Technical Report

Number 729

UNIVERSITY OF CAMBRIDGE

Computer Laboratory

Weighted spectral distribution

Damien Fay, Hamed Haddadi, Steve Uhlig,
Andrew W. Moore, Richard Mortier,
Almerima Jamakovic

September 2008

Abstract

Comparison of graph structures is a frequently encountered problem across a number of problem domains. Comparing graphs requires a metric to discriminate which features of the graphs are considered important. The spectrum of a graph is often claimed to contain all the information within a graph, but the raw spectrum contains too much information to be directly used as a useful metric. In this paper we introduce a metric, the *weighted spectral distribution*, that improves on the raw spectrum by discounting those eigenvalues believed to be unimportant and emphasizing the contribution of those believed to be important.

We use this metric to optimize the selection of parameter values for generating Internet topologies. Our metric leads to parameter choices that appear sensible given prior knowledge of the problem domain: the resulting choices are close to the default values of the topology generators and, in the case of the AB generator, fall within the expected region. This metric provides a means for meaningfully optimizing parameter selection when generating topologies intended to share structure with, but not match exactly, measured graphs.

1   Introduction

In this paper we present a metric, *weighted spectral distribution*, for comparing graphs based on the distribution of their internal structure. Graph comparison is a problem that occurs in many branches of computing, from vision to speech processing to systems. The metric we present differs from existing graph-matching techniques which seek to identify graphs which share common *clusters*, i.e., are similar. Instead, our metric is designed specifically for situations where the graphs being compared are in general dis-similar but can be expected to share, in some sense, a common *structure*. For example, when generating synthetic workloads from trace data described as a graph, the generated graphs should *not* match the original trace data exactly but *should* share some common structure with them. Situations where this is encountered include workload generation, e.g., as in Magpie [2], and Internet topology generation.

We specifically focus on the latter problem domain in this paper, addressing the problem of generating synthetic topologies designed to mimic the structure of the Internet. The Internet topology's structure is not easy to characterize. In the core there is a full mesh formed between various tier-1 Internet Service Providers (ISPs). However, at the edges there are a huge number of smaller ISPs and customer networks which connect through upstream providers. These smaller ISPs and customer networks may have only one upstream provider, or may have many for resilience and performance reasons. This rich and varied structure makes it difficult for researchers to provide a single model, and hence a single metric, that captures all the characteristics of various topologies. Many attempts to capture one or even several characteristics have been made, resulting in several topology generators which each synthesize Internet-like topologies using different models and parameters. Unfortunately, little or no guidance is available on how to set these parameters, with the default values subjectively chosen by the original authors usually being used.

Using our metric, we compare five different topology generators and a measurement of the existing Internet's AS (Autonomous System) topology. Empirical evidence from

this comparison shows that the weighted spectral distribution is consistent with expected results. Using our metric we are also able to give optimum parameter settings for these topology generators with respect to the measured AS data and the weighted spectral distribution.

In summary, we present three contributions in this paper: (*i*) a metric for comparing the structure of graphs that reveals important characteristics such as in what way two graphs differ; (*ii*) a comparison of the outputs of five major Internet topology generators and a measured dataset; (*iii*) optimal parameterizations under our metric of these topology generators with respect to the measured dataset.

2   Background

Graph analysis is typically concerned with determining the relationships among the vertices of a graph, and its various applications can be broadly classified as topology tuning, graph matching and cluster determination, examples of which will now be discussed.

Topology tuning is the problem domain we are concerned with here, namely adjusting parameters used to generate topologies with the aim of generating topologies "close enough" to some representative. Graph spectra have not been used for this purpose before, although Hanna [8] uses graph spectra for numerical comparison of architectural spaces in large building plans. By defining space as a graph, they show that the spectra of two plan types can be effectively used to judge the effects of global *vs.* local changes, and hence the edit distances, to the plans. Hanna believes spectra are a reliable metric for capturing the local relationships and can be used to guide optimization algorithms for reproducing plans.

Alternatively, graph matching is concerned with the comparison of two or more graphs to determine which clusters in the graphs are related. For example, Luo and Hancock [10] compare several images of an object taken at different angles and determine the correspondence between them by representing the images as graphs. Their technique seeks to maximize the likelihood that groups of vertices in one graph correspond to those in another. However, the aim of this work differs from ours as it seeks to identify the edges between specific nodes and clusters.

Techniques for cluster identification differ mainly in the choice of matrix associated with a graph. Ng *et al.* [12] present an algorithm for determining the dominant clusters in a graph by examining the eigenpairs of the normalized graph Laplacian. They suggest that a graph with n-vertices may be represented in $k < n$-dimensions by choosing the first k eigenvectors as ordered by the k smallest eigenvalues. The value of k is determined by examining the eigenvalues of L, denoted by $\lambda_i, i = 0, \ldots, n-1$. The eigenvector associated with the first (non-zero) eigenvalue determines the largest clusters in the graph with subsequent eigenvectors determining finer subclusters. If there exists an eigenvalues λ_i, for which $\mathrm{abs}(1-\lambda_{i+1}) >> \mathrm{abs}(1-\lambda_i)$ then λ_{i+1} to λ_{n-1} are clustered around 1, and this is known as the spectral gap. The spectral gap represents eigenvalues which have little power and can be thought of as representing noise or small variations in the graph structure. Typically, the value of k is determined by examination of a plot of the eigenvalues and set by the spectral gap. The $n \times k$ matrix of eigenvectors is then clustered using the k-means clustering algorithm, although variations using other clustering algorithms exist.

Graph spectra have also been used for characterization of Internet topologies. Gkantsidis *et al.* [6] perform a comparison of clustering coefficients by using the eigenvectors of the k largest eigenvalues of adjacency matrices on BGP topology graphs. However, the choice of k is arbitrary and those chosen are given equal importance. They consider the rest of the spectrum as noise, despite works that have shown that the eigenvalues of the adjacency matrix or the normalized Laplacian matrix can be used to accurately represent a topology and some specific eigenvalues a measure of properties such as robustness of a network to failures [15, 9].

Vukadinovic *et al.* [17] use the normalized Laplacian spectrum for analysis of AS graphs, proposing it as a fingerprint for Internet-like graphs. Using the Inet [19] generator and AS graphs extracted from BGP data, they obtain eigenvalues of the normalized Laplacian matrix. They believe that the graph spectrum should be considered as 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 the structural differences between two topologies.

We now present our metric, the weighted spectral distribution, before using it to compare synthetic and measured topologies, and to optimize parameter selection for the topology generators with respect to the measured topology.

3 Methodology

We now derive our metric, the *weighted spectral distribution*, relating it to another common structural metric, the clustering coefficient, before showing how it characterises networks with different mixing properties.

Denote an undirected graph as $G = (V, E)$ where V is the set of vertices (nodes) and E is the set of edges (links). The adjacency matrix of G, $A(G)$, has an entry of one if two nodes, u and v, are connected and zero otherwise

$$A(G)(u,v) = \begin{cases} 1, & \text{if } u, v \text{ are connected} \\ 0, & \text{if } u, v \text{ are not connected} \end{cases} \tag{1}$$

Let d_v be the degree of node v and $D = diag(sum(A))$ be the diagonal matrix having the degrees along its diagonal. Denoting by I the identity matrix $(I)_{i,j} = 1$ if $i = j, 0$ otherwise, the Normalised Laplacian L associated with graph G is constructed from A by normalising the entries of A by the node degrees of A as

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

or equivalently

$$L(G)(u,v) = \begin{cases} 1, & \text{if } u = v \text{ and } d_v \neq 0 \\ -\dfrac{1}{\sqrt{d_u d_v}}, & \text{if } u \text{ and } v \text{ are adjacent} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

As L is a real symmetric matrix there is an orthonormal basis of real eigenvectors $e_0, \ldots, e_{n-1}$ (i.e., $e_i e_j^T = 0, i \neq j$ and $e_i e_i^T = 1$) with associated eigenvalues $\lambda_0, \ldots, \lambda_{n-1}$.

It is convenient to label these so that $\lambda_0 \le \ldots \le \lambda_{n-1}$. The set of pairs (eigenvectors and eigenvalues of L) is called the spectrum of the graph. It can be seen that

$$L(G) = \sum_i \lambda_i e_i e_i^T \tag{4}$$

The eigenvalues $\lambda_0, \ldots, \lambda_{n-1}$ represent the strength of projection of the matrix onto the basis elements. This may be viewed from a statistical point of view [14] where each $\lambda_i e_i e_i^T$ may be used to approximate $A(G)$ with approximation error inversely proportional to $1 - \lambda_i$. However, for a graph, those nodes which are best approximated by $\lambda_i e_i e_i^T$ in fact form a cluster of nodes. This is the basis for spectral clustering, a technique which uses the eigenvectors of L to perform clustering of a dataset or graph [11]. The first (smallest) non-zero 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 [13]. Typically, there exists what is called a *spectral gap* in which for some k and j, $\lambda_k \ll \lambda_{k+1} \approx 1 \approx \lambda_{j-1} \ll \lambda_j$. That is, eigenvalues $\lambda_{k+1}, \ldots, \lambda_{j-1}$[1] are approximately equal to one and are likely to represent noise in the original dataset, i.e., links in a graph which do not belong to any particular cluster. It is then usual to reduce the dimensionality of the data using an approximation based on the spectral decomposition. However, in this paper we are interested in representing the global structure of a graph (e.g., we are interested in the presence or absence of many small clusters), which is essentially the spread of clustering across the graph. This information is contained in all the eigenvalues of the spectral decomposition.

Let $x = (x_0, \ldots, x_{n-1})$ be a vector. From (3) we see that

$$xLx^T = \sum_{uv \in E} (x_u/\sqrt{d_u} - x_v/\sqrt{d_v})^2 \tag{5}$$

Now, the eigenvalues cannot be large because from (5) we obtain

$$xLx^T \le \sum_{uv \in E} (x_u/\sqrt{d_u} - x_v/\sqrt{d_v})^2$$
$$+ (x_u/\sqrt{d_u} + x_v/\sqrt{d_v})^2$$
$$= 2 \sum_u x_u^2 = 2xx^T \tag{6}$$

and so $\lambda_i = e_i L e_i^T \le 2$. What is more, the mean of the eigenvalues is 1 because

$$\sum_i \lambda_i = \operatorname{tr}(L) = n \tag{7}$$

by (3), where $\operatorname{tr}(L)$ is the *trace* of L.

To summarize: the eigenvalues of L lie in the range 0 to 2 (the smallest being 0), i.e., $0 = \lambda_0 \le \ldots \le \lambda_{n-1} \le 2$, and their mean is 1.

The distribution of the n numbers $\lambda_0, \ldots, \lambda_{n-1}$ contains useful information about the network, as will be seen. In turn, information about this distribution is given by

---

[1]i.e., the eigenvalues at the centre of the spectrum.

its moments in the statistical sense, where the Nth moment is $1/n \sum_i (1 - \lambda_i)^N$. These moments have a direct physical interpretation in terms of the network, as follows. Writing B for the matrix $D^{-1/2}AD^{-1/2}$, so that $L = I - B$, then by (3) the entries of B are given by

$$(D^{-1/2}AD^{-1/2})_{i,j} = \frac{A_{i,j}}{\sqrt{d_i}\sqrt{d_j}} \tag{8}$$

Now the numbers $1 - \lambda_i$ are the eigenvalues of $B = I - L$, and so $\sum_i (1 - \lambda_i)^N$ is just $\text{tr}(B^N)$. Writing $b_{i,j}$ for the (i,j)-th entry of B, the (i,j)-th entry of B^N is the sum of all products $b_{i_0,i_1} b_{i_1,i_2} \ldots b_{i_{N-1} i_N}$ where $i_0 = i$ and $i_N = j$. But $b_{i,j}$, as given by (8), is zero unless nodes i and j are adjacent. So we define an N-cycle in G to be a sequence of vertices $u_1 u_2 \ldots u_N$ with u_i adjacent to u_{i+1} for $i = 1, \ldots, N - 1$ and with u_N adjacent to u_1. (Thus, for example, a triangle in G with vertices set $\{a, b, c\}$ gives rise to six 3-cycles abc, acb, bca, bac, cab and cba. Note that, in general, an N-cycle might have repeated vertices.) We now have

$$\sum_i (1 - \lambda_i)^N = \text{tr}(B^N) = \sum_C \frac{1}{d_{u_1} d_{u_2} \ldots d_{u_N}} \tag{9}$$

the sum being over all N-cycles $C = u_1 u_2 \ldots u_N$ in G. Therefore, $\sum_i (1 - \lambda_i)^N$ counts the number of N-cycles, normalised by the degree of each node in the cycle.

The number of N-cycles is related to various graph properties. The number of 2-cycles is just (twice) the number of edges and the number of 3-cycles is (six times) the number of triangles. Hence $\sum_i (1 - \lambda)^3$ is related to the clustering coefficient, as discussed below. An important graph property is the number of 4-cycles. A graph which has the minimum number of 4-cycles, for a graph of its density, is quasi-random, i.e., it shares many of the properties of random graphs, including, typically, high connectivity, low diameter, having edges distributed uniformly through the graph, and so on. This statement is made precise in [16] and [5]. For regular graphs, (9) shows that the sum $\sum_i (1 - \lambda)^4$ is directly to the number of 4-cycles. In general, the sum counts the 4-cycles with weights: for the relationship between the sum and the quasi-randomness of the graph in the non-regular case, see the more detailed discussion in [4, Chapter 5]. The right hand side of (9) can also be seen in terms of random walks. A random walk starting at a vertex with degree d_u will choose an edge with probability $1/d_u$ and at the next vertex, say v, choose an edge with probability $1/d_v$ and so on. Thus the probability of starting and ending randomly at a vertex after N steps is the sum of the probabilities of all N-cycles that start and end at that vertex. In other words exactly the right hand side of (9). As pointed out in [18], random walks are an integral part of the Internet AS structure.

The left hand side of Equation (9) provides an interesting insight into graph structure. The right hand side is the sum of normalised N-cycles whereas the left hand side involves the spectral decomposition. We note in particular that the spectral gap is diminished because eigenvalues close to one are given a very low weighting compared to eigenvalues far from one. This is important as the eigenvalues in the spectral gap typically represent "random" links in the network and are not therefore important parts of the larger structure of the network.

Next, we consider the well-known clustering coefficient. It should be noted that there is little connection between the clustering coefficient, and cluster identification, referred

to above. The clustering coefficient, $\gamma(G)$, is defined as the average number of triangles divided by the total number of possible triangles

$$\gamma(G) = 1/n \sum_i \frac{T_i}{d_i(d_i - 1)/2}, d_i \geq 2 \tag{10}$$

where T_i is the number of triangles for node i and d_i is the degree of node i. Now consider a specific triangle between nodes a, b and c. For the clustering coefficient, noting that the triangle will be considered three times, once from each node, the contribution to the average is

$$\frac{1}{d_a(d_a - 1)/2} + \frac{1}{d_b(d_b - 1)/2} + \frac{1}{d_c(d_c - 1)/2} \tag{11}$$

However, for the weighted spectrum (with $N = 3$), this particular triangle gives rise to six 3-cycles and contributes

$$\frac{6}{d_a d_b d_c} \tag{12}$$

So, it can be seen that the clustering coefficient normalises each triangle according to the total number of possible triangles while the weighted spectrum (with $N = 3$) instead normalises using a product of the degrees. Thus, the two metrics can be considered to be similar but not equal. Indeed, it should be noted that the clustering coefficient is in fact not a metric in the strict sense. While two networks can have the same clustering coefficient they may differ significantly in structure. In contrast, the elements of $\sum_i (1 - \lambda)^3$ will only agree if two networks are isomorphic.

We now formally define the *weighted spectrum* as the normalised sum of N-cycles as

$$W(G, N) = \sum_i (1 - \lambda_i)^N \tag{13}$$

However, calculating the eigenvalues of a large (even sparse) matrix is computationally expensive. In addition, the aim here is to represent the *global* structure of a graph and so precise estimates of *all* the eigenvalue values are not required. Thus, the distribution[2] of eigenvalues is sufficient. In this paper the distribution of eigenvalues $f(\lambda = k)$ is estimated using pivoting and Sylvester's Law of Inertia to compute the number of eigenvalues that fall in a given interval. A measure of the graph can then be constructed by considering the distribution of the eigenvalues as

$$\omega(G, N) = \sum_{k \in K} (1 - k)^N f(\lambda = k) \tag{14}$$

where the elements of $\omega(G, N)$ form the *weighted spectral distribution*:

$$WSD : G \rightarrow \Re^{|K|} \{ k \in K : ((1 - k)^N f(\lambda = k)) \} \tag{15}$$

In addition, a metric can then be constructed from $\omega(G)$ for comparing two graphs, G_1 and G_2, as

$$\Im(G_1, G_2, N) = \sum_{k \in K} (1 - k)^N (f_1(\lambda = k) - f_2(\lambda = k))^2 \tag{16}$$

---

[2]The eigenvalues of a given graph are deterministic and so *distribution* here is not meant in a statistical sense.
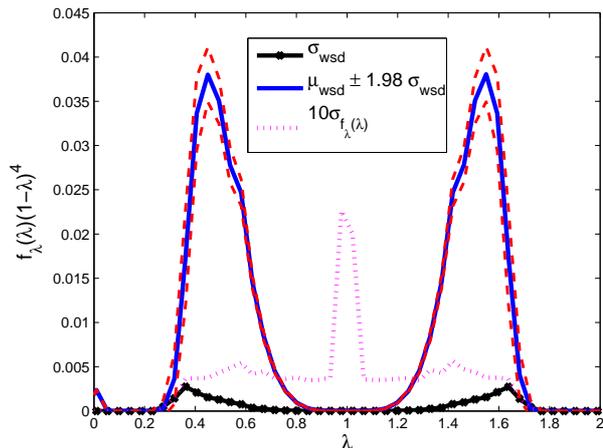
Figure 1: Mean and standard deviations for WSD and spectrum for the AB model over 50 simulations.

where f_1 and f_2 are the eigenvalue distributions of G_1 and G_2 and the distribution of eigenvalues is estimated in the set K of bins $\in [0, 2]$. Equation (16) satisfies all the properties of a metric.

We consider 3 and 4 to be suitable values of N for the current application: $N = 3$ is related to the well-known and understood clustering co-efficient; and $N = 4$ as a 4-cycle represents two routes (i.e., minimal redundancy) between two nodes. For other applications, other values of N may be of interest.

The statistical properties of the WSD are examined empirically in Figure 1. This plot was created by generating 50 topologies using the AB [1] generator with the (fixed) optimum parameters determined in Section 4, but with different initial conditions. For each run the spectral and weighted spectral distributions are recorded yielding 50×50 bin values which are then used to estimate standard deviations. As the underlying model (i.e. the AB generator) is the same for each run, the *structure* might be expected to remain the same and so a "structural metric" should be insensitive to random initial conditions. As can be seen the standard deviation[3] of the (unweighted) spectrum, $\sigma_{f_\lambda}(\lambda)$, is significantly higher at the centre of the spectrum, reflecting that the spectral gap contains random connections. However, for the WSD, the standard deviation, σ_{wsd}, peaks at the same point as the WSD; the noise in the spectral gap has been suppressed.

4 Results and discussions

In this section we minimise the cost function defined in Equation 16 to obtain parameter estimates for four Internet topology generators [7]: the Waxman model, the Albert and Barabasi Model (AB), the Generalized Linear Preference model (GLP) and the Inet model. These are compared with the Skitter dataset [3] and the Positive Feedback Preference model (PFP), which has no parameters. In addition, we show that the spectrum on its own, equivalent to $p = 1$, $\mu(\lambda) = 1$, is not sufficient to obtain parameter estimates.

---

[3]Multiplied by a factor of ten for clarity.

(a) Waxman

(b) AB

(c) GLP

(d) Inet
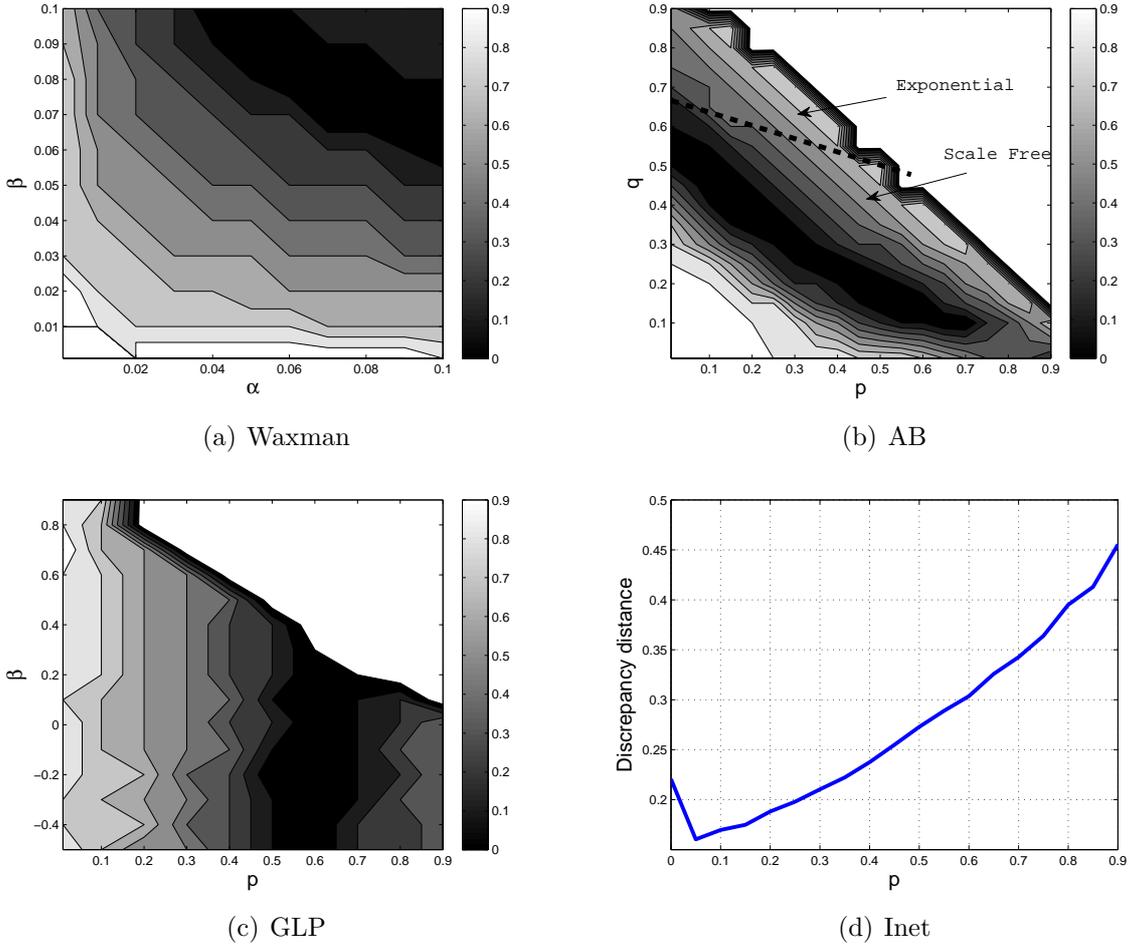
Figure 2: Grid of sum squared error, $\Im(G_1, G_2, 4)$, of weighted spectra for topology generators.

Figure 2 shows a plot of the weighted spectral distribution distance between the four topology generators and the Skitter dataset, as a function of values of the topology generator parameters. Each grid (Figure 2(d) is a curve as Inet has only one parameter) shows a quantile contour plot of the surface of the distances at different parameter values. It is encouraging to note that the minima in each case lie close to the default values (see Table 1). In addition, it is known that the behavior of the AB model splits into two regions: exponential behavior and scale free behavior. The Internet is known to exhibit scale free behavior, and the area of minimum distance lies in this area.

Figure 3 shows the weighted spectrum (each element that makes up the summation in Equation 15) for the optimal values of the parameters calculated using the Nelder Meade optimization algorithm. As can be seen the results are quite distinct, showing that no single topology is capable of generating the same distribution of structure as the Skitter data (it is in any case questionable that Skitter represents the actual Internet due to inherent measurement difficulties). However, it is the way in which each topology differs that is revealing. The Waxman model has peaks at 0.6, the closest peaks to 1 of all the topologies examined. This implies that the Waxman topologies exhibit more random behavior than desired, as expected. Alternatively the Inet topology correctly peaks at

10

Table 1: Optimum parameter values for matching Skitter topology.

Waxman	$\alpha = 0.08$ (default=0.15)	$\beta = 0.08$ (default=0.2)	$\Im_{min} = 0.0026$
AB	$p = 0.2865$ (default=0.6)	$q = 0.3145$ (default=0.3)	$\Im_{min} = 0.0014$
GLP	$p = 0.5972$ (default=0.45)	$\beta = 0.1004$ (default=0.64)	$\Im_{min} = 0.0021$
Inet	$\alpha = 0.1013$ (default=0.3)	−	$\Im_{min} = 0.0064$
PFP	−	−	$\Im_{min} = 0.0014$
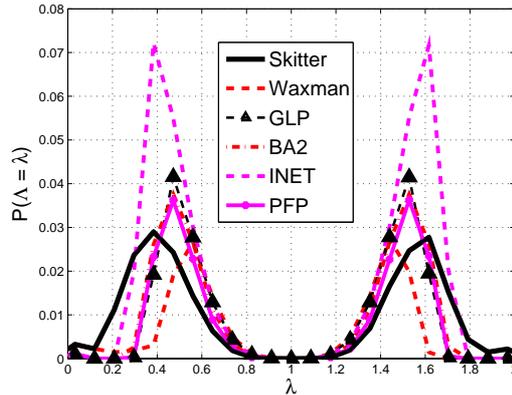
Figure 3: Comparison of the weighted spectra.

0.4, but exhibits too strong structure at this point. The best model is the PFP model which is a non-linear model considered to be a good approximation to the actual Internet. Table 1 confirms these results.
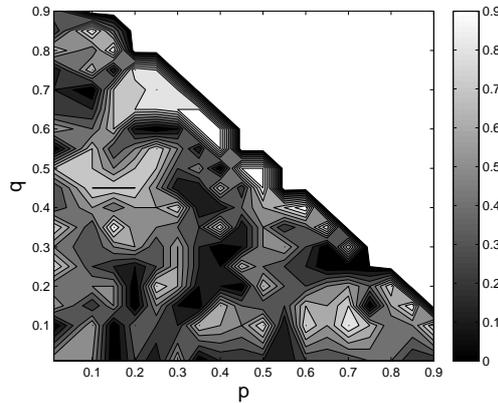
Figure 4: Parameter grid for sum of absolute differences of AB spectrum CDF.

As an example of using the unweighted spectrum, the grid for the AB model is shown in Figure 4. As can be seen there exists no minimum. This is because the unweighted spectrum weights each eigenvalue equally and so random structure, mainly in the spectral gap, makes the surface noisy. We show only the AB result here due to space constraints, but a similar situation was found for all of the topology generators. We thus conclude that the spectrum on its own is not sufficient to reveal the structure of the topologies.

5 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 metric 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 information to be useful on its own. In this paper we have introduced a new metric, the weighted spectral distribution, that improves on the graph spectrum by discounting those eigenvalues that are believed to be unimportant and emphasizing the contribution of those believed to be important.

We use this metric to optimize the selection of parameter values of Internet topology generation. The weighted spectral distribution was shown to be a useful metric in that it leads to parameter choices that appear sensible given prior knowledge of the problem domain: the resulting choices are close to the default values and, in the case of the AB 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 metric will be useful in the future.

Acknowledgments

References

[1] R. Albert and A.-L. Barabasi. Topology of evolving networks: local events and universality. *Physical Review Letters*, 85:5234, 2000.

[2] P. Barham, A. Donnelly, R. Isaacs, and R. Mortier. Using Magpie for request extraction and workload modelling. In *Proceedings of USENIX OSDI 2004*, pages 18–18, Berkeley, CA, USA, 2004. USENIX Association.

[3] CAIDA . Skitter Internet topology measurement tool.

[4] F. R. K. Chung. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics)*. American Mathematical Society, 1997.

[5] F. R. K. Chung, R. L. Graham, and R. M. Wilson. Pseudo-random graphs. *Combinatorica*, 9(4):345–362, 1989.

[6] C. Gkantsidis, M. Mihail, and E. Zegura. Spectral analysis of Internet topologies. In *Proceedings of IEEE Infocom 2003*, San Francisco, CA, Apr. 2003.

[7] H. Haddadi, G. Iannaccone, A. Moore, R. Mortier, and M. Rio. Network topologies: Inference, modelling and generation. In *IEEE Communications Surveys and Tutorials*, TO APPEAR in 2008.

[8] S. Hanna. Representation and generation of plans using graph spectra. In *Proceedings of the 6th International Space Syntax Symposium*, Istanbul, 2007.

[9] 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.

[10] B. Luo and E. Hancock. Structural graph matching using the em algorithm and singular value decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1120–1136, Oct. 2001.

[11] 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).

[12] 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.

[13] 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).

[14] A. Seary and W. Richards. Spectral methods for analyzing and visualizing networks: an introduction. In *Dynamic Social Network Modeling and Analysis*, pages 209–228. National Academic Press, 2003.

[15] Steve Butler . Lecture notes for spectral graph theory. *Lectures in Nankai University, Tianjin, China*, 2006.

[16] A. G. Thomason. Pseudo-random graphs. *Random Graphs '85, North-Holland Mathematical Study*, 144:307–331, 1987.

[17] D. Vukadinovic, P. Huang, and T. Erlebach. On the spectrum and structure of Internet topology graphs. In *Proceedings of the Second International Workshop on Innovative Internet Computing Systems (IICS)*, 2002.

[18] X. Wang and D. Loguinov. Wealth-based evolution model for the internet as-level topology. *INFOCOM 2006. Proceedings*, pages 1–11, April 2006.

[19] 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.