k_i) = \frac{k_i + 1}{\sum_j k_j + 1} \tag{24}$$

where k_j is the degree of node j, with probability q, m links are rewired and new nodes are added with probability $1-p-q$. A new node m has m new links that, with probability $\Pi(k_i)$, are connected to nodes i already present in the system. We use the BRITE [19] implementation of this model in this paper.

4.4 GLP

Our third model is the Generalized Linear Preference model (GLP) [4]. It focuses on matching characteristic path length and clustering coefficients. It uses a probabilistic method for adding nodes and links recursively while preserving selected power law properties. In the GLP model, when starting with m_0 links, the probability of adding new links is defined as p where $p \in [0,1]$. Let $\Pi(d_i)$ be the probability of choosing node i. For each end of each link, node i is chosen with probability $\Pi(d_i)$ defined as:

$$\Pi(d_i) = (d_i - \beta) / \sum_j (d_j - \beta) \tag{25}$$

where $\beta \in (-\infty, 1)$ is a tunable parameter indicating the preference of nodes to connect to existing popular nodes. We use the BRITE implementation of this model in this paper.

4.5 Inet

Inet [26] produces random networks using a preferential linear weight for the connection probability of nodes after modeling the core of the generated topology as a full mesh network. Inet sets the minimum number of nodes at 3037, the number of ASes on the Internet at the time of Inet's development. By default, the fraction of degree 1 nodes α is set to 0.3, based on measurements from Routeviews[6] and NLANR[7] BGP table data in 2002.

4.6 PFP

In the Positive Feedback Preference (PFP) model [28], the AS topology of the Internet is considered to grow by interactive probabilistic addition of new nodes and links. It uses a nonlinear preferential attachment probability when choosing older nodes for the interactive growth of the network, inserting edges between existing and newly added nodes. As the PFP generator does not have any user-tunable parameters we include it only in the last part of Section 5 for completeness.

4.7 Observed Topology

Our observed topology dataset comes from the CAIDA Skitter project.[8] CAIDA computes the adjacency matrix of the AS topology from the daily Skitter measurements. These are obtained by running traceroutes over a large range of IP addresses and mapping the prefixes to AS numbers using RouteViews BGP data. Since the Skitter data represents paths that have actually been traversed by packets to their destinations, rather than paths calculated and propagated by BGP system, it is more likely to faithfully represent the IP topology than the BGP data alone. For our study, we used the graphs for March 2004 as used by Mahadevan *et al.* [18]. This dataset reports 9,204 unique ASes across the Internet.

[6] `http://www.routeviews.org/`

[7] `http://www.nlanr.net/`

[8] `http://www.caida.org/tools/measurement/Skitter/`

5 Results

The aim of this section is to examine how well the topology generators match the Skitter topology for different values of their parameters. To facilitate this comparison, grids are constructed over the possible values of the parameter spaces and various cost functions are evaluated as follows:

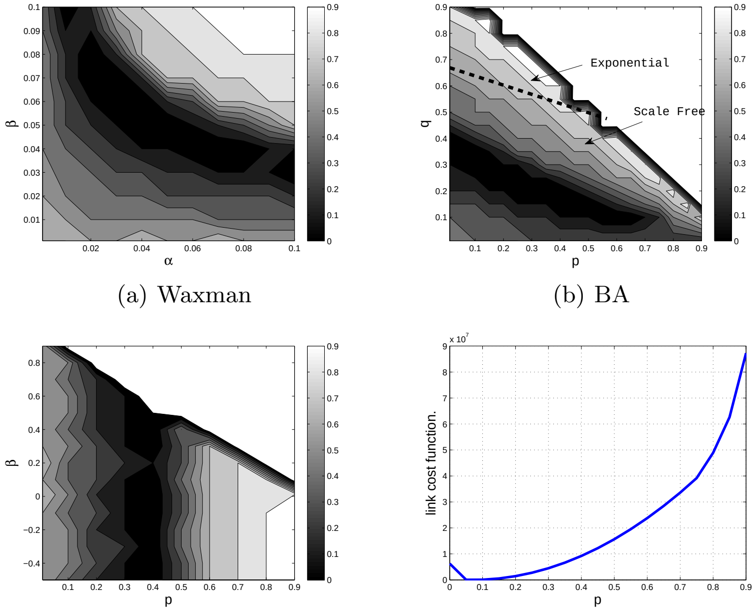
1. A cost function measuring the matching between the number of links in skitter and the generated topologies:
$$C_1(\theta) = (l_t(\theta) - l_{skitter})^2 \tag{26}$$
where C_1 is the first cost function, θ are the model parameters (which differ for each topology generator), l_t is the number of links (which is a function of the parameters) and $l_{skitter}$ is the number of links in the Skitter dataset.
2. A cost function measuring the matching between the spectra of the Skitter network and of the generated topologies:
$$C_2(\theta) = \sum_i (P(\Lambda \leq \lambda_{t,i}) - P(\Lambda \leq \lambda_{skitter,i}))^2 \tag{27}$$
where $\lambda_{t,i}$ is the i^{th} eigenvalue for topology t.
3. A cost function measuring the matching of the weighted spectra:
$$C_3(\theta) = \sum_i ((w * P(\Lambda = \lambda_{t,i}) - w * P(\Lambda = \lambda_{skitter,i}))^2 \tag{28}$$
where weight $w = (1 - i)^4$.

In addition to examining different parameter values across a grid, the optimum parameters with respect to $C_3(\theta)$ are estimated using the Nelder Meade simplex search algorithm [21, 9]. Note that the topologies generated by the topology generators are random in a statistical sense, due to differing random seeds for each run. Ten topologies are generated for each value of θ and the average spectral distribution is calculated. We found that the variance of the spectral distributions was sufficiently low to allow reasonable estimates of the minima in each case.

5.1 Link Densities

Figure 1 displays the value of the cost function $C_1(\theta)$ as a function of the topology generator parameters. On the upper and lower left graphs, the grayscale color indicates the value of the cost function. For Inet (lower right) there is only one parameter, p, so it is plotted as a curve in Figure 1(d). Figure 1 shows that a minimum exists for each topology in approximately the same regions as the default values of each generator.[9] For the BA generator it is known that for

[9] Some of these default values are listed in table 1.



(a) Waxman

(b) BA

(c) GLP

(d) Inet

Fig. 1. Topology generator parameter grid for sum squared error from number of links.

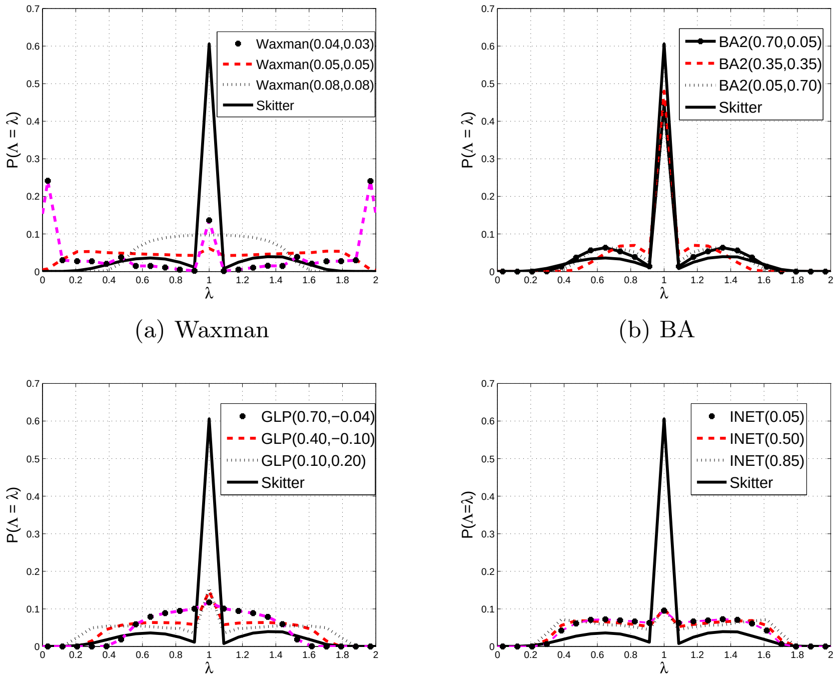
values of p and q above the line shown in Figure 1(b), the topologies generated follow an exponential node degree distribution while those below follow a scale-free distribution. It is encouraging to note that the values of $C_1(\theta)$ are large in the exponential region and the minimum is in the scale-free region as the node degree distribution of the Internet is known to be approximately scale free [2]. Overall the results obtained by tuning the parameters based on $C_1(\theta)$ appear reasonable. For link density matching it is possible to obtain parameter values which match the link densities exactly. Indeed, there is a ridge of parameters for BA, GLP and Waxman for which the link densities can be matched. However, as noted in the introduction, there is no control over any other characteristic of the graph using this method.

5.2 Spectra PDF

Figure 2 shows the spectral PDF of the Skitter dataset and the four topology generators calculated at three parameters values in each grid (the parameter values are indicated in brackets in the legends). The aim is to illustrate how much the spectral PDFs change with the values of the parameters. The spectral PDFs of Waxman (Figure 2(a)) vary significantly for different values of α and β. Furthermore, none of the Waxman PDFs match well the spectral PDF of the Skitter graph. The BA PDFs vary to a lesser extent (Figure 2(b)) and appear to give a much better match than the Waxman model, especially around eigenvalue 1 ($\lambda = 1$). This better match of BA is not surprising as the Waxman model is not a good model for the Internet as noted in Section 4. GLP (Figure 2(c)) and Inet (Figure 2(d)) give similar results to BA, with a poor match outside eigenvalue 1. The better match of the BA model around eigenvalue 1 is interesting. As noted in Section 3 the regions away from eigenvalue 1 are far more important than the region around $\lambda = 1$. However, what is required is a technique that reveals the differences with distance from one as these are more important. Thus it would appear difficult to evaluate which model, or even which parameter, is better based on the PDFs alone. This point is now further explored by analysis of the grids calculated with respect to $C_2(\theta)$.

5.3 Limitations of Spectra CDF

Figure 3 shows the value of the second cost function $C_2(\theta)$ as a function of the topology generator parameters, in the same way as Figure 1. As can be seen in Figure 3, there are many islands corresponding to local minima, creating a rugged landscape. The variance in the PDFs referred to in this section is actually greater than any gradient that might exist in the grid. This means that it is not



(a) Waxman

(b) BA

(c) GLP

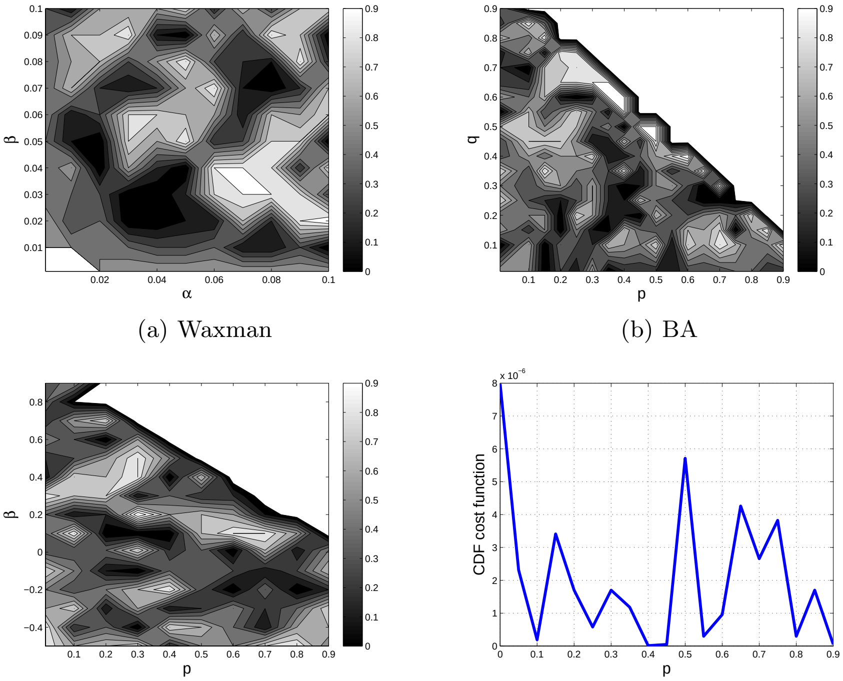
(d) Inet

Fig. 2. PDF of Spectra

possible to estimate the minimum with respect to $C_2(\theta)$. Figure 3 shows that the spectrum on its own is not sufficient to identify the optimum parameters of any of the topology generators. This is because each eigenvalue in $C_2(\theta)$ is weighted equally. As noted in Section 3, the eigenvalues close to 1 are more likely to be affected by the random seeds for each topology generator and are the source of the noise on the grid.

5.4 Weighted Spectra

The previous section illustrated the limitations of using the raw eigenvalues to find optimal topology generator parameters to match the Skitter topology. Figure 4 shows a plot of the weighted spectra of the same topologies as those shown on Figure 2. As can be seen the results are quite different from those shown in Figure 2. The Waxman weighted spectra still shows a bad fit with respect to the Skitter data (mainly around 0 and 2) compared to the other generators. The other generators (BA, GLP and Inet) now show that they are capable of matching the weighted spectra of the Skitter topology, especially around the point of greatest weight ($\lambda = 0.4$ or 1.6). The difference between the weighted spectra around 1 is no longer of importance (in contrast to Figure 2), reflecting



(a) Waxman

(b) BA

(c) GLP

(d) Inet

Fig. 3. Parameter grid for sum of absolute differences of spectra CDFs.

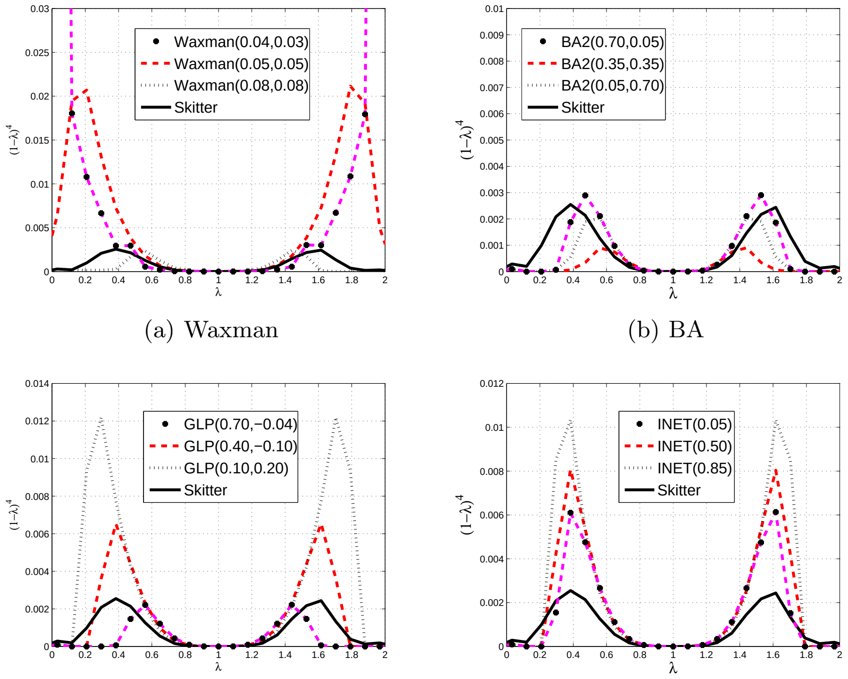
that the weights here approach zero as we approach eigenvalue 1. In the next section the optimum values and the resulting weighted spectra will be compared.

5.5 Weighted Spectra Comparison

Figure 5 shows the grids associated with $C_3(\theta)$. As can be seen the grids show that there is a region with a minima in each case and in addition, comparing Figure 5 and Figure 1 it can be seen that these minima lie in a region close to those for $C_1(\theta)$. However, it should be noted that the weighted spectra will try to fit more than just the number of links in a topology. This demonstrates the inherent trade-off. Also of note is that the region of interest for the BA model lies inside the region of scale-free behaviour as shown in Figure 5(b).

6 Generating Topologies with Optimum Value Parameters

Table 1 displays the optimum values for the topology generators for generating networks that are close to the Skitter graph. In addition, we give the values for



(a) Waxman

(b) BA

(c) GLP

(d) Inet

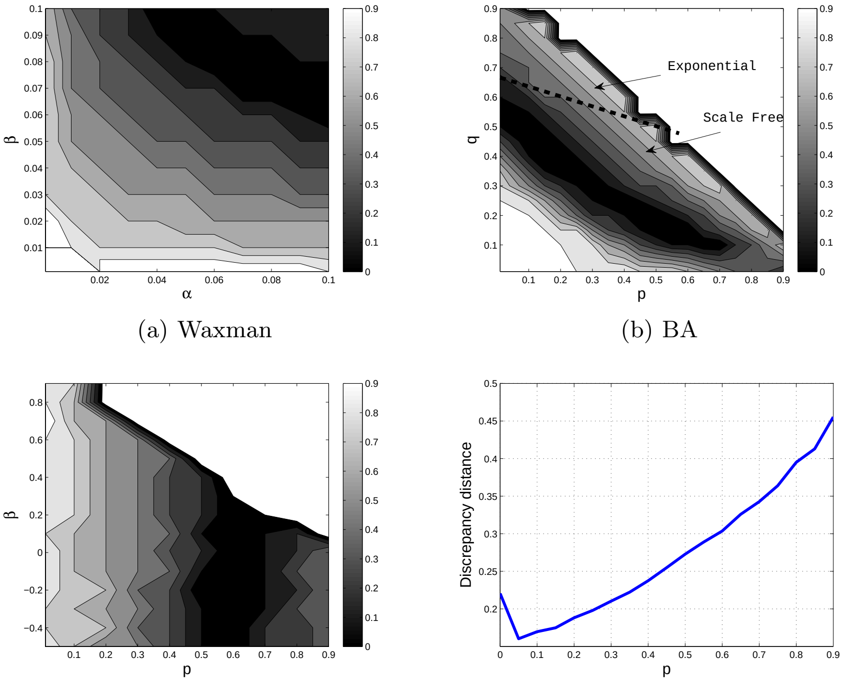
Fig. 4. Weighted spectra grid for generator parameters.

$C_3(\theta)$, which show that PFP gives the closest fit followed by BA, GLP, Waxman and finally Inet. While these results are mostly expected, the ranking of Inet as the worst topology generator is surprising. We have also listed some of the default parameters used in certain generators such as BRITE [19]. While many of the optimised parameters are close to the default values, which is encouraging, it should be noted that the default parameters are for a *typical* graph and are not selected for any particular situation. Thus a direct comparison is meaningless.

Table 1. Optimum parameter values for matching Skitter topology.

Waxman	$\alpha = 0.08$ (default= 0.15)	$\beta = 0.08$ (default= -0.2)	$C_3(\theta) = 0.0026$	$\overline{C_3}(\theta) = 0.0797$
BA	$p = 0.2865$ (default= 0.6)	$q = 0.3145$ (default= 0.3)	$C_3(\theta) = 0.0014$	$\overline{C_3}(\theta) = 0.0300$
GLP	$p = 0.5972$ (default= 0.45)	$\beta = 0.1004$ (default= 0.64)	$C_3(\theta) = 0.0021$	$\overline{C_3}(\theta) = 0.0446$
Inet	$\alpha = 0.1013$ (default= 0.3)	$-$	$C_3(\theta) = 0.0064$	$\overline{C_3}(\theta) = 0.0150$
PFP	$-$	$-$	$C_3(\theta) = 0.0014$	$\overline{C_3}(\theta) = 0.0371$

Figure 6(a) shows the weighted spectra for each of the topology generators and inspection of this figure goes some way to explaining the discrepancy in the



(a) Waxman

(b) BA

(c) GLP

(d) Inet

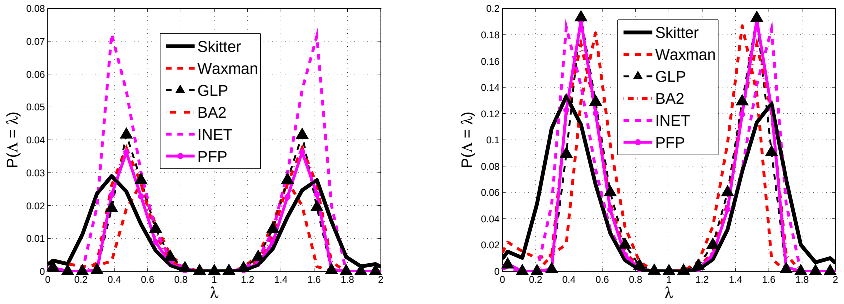
Fig. 5. Grid of sum squared error of weighted spectra for topology generators

results. As can be seen the main peak in the weighted spectra for the Skitter data occurs at a value of $\lambda = 0.4$. The Waxman generator peak occurs at $\lambda = 0.6$ which is closer to 1 demonstrating the greater amount of random structure in the Waxman topologies. However, for the Inet generator the peak occurs at the correct point (λ =0.4) but the weighted power at this point is far greater than in the skitter topology. By normalizing the weighted spectrum this point becomes clear:

$$\overline{C_3}(\theta) = \sum_i \frac{((w_i * P(\mathit{\Lambda} = \lambda_{t,i}))}{\sum_i ((w_i * P(\mathit{\Lambda} = \lambda_{t,i}))} - \frac{((w_i * P(\mathit{\Lambda} = \lambda_{skitter}))}{\sum_i ((w_i * P(\mathit{\Lambda} = \lambda_{skitter}))} \qquad (29)$$

Using the normalised weighted spectrum the results in Figure 6(b) show that Inet is the best match for the Skitter data while the Waxman model still performs worse than the other models. Further research is required before stating which version of C_3 is superior.

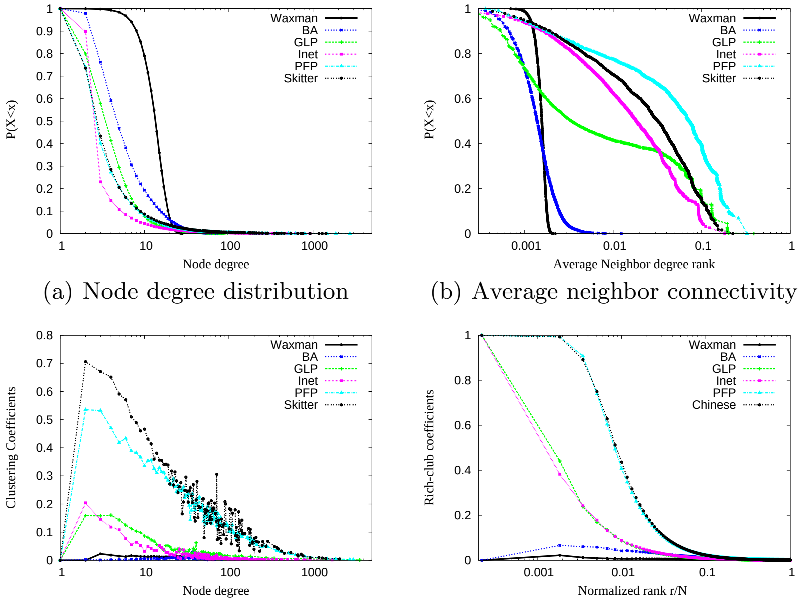
Figure 7 shows a comparison of the optimized topologies with respect to four typical network metrics: the node degree distribution, the average neighbor connectivity, the clustering coefficient and the rich-club connectivity [28]. As can be seen PFP gives the best match for these metrics in agreement with our proposed metric $C_3(\theta)$. The performance of the other topologies is mixed showing that while one topology is able to match one metric it fails to match another. For example, the GLP generator achieves a reasonable match for the node degree distribution but fails to match the average neighbor connectivity. It is interesting to note that BA does not match the rich club connectivity which is not evident in our metric.



(a) Weighted spectra

(b) Normalized weighted spectra

Fig. 6. Comparison of the weighted spectra.



(a) Node degree distribution

(b) Average neighbor connectivity

(c) Clustering coefficients

(d) Rich-Club connectivity

Fig. 7. Comparison of topology generators and Skitter topology.

7 Conclusions

Comparison of graph structures is a frequently encountered problem across a number of problem domains. To perform a useful comparison requires definition of a cost function that encodes which features of the graphs are considered important. Although the spectrum of a graph is often claimed to be a way to encode a graph's features, the raw spectrum contains too much noise to be useful on its own. In this paper we have introduced a new cost function, the *weighted graph spectrum*, that improves on the graph spectrum by discounting those eigenvalues that are believed to be unimportant and emphasising the contribution of those believed to be important.

We use this cost function to optimise the selection of parameter values within the particular problem domain of Internet topology generation. The weighted spectrum was shown to be a useful cost function in that it leads to parameter choices that appear sensible given prior knowledge of the problem domain, i.e., are close to the default values and, in the case of the BA generator, fall within the expected region. In addition, as the metric is formed from a summation, it is possible to go further and identify which particular eigenvalues are responsible for significant differences. Although it is currently difficult to assign specific features to specific eigenvalues, it is hoped that this feature of our cost function will be useful in the future.

Acknowledgments

We would like to thank Andrew Thomason of Cambridge University for advice on comparison techniques and insight on use of graph deviation. We greatly acknowledge encouragement and advice of Richard Gibbens for pursuing this research. We also appreciate the comments and constructive feedback by anonymous reviewers. This work is conducted as part of the EPSRC UKLIGHT/MASTS project under grants GR/T10503/01 and GR/T10510/03.

References

1. W. Aiello, F. Chung, and L. Lu. A random graph model for massive graphs. In *STOC'00: Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pages 171–180, Portland, OR, May (2000).
2. R. Albert and A.-L. Barabasi. Topology of evolving networks: local events and universality. *Physical Review Letters*, 85:5234, (2000).
3. A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, (1999).
4. T. Bu and D. Towsley. On distinguishing between Internet power law topology generators. In *Proceedings of IEEE Infocom 2002*, New York, NY, June (2002).
5. S. Butler. Lecture notes for spectral graph theory. *Lectures in Nankai University, Tianjin, China*, (2006).
6. K. L. Calvert, M. B. Doar, and E. W. Zegura. Modeling Internet topology. *IEEE Communications Magazine*, 35(6):160–163, (1997).
7. F. Chung and R. Graham. Quasi-random graphs with given degree sequences. *Random Struct. Algorithms*, 32(1):1–19, (2008).
8. F. R. K. Chung. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics)*. American Mathematical Society, (1997).
9. J. Dennis and D. Woods. Optimization in microcomputers: The nelder-meade simplex algorithm. In A. Wouk, editor, *New Computing Environments: Microcomputers in Large-Scale Computing*, pages 116–122. SIAM, (1987).
10. M. B. Doar. A better model for generating test networks. In *IEEE GLOBECOM'96*, London, UK, Nov. (1996).
11. P. Erdös and A. Rényi. On random graphs. In *Mathematical Institute Hungarian Academy, 196*, London, (1985).
12. M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the Internet topology. In *Proceedings of ACM SIGCOMM 1999*, pages 251–262, Cambridge, Massachusetts, United States, (1999).
13. C. Gkantsidis, M. Mihail, and E. Zegura. Spectral analysis of Internet topologies. In *Proceedings of IEEE Infocom 2003*, San Francisco, CA, Apr. (2003).
14. S. Hanna. Representation and generation of plans using graph spectra. In *6th International Space Syntax Symposium*, Istanbul, (2007).
15. O. Heckmann, M. Piringer, J. Schmitt, and R. Steinmetz. On realistic network topologies for simulation. In *MoMeTools '03: Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research*, pages 28–32, New York, NY, USA, (2003).
16. A. Jamakovic and S. Uhlig. On the relationship between the algebraic connectivity and graph's robustness to node and link failures. In *Next Generation Internet Networks, 3rd EuroNGI Conference on*, Trondheim, Norway, (2007).

17. U. Luxburg, O. Bousquet, and M. Belkin. Limits of spectral clustering. In *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA., (2005).
18. P. Mahadevan, D. Krioukov, M. Fomenkov, X. Dimitropoulos, k c claffy, and A. Vahdat. The Internet AS-level topology: three data sources and one definitive metric. *SIGCOMM Computer Communication Review*, 36(1):17–26, (2006).
19. A. Medina, A. Lakhina, I. Matta, and J. Byers. BRITE: an approach to universal topology generation. In *IEEE MASCOTS*, pages 346–353, Cincinnati, OH, USA, Aug. (2001).
20. B. Nadler, S. Lafon, R. Coifman, and I. Kevrekidis. Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators. In *Neural Information Processing Systems (NIPS)*, (2005).
21. J. Nelder and R. Mead. A simplex method for function minimization. *Comput. J.*, 7:308–313, (1965).
22. A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*. MIT Press, (2002).
23. H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger. Network topology generators: degree-based vs. structural. In *Proceedings of ACM SIGCOMM 2002*, pages 147–159, Pittsburgh, PA, (2002).
24. D. Vukadinovic, P. Huang, and T. Erlebach. On the spectrum and structure of Internet topology graphs. In *IICS '02: Proceedings of the Second International Workshop on Innovative Internet Computing Systems*, (2002).
25. B. M. Waxman. Routing of multipoint connections. *IEEE Journal on Selected Areas in Communications (JSAC)*, 6(9):1617–1622, Dec. (1988).
26. J. Winick and S. Jamin. Inet-3.0: Internet topology generator. Technical Report CSE-TR-456-02, University of Michigan Technical Report CSE-TR-456-02, (2002).
27. E. W. Zegura, K. L. Calvert, and M. J. Donahoo. A quantitative comparison of graph-based models for Internet topology. *IEEE/ACM Transactions on Networking (TON)*, 5(6):770–783, (1997).
28. S. Zhou. Characterising and modelling the Internet topology, the rich-club phenomenon and the PFP model. *BT Technology Journal*, 24, (2006).