Network Density of States

Kun Dong
Cornell University
Ithaca, New York
kd383@cornell.edu

Austin R. Benson
Cornell University
Ithaca, New York
arb@cs.cornell.edu

David Bindel
Cornell University
Ithaca, New York
bindel@cornell.edu

ABSTRACT

Spectral analysis connects graph structure to the eigenvalues and eigenvectors of associated matrices. Much of spectral graph theory descends directly from spectral geometry, the study of differentiable manifolds through the spectra of associated differential operators. But the translation from spectral geometry to spectral graph theory has largely focused on results involving only a few extreme eigenvalues and their associated eigenvalues. Unlike in geometry, the study of graphs through the overall distribution of eigenvalues — the *spectral density* — is largely limited to simple random graph models. The interior of the spectrum of real-world graphs remains largely unexplored, difficult to compute and to interpret.

In this paper, we delve into the heart of spectral densities of real-world graphs. We borrow tools developed in condensed matter physics, and add novel adaptations to handle the spectral signatures of common graph motifs. The resulting methods are highly efficient, as we illustrate by computing spectral densities for graphs with over a billion edges on a single compute node. Beyond providing visually compelling fingerprints of graphs, we show how the estimation of spectral densities facilitates the computation of many common centrality measures, and use spectral densities to estimate meaningful information about graph structure that cannot be inferred from the extremal eigenpairs alone.

ACM Reference Format:
Kun Dong, Austin R. Benson, and David Bindel. 2019. Network Density of States. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3292500.3330891

1 INTRODUCTION

Spectral theory is a powerful analysis tool in graph theory [9, 10, 13], geometry [6], and physics [27]. One follows the same steps in each setting:

- Identify an object of interest, such as a graph or manifold;
- Associate the object with a matrix or operator, often the generator of a linear dynamical system or the Hessian of a quadratic form over functions on the object; and
- Connect spectral properties of the matrix or operator to structural properties of the original object.

In each case, the *complete* spectral decomposition is enough to recover the original object; the interesting results relate structure to *partial* spectral information.

Many spectral methods use extreme eigenvalues and associated eigenvectors. These are easy to compute by standard methods, and are easy to interpret in terms of the asymptotic behavior of dynamical systems or the solutions to quadratic optimization problems with quadratic constraints. Several network centrality measures, such as PageRank [40], are expressed via the stationary vectors of transition matrices, and the rate of convergence to stationarity is bounded via the second-largest eigenvalue. In geometry and graph theory, Cheeger's inequality relates the second-smallest eigenvalue of a Laplacian or Laplace-Beltrami operator to the size of the smallest bisecting cut [7, 37]; in the graph setting, the associated eigenvector (the Fiedler vector) is the basis for spectral algorithms for graph partitioning [42]. Spectral algorithms for graph coordinates and clustering use the first few eigenvectors of a transition matrix or (normalized) adjacency or Laplacian [5, 39]. For a survey of such approaches in network science, we refer to [9].

Mark Kac popularized an alternate approach to spectral analysis in an expository article [28] in which he asked whether one can determine the shape of a physical object (Kac used a drum as an example) given the spectrum of the Laplace operator; that is, can one "hear" the shape of a drum? One can ask a similar question in graph theory: can one uniquely determine the structure of a network from the spectrum of the Laplacian or another related matrix? Though the answer is negative in both cases [13, 22], the spectrum is enormously informative even without eigenvector information. Unlike the extreme eigenvalues and vectors, eigenvalues deep in the spectrum are difficult to compute and to interpret, but the overall distribution of eigenvalues — known as the spectral density or density of states — provides valuable structural information. For example, knowing the spectrum of a graph adjacency matrix is equivalent to knowing $\mathrm{trace}(A^k)$, the number of closed walks of any given length k. In some cases, one wants *local* spectral densities in which the eigenvalues also have positive weights associated with a location. Following Kac, this would give us not only the frequencies of a drum, but also amplitudes based on where the drum is struck. In a graph setting, the local spectral density of an adjacency matrix at node j is equivalent to knowing $(A^k)_{jj}$, the number of closed walks of any given length k that begin and end at the node.

Unfortunately, the analysis of spectral densities of networks has been limited by a lack of scalable algorithms. While the normalized Laplacian spectra of Erdős-Rényi random graphs have an approximately semicircular distribution [50], and the spectral distributions for other popular scale-free and small-world random graph models are also known [19], there has been relatively little work on computing spectral densities of large "real-world" networks. Obtaining the full eigendecomposition is $O(N^3)$ for a graph with N nodes,

KDD '19, August 4–8, 2019, Anchorage, AK, USA

ACM ISBN 978-1-4503-6201-6/19/08…$15.00
https://doi.org/10.1145/3292500.3330891

which is prohibitive for graphs of more than a few thousand nodes. In prior work, researchers have employed methods, such as thick-restart Lanczos, that still do not scale to very large graphs [19], or heuristic approximations with no convergence analysis [2]. It is only recently that clever computational methods were developed simply to *test* for hypothesized power laws in the spectra of large real-world matrices by computing only *part* of the spectrum [18].

In this paper, we show how methods used to study densities of states in condensed matter physics [48] can be used to study spectral densities in networks. We study these methods for both the *global* density of states and for *local* densities of states weighted by specific eigenvector components. We adapt these methods to take advantage of graph-specific structure not present in most physical systems, and analyze the stability of the spectral density to perturbations as well as the convergence of our computational methods. Our methods are remarkably efficient, as we illustrate by computing densities for graphs with billions of edges and tens of millions of nodes on a single cloud compute node. We use our methods for computing these densities to create compelling visual fingerprints that summarize a graph. We also show how the density of states reveals graph properties that are not evident from the extremal eigenvalues and eigenvectors alone, and use it as a tool for fast computation of standard measures of graph connectivity and node centrality. This opens the door for the use of complete spectral information as a tool in large-scale network analysis.

2 BACKGROUND

2.1 Graph Operators and Eigenvalues

We consider weighted, undirected graphs $G = (V, E)$ with vertices $V = \{v_1, \cdots, v_N\}$ and edges $E \subseteq V \times V$. The weighted adjacency matrix $A \in \mathbb{R}^{N\times N}$ has entries $a_{ij} > 0$ to give the weight of an edge $(i, j) \in E$ and $a_{ij} = 0$ otherwise. The degree matrix $D \in \mathbb{R}^{N\times N}$ is the diagonal matrix of weighted node degrees, i.e. $D_{ii} = \sum_j a_{ij}$. Several of the matrices in spectral graph theory are defined in terms of D and A. We describe a few of these below, along with their connections to other research areas. For each operator, we let $\lambda_1 \leq \ldots \leq \lambda_N$ denotes the eigenvalues in ascending order.

Adjacency Matrix: A. Many studies on the spectrum of A originate from random matrix theory where A represents a random graph model. In these cases, the limiting behavior of eigenvalues as $N \to \infty$ is of particular interest. Besides the growth of extremal eigenvalues [10], Wigner's semicircular law is the most renowned result about the spectral distribution of the adjacency matrix [50]. When the edges are i.i.d. random variables with bounded moments, the density of eigenvalues within a range converges to a semicircular distribution. One famous graph model of this type is the Erdős-Rényi graph, where $a_{ij} = a_{ji} = 1$ with probability $p < 1$, and 0 with probability $1 - p$. Farkas et al. [19] has extended the semicircular law by investigating the spectrum of scale-free and small-world random graph models. They show the spectra of these random graph models relate to geometric characteristics such as the number of cycles and the degree distribution.

Laplacian Matrix: $L = D - A$. The Laplace operator arises naturally from the study of dynamics in both spectral geometry and spectral graph theory. The continuous Laplace operator and its generalizations are central to the description of physical systems including heat diffusion [34], wave propagation [32], and quantum mechanics [17]. It has infinitely many non-negative eigenvalues, and Weyl's law [49] relates their asymptotic distribution to the volume and dimension of the manifold. On the other hand, the discrete Laplace matrix appears in the formulation of graph partitioning problems. If $f \in \{\pm 1\}^N$ is an indicator vector for a partition $V = V_+ \cup V_-$, then $f^T Lf/4$ is the number of edges between V_+ and V_-, also known as the cut size. L is a positive-semidefinite matrix with the vector of all ones as a null vector. The eigenvalue λ_2, called the *algebraic connectivity*, bounds from below the smallest bisecting cut size; $\lambda_2 = 0$ if and only if the graph is disconnected. In addition, eigenvalues of L also appear in bounds for vertex connectivity (λ_2) [12], minimal bisection (λ_2) [15], and maximum cut (λ_N) [46].

Normalized Laplacian Matrix: $\bar{L} = I - D^{-1/2}AD^{-1/2}$. We will also mention the normalized adjacency matrix $\bar{A} = D^{-1/2}AD^{-1/2}$ and graph random walk matrix $P = D^{-1}A$ here, because these matrices have the same eigenvalues as $\bar{L}$ up to a shift. The connection to some of the most influential results in spectral geometry is established in terms of eigenvalues and eigenvectors of normalized Laplacian. A prominent example is the extension of Cheeger's inequality to the discrete case, which relates the set of smallest conductance $h(G)$ (the Cheeger constant) to the second smallest eigenvalue of the normalized Laplacian, $\lambda_2(\bar{L})$ [38]:

$$\lambda_2(\bar{L})/2 \leq h(G) = \min_{S\subset V} \frac{|\{(i,j) \in E, i \in S, j \notin S\}|}{\min(\text{vol}(S), \text{vol}(V\backslash S))} \leq \sqrt{2\lambda_2(\bar{L})},$$

where $\text{vol}(T) = \sum_{i\in T}\sum_{j=1}^{N} a_{ij}$. Cheeger's inequality offers crucial insights and powerful techniques for understanding popular spectral graph algorithms for partitioning [35] and clustering [39]. It also plays a key role in analyzing the mixing time of Markov chains and random walks on a graph [36, 44]. For all these problems, extremal eigenvalues again emerge from relevant optimization formulations.

2.2 Spectral Density (Density of States — DOS)

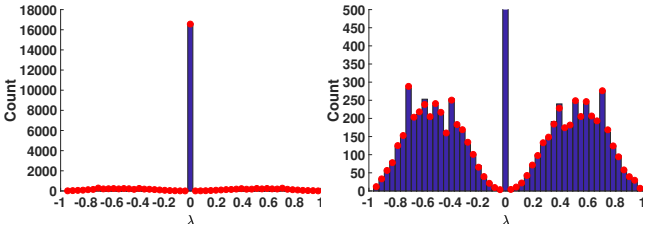
Let $H = \mathbb{R}^{N\times N}$ be any symmetric graph matrix with an eigendecomposition $H = Q\Lambda Q^T$, where $\Lambda = \text{diag}(\lambda_1, \cdots, \lambda_N)$ and $Q = [q_1, \cdots, q_N]$ is orthogonal. The spectral density induced by H is the generalized function

$$\mu(\lambda) = \frac{1}{N}\sum_{i=1}^{N} \delta(\lambda - \lambda_i), \quad \int f(\lambda)\mu(\lambda) = \text{trace}(f(H)) \qquad (1)$$

where δ is the Dirac delta function and f is any analytic test function. The spectral density μ is also referred to as the *density of states* (DOS) in the condensed matter physics literature [48], as it describes the number of states at different energy levels. For any vector $u \in \mathbb{R}^N$, the local density of states (LDOS) is

$$\mu(\lambda; u) = \sum_{i=1}^{N} |u^T q_i|^2 \delta(\lambda - \lambda_i), \quad \int f(\lambda)\mu(\lambda; u) = u^T f(H)u. \quad (2)$$

Most of the time, we are interested in the case $u = e_k$ where e_k is the kth standard basis vector—this provides the spectral information about a particular node. We will write $\mu_k(\lambda) = \mu(\lambda; e_k)$ for the pointwise density of states (PDOS) for node v_k. It is noteworthy $|e_k^T q_i| = |q_i(k)|$ gives the magnitude of the weight for v_k in the i-th eigenvector, thereby the set of $\{\mu_k\}$ encodes the entire spectral information of the graph up to sign differences. These concepts can



(a) Spectral Histogram (b) Zoom-in View

Figure 1: Spectral histogram for the normalized adjacency matrix for the CAIDA autonomous systems graph [24], an Internet topology with 22965 nodes and 47193 edges. Blue bars are the real spectrum, and red points are the approximated heights. (1a) contains high multiplicity around eigenvalue 0, so (1b) zooms in to height between [0, 500].

be easily extended to directed graphs with asymmetric matrices, for which the eigenvalues are replaced by singular values, and eigenvectors by left/right singular vectors.

Naively, to obtain the DOS and LDOS requires computing all eigenvalues and eigenvectors for an N-by-N matrix, which is infeasible for large graphs. Therefore, we turn to algorithms that approximate these densities. Since the DOS is a generalized function, it is important we specify how the estimation is evaluated. One choice is to treat μ (or μ_k) as a distribution, and measure its approximation error with respect to a chosen function space $\mathcal{L}$. For example, when $\mathcal{L}$ is the set of Lipschitz continuous functions taking the value 0 at 0, the error for estimated $\widetilde{\mu}$ is in the Wasserstein distance (a.k.a. earth-mover distance) [29]

$$W_1(\mu, \widetilde{\mu}) = \sup \left\{ \int (\mu(\lambda) - \widetilde{\mu}(\lambda)) f(\lambda) d\lambda : \mathrm{Lip}(f) \le 1 \right\}. \quad (3)$$

This notion is particularly useful when μ is integrated against in applications such as computing centrality measures.

On the other hand, we can regularize μ with a mollifier K_σ (i.e., a smooth approximation of the identity function):

$$(K_\sigma * \mu)(\lambda) = \int_{\mathbb{R}} \sigma^{-1} K\left(\frac{\lambda - \nu}{\sigma}\right) \mu(\nu) d\nu \quad (4)$$

A simplified approach is numerically integrating μ over small intervals of equal size to generate a spectral histogram. The advantage is the error is now easily measured and visualized in the L_∞ norm. For example, Figure 1 shows the exact and approximated spectral histogram for the normalized adjacency matrix of an Internet topology.

3 METHODS

The density of states plays a significant role in understanding electronic band structure in solid state physics, and so several methods have been proposed in that literature to estimate spectral densities. We review two such methods: the kernel polynomial method (KPM) which involves a polynomial expansion of the DOS/LDOS, and the Gauss Quadrature via Lanczos iteration (GQL). These methods have not previously been applied in the network setting, though Cohen-Steiner et al. [11] have independently invented an approach similar to KPM for the global DOS alone, albeit using a less numerically stable polynomial basis (the power basis associated with random walks). We then introduce a new direct nested dissection method for LDOS, as well as new graph-specific modifications to improve the convergence of the KPM and GQL approaches.

Throughout this section, H denotes any symmetric matrix.

3.1 Kernel Polynomial Method (KPM)

The Kernel Polynomial Method (KPM) [48] approximates the spectral density through an expansion in the dual basis of an orthogonal polynomial basis. Traditionally, the Chebyshev basis $\{T_m\}$ is used because of its connection to the best polynomial interpolation. Chebyshev approximation requires the spectrum to be supported on the interval $[-1, 1]$ for numerical stability. However, this condition can be satisfied by any graph matrix after shifting and rescaling:

$$\widetilde{H} = \frac{2H - (\lambda_{\max}(H) + \lambda_{\min}(H))}{\lambda_{\max}(H) - \lambda_{\min}(H)} \quad (5)$$

We can compute these extremal eigenvalues efficiently for our sparse matrix H, so the pre-computation is not an issue [41].

The Chebyshev polynomials $T_m(x)$ satisfy the recurrence

$$T_0(x) = 1,\ T_1(x) = x,\ T_{m+1}(x) = 2xT_m(x) - T_{m-1}(x). \quad (6)$$

They are orthogonal with respect to $w(x) = 2/[(1 + \delta_{0n})\pi\sqrt{1 - x^2}]$:

$$\int_{-1}^{1} w(x) T_m(x) T_n(x) dx = \delta_{mn}. \quad (7)$$

(Here and elsewhere, δ_{ij} is the Kronecker delta: 1 if $i = j$ and 0 otherwise.) Therefore, $T^*_m(x) = w(x)T_m(x)$ also forms the dual Chebyshev basis. Using (7), we can expand our DOS $\mu(\lambda)$ as

$$\mu(\lambda) = \sum_{m=0}^{\infty} d_m T^*_m(\lambda) \quad (8)$$

$$d_m = \int_{-1}^{1} T_m(\lambda)\mu(\lambda) d\lambda = \frac{1}{N}\sum_{i=1}^{N} T_m(\lambda_i) = \frac{1}{N}\mathrm{trace}(T_m(H)), \quad (9)$$

Here, $T_m(H)$ is the mth Chebyshev polynomial of the matrix H. The last equality comes from the spectral mapping theorem, which says that taking a polynomial of H maps the eigenvalues by the same polynomial. Similarly, we express the PDOS $\mu_k(\lambda)$ as

$$d_{mk} = \int_{-1}^{1} T_m(\lambda)\mu_k(\lambda) d\lambda = \sum_{i=1}^{N} |q_i(k)|^2 T_m(\lambda_i) = T_m(H)_{kk}. \quad (10)$$

We want to efficiently extract the diagonal elements of the matrices $\{T_m(H)\}$ without forming them explicitly; the key idea is to apply the stochastic trace/diagonal estimation, proposed by Hutchinson [25] and Bekas et al. [4]. Given a random probe vector z such that z_i's are i.i.d. with mean 0 and variance 1,

$$\mathbb{E}[z^T Hz] = \sum_{i,j} H_{ij}\mathbb{E}[z_i z_j] = \mathrm{trace}(H) \quad (11)$$

$$\mathbb{E}[z \odot Hz] = \mathrm{diag}(H) \quad (12)$$

where $\odot$ represents the Hadamard (elementwise) product. Choosing N_z independent probe vectors Z_j, we obtain the unbiased estimator

$$\mathrm{trace}(H) = \mathbb{E}[z^T Hz] \approx \frac{1}{N_z}\sum_{j=1}^{N_z} Z_j^T HZ_j$$

and similarly for the diagonal. Avron and Toledo [1] review many possible choices of probes for eqs. (11) and (12); a common choice is

vectors with independent standard normal entries. Using the Chebyshev recurrence (eq. (6)), we can compute the sequence $T_j(H)z$ for each probe at a cost of one matrix-vector product per term, for a total cost of $O(|E|N_z)$ time per moment $T_m(H)$.

In practice, we only use a finite number of moments rather than an infinite expansion. The number of moments required depends on the convergence rate of the Chebyshev approximation for the class of functions DOS/LDOS is integrated with. For example, the approximation error decays exponentially for test functions that are smooth over the spectrum [45], so only a few moments are needed. On the other hand, such truncation leads to Gibbs oscillations that cause error in the interpolation [45]. However, to a large extent, we can use smoothing techniques such as Jackson damping to resolve this issue [26] (we will formalize this in theorem 4.1).

3.2 Gauss Quadrature and Lanczos (GQL)

Golub and Meurant developed the well-known Gauss Quadrature and Lanczos (GQL) algorithm to approximate bilinear forms for smooth functions of a matrix [21]. Using the same stochastic estimation from §3.1, we can also apply GQL to compute DOS.

For a starting vector z and graph matrix H, Lanczos iterations after M steps produce a decomposition

$$HZ_M = Z_M^T \Gamma_M + r_M e_M^T$$

where $Z_M^T Z_M = I_M$, $Z_M^T r_M = 0$, and Γ_M tridiagonal. GQL approximates $z^T f(H) z$ with $\|z\|^2 e_1^T f(T_M) e_1$, implying

$$z^T f(H) z = \sum_{i=1}^{N} |z^T q_i|^2 f(\lambda_i) \approx \|z\|^2 \sum_{i=1}^{M} |p_{i1}|^2 f(\tau_i)$$

where $(\tau_1, p_1) \cdots, (\tau_M, p_M)$ are the eigenpairs of Γ_M. Consequently,

$$\|z\|^2 \sum_{i=1}^{M} |p_{i1}|^2 \delta(\lambda - \tau_i)$$

approximates the LDOS $\mu(\lambda; z)$.

Building upon the stochastic estimation idea and the invariance of probe vectors under orthogonal transformation, we have

$$\mathbb{E}[\mu(\lambda; z)] = \sum_{i=1}^{N} \delta(\lambda - \lambda_i) = N\mu(\lambda)$$

Hence

$$\mu(\lambda) \approx \sum_{i=1}^{M} |p_{i1}|^2 \delta(\lambda - \tau_i).$$

The approximate generalized function is exact when applied to polynomials of degree $\leq 2M + 1$. Furthermore, if we let $z = e_k$ then GQL also provides an estimation for the PDOS $\mu_k(\lambda)$. Estimation from GQL can also be converted to Chebyshev moments if needed.

3.3 Nested Dissection (ND)

The estimation error via Monte Carlo method intrinsically decays at the rate $O(1/\sqrt{N_z})$, where N_z is the number of random probing vectors. Hence, we have to tolerate the higher variance when increasing the number of probe vectors becomes too expensive. This is particularly problematic when we try to compute the PDOS for all nodes using the stochastic diagonal estimator. Therefore, we introduce an alternative divide-and-conquer method, which computes more accurate PDOS for any set of nodes at a cost comparable to the stochastic approach in practice.

Suppose the graph can be partitioned into two subgraphs by removal of a small vertex separator. Permuting the vertices so that the two partitions appear first, followed by the separator vertices. Up to vertex permutations, we can rewrite H in block form as

$$H = \begin{bmatrix} H_{11} & 0 & H_{13} \\ 0 & H_{22} & H_{23} \\ H_{13}^T & H_{23}^T & H_{33} \end{bmatrix},$$

where the indices indicate the groups identities. Leveraging this structure, we can update the recurrence relation for Chebyshev polynomials to become

$$T_{m+1}(H)_{11} = 2H_{11}T_m(H)_{11} - T_{m-1}(H)_{11} + 2H_{13}T_m(H)_{31} \quad (13)$$

Recursing on the partitioning will lead to a nested dissection, after which we will use direct computation on sufficiently small sub-blocks. We denote the indexing of each partition with $I_p^{(t)} = I_s^{(t)} \bigcup I_\ell^{(t)} \bigcup I_r^{(t)}$, which represents all nodes in the current partition, the separators, and two sub-partitions, respectively. For the separators, equation 13 leads to

$$T_{m+1}(H)(I_p^{(t)}, I_s^{(t)}) = 2H(I_p^{(t)}, I_p^{(t)})T_m(H)(I_p^{(t)}, I_s^{(t)}) - T_{m-1}(H)(I_p^{(t)}, I_s^{(t)}) + 2 \sum_{t' \in S_t} H(I_p^{(t)}, I_s^{(t')})T_m(H)(I_s^{(t')}, I_s^{(t)}) \quad (14)$$

where S_t is the path from partition t to the root; and for the leaf blocks, $I_s^{(t)} = I_p^{(t)}$ in equation 14. The result is Algorithm 1.

Algorithm 1: Nested Dissection for PDOS Approximation

Input: Symmetric graph matrix H with eigenvalues in $[-1, 1]$
Output: $C \in \mathbb{R}^{N \times M}$ where c_{ij} is the j-th Chebyshev moment for i-th node.
begin
  Obtain partitions $\{I_p^{(t)}\}$ in a tree structure through multilevel nested dissection.
  for $m = 1$ ***to*** M **do**
    Traverse partition tree in pre-order:
      Compute the separator columns with eq. (14).
      if $I_p^{(t)}$ *is a leaf block* **then**
        Compute the diagonal entries with equation (14).
      end
  end
end

The multilevel nested dissection process itself has a well-established algorithm by Karypis and Kumar, and efficient implementation is available in *METIS* [30]. Note that this approach is only viable when the graph can be partitioned with a separator of small size. Empirically, we observe this assumption to hold for many real-world networks. The biggest advantage of this approach is we can very efficiently obtain PDOS estimation for a subset of nodes with much better accuracy than KPM.

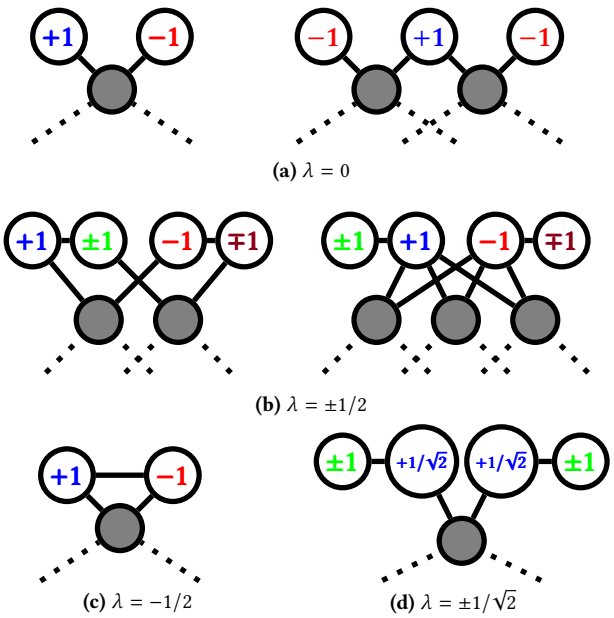


Figure 2: Common motifs (induced subgraphs) in graph data that result in localized spikes in the spectral density. Each motif generates a specific eigenvalue with locally-supported eigenvectors. Here we uses the normalized adjacency matrix to represent the graph, although we can perform the same analysis for the adjacency, Laplacian, or normalized Laplacian (only the eigenvalues would be different). The eigenvectors are supported only on the labeled nodes.

3.4 Motif Filtering

In many graphs, there are large spikes around particular eigenvalues; for example, see fig. 1. This phenomenon affects the accuracy of DOS estimation in two ways. First, the singularity-like behavior means we need many more moments to obtain a good approximation in polynomial basis. Secondly, due to the equi-oscillation property of Chebyshev approximation, error in irregularities (say, at a point of high concentration in the spectral density), spreads to other parts of the spectrum. This is a problem in our case, as the spectral density of real-world networks are far from uniform.

High multiplicity eigenvalues are typically related to local symmetries in a graph. The most prevalent example is two dangling nodes attached to the same neighbor as shown in 2a, which accounts for most eigenvalues around 0 for (normalized) adjacency matrix with a localized eigenvector taking value +1 on one node and −1 on the other. In addition, we list a few more motifs in figure 2 that appear most frequently in real-world graphs. All of them can be associated with specific eigenvalues, and we include the corresponding ones in normalized adjacency matrix for our example.

To detect these motifs in large graphs, we deploy a randomized hashing technique. Given a random vector z, the hashing weight $w = Hz$ encodes all the neighborhood information of each node. To find node copies (left in Figure 2a), we seek pairs (i, j) such that $w_i = w_j$; with high probability, this only happens when v_i and v_j share the same neighbors. Similarly, all motifs in Figure 2 can be characterized by union and intersection of neighborhood lists.

After identifying motifs, we need only approximate the (relatively smooth) density of the remaining spectrum. The eigenvectors associated with these remaining non-motif eigenvalues must be constant across cycles in the canonical decomposition of the associated permutations. Let $P \in \mathbb{R}^{N\times r}$ denote an orthonormal basis for the space of such vectors formed from columns of the identity and (normalized) indicators for nodes cyclically permuted by the motif. The matrix $H_r = P^T HP$ then has identical eigenvalues to H, except with all the motif eigenvalues omitted. We may form H_r explicitly, as it has the same sparsity structure as H but with a supernode replacing the nodes in each instance of a motif cycle; or we can achieve the same result by replacing each random probe Z with the projected probe $Z_r = PP^T Z$ at an additional cost of $O(N_{\text{motif}})$ per probe, where N_{motif} is the number of nodes involved in motifs.

The motif filtering method essentially allow us to isolate the spiky components from the spectrum. As a result, we are able to obtain a more accurate approximation using fewer Chebyshev moments. Figure 3 demonstrates the improvement on the approximation as we procedurally filter out motifs at 0, $-1/3$, $-1/2$, and $-1/4$. The eigenvalue $-1/m$ can be generated by an edge attached to the graph through $m-1$ nodes, similar to motif (2c).

4 ERROR ANALYSIS

4.1 KPM Approximation Error

This section provides an error bound for our regularized DOS approximation $K_\sigma * \mu$. We will start with the following theorem.

Theorem 4.1 (Jackson's Theorem [26]). *If $f : [-1, 1] \to \mathbb{R}$ is Lipschitz continuous with constant L, its best degree M polynomial approximation $\widehat{f}^M$ has an L_∞ error of at most $6L/M$. The approximation can be constructed as*

$$\widehat{f}^M = \sum_{m=0}^{M} J_m c_m T_m(x)$$

where J_m are Jackson smoothing factors and c_m are the Chebyshev coefficients.

We can pick a smooth mollifier K with $\text{Lip}(K) = 1$. For any $\nu \in \mathbb{R}$ and $\lambda \in [-1, 1]$ there exists a degree M polynomial such that

$$|K_\sigma(\nu - \lambda) - \widehat{K}^M_\sigma(\nu - \lambda)| < \frac{6L}{M\sigma}$$

Define $\widehat{\mu}^M = \sum_{m=0}^{M} J_m d_m \phi_m$ to be the truncated DOS series,

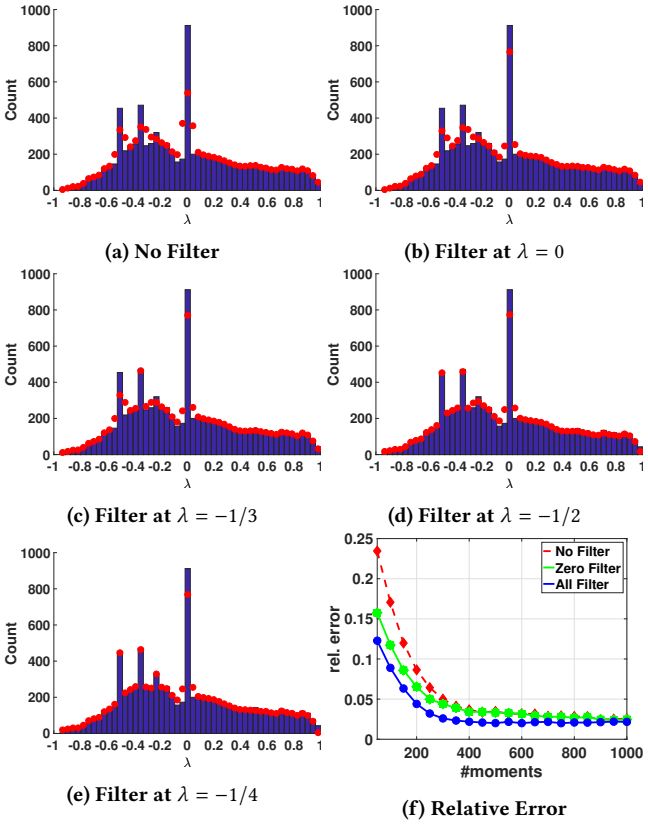
$$\int_{-1}^{1} \widehat{f}^M(\lambda)\mu(\lambda)d\lambda = \int_{-1}^{1} f(\lambda)\widehat{\mu}^M(\lambda)d\lambda = \sum_{m=0}^{M} J_m c_m d_m.$$

Therefore,

$$\begin{aligned}\|K_\sigma * (\mu - \widehat{\mu}^M)\|_\infty &= \max_\nu \left| \int_{-1}^{1} K_\sigma(\nu - \lambda)(\mu(\lambda) - \widehat{\mu}^M(\lambda))d\lambda \right| \\ &\le \max_\nu \int_{-1}^{1} |K_\sigma(\nu - \lambda) - \widehat{K}^M_\sigma(\nu - \lambda)|\mu(\lambda)d\lambda \\ &\le \frac{6L}{M\sigma}.\end{aligned}$$

Consider $\widetilde{\mu}^M$ to be the degree M approximation from KPM,

$$\|K_\sigma * (\mu - \widetilde{\mu}^M)\|_\infty \le \|K_\sigma * (\mu - \widehat{\mu}^M)\|_\infty + \|K_\sigma\|_\infty \|\widehat{\mu}^M - \widetilde{\mu}^M\|_1.$$



(a) No Filter (b) Filter at $\lambda = 0$

(c) Filter at $\lambda = -1/3$ (d) Filter at $\lambda = -1/2$

(e) Filter at $\lambda = -1/4$ (f) Relative Error

Figure 3: The improvement in accuracy of the spectral histogram approximation on the normalized adjacency matrix for the High Energy Physics Theory (HepTh) Collaboration Network, as we sweep through spectrum and filter out motifs. The graph has 8638 nodes and 24816 edges. Blue bars are the real spectrum, and red points are the approximated heights. (3a-3e) use 100 moments and 20 probe vectors. (3f) shows the relative L_1 error of the spectral histogram when using no filter, filter at $\lambda = 0$, and all filters.

If we use a probe z with independent standard normal entries for the trace estimation,

$$\widetilde{\mu}(\lambda) = \sum_{i=1}^{N} w_i^2 \delta(\lambda - \lambda_i)$$

where $w = Q^T z$ is the weight for z in the eigenbasis. Hence

$$\|\widehat{\mu}^M - \widetilde{\mu}^M\|_1 \leq \sum_{i=1}^{N} |1 - w_i^2|.$$

Finally,

$$\mathbb{E}\left[\|K_\sigma * (\mu - \widetilde{\mu}^M)\|\right] \leq \frac{1}{\sigma}\left(\frac{6L}{M} + \|K\|_\infty \mathbb{E}[|1 - w_1^2|]\right)$$

If we take N_z independent probe vectors, then $N_z w_1^2 \sim \chi^2(N_z)$, which means the expectation decays asymptotically like $\sqrt{2/(\pi N_z)}$.

4.2 Perturbation Analysis

In this section, we limit our attention to symmetric graph matrix H. Extracting graph information using DOS, whether as a distribution for functions on a graph or as a direct feature in the form of spectral moments, requires stability under small perturbations. In the case of removing/adding a few number of nodes/edges, the Cauchy Interlacing Theorem [33] gives a bound on each individual new eigenvalue by the old ones. For example, if we remove $r \ll N$ nodes to get a new graph matrix $\widetilde{H}$, then

$$\lambda_i(H) \leq \lambda_i(\widetilde{H}) \leq \lambda_{i+r}(H) \quad \text{for} \quad i \leq N - r \tag{15}$$

However, this bound may not be helpful when the impact of the change is not reflected by its size. Hence, we provide a theorem that relates the Wasserstein distance (see equation 3) change and the Frobenius norm of the perturbation. Without loss of generality, we assume the eigenvalues of H lie in $[-1, 1]$ already.

Theorem 4.2. *Suppose $\widetilde{H} = H + \delta H$ is the perturbed graph matrix with spectral density $\widetilde{\mu}$, then*

$$W_1(\mu, \widetilde{\mu}) \leq \|\delta H\|_F$$

Proof. Let $\mathcal{L}$ be the space of Lipschitz functions with $f(0) = 0$.

$$\begin{aligned} W_1(\mu, \widetilde{\mu}) &= \sup_{f \in \mathcal{L}, \mathrm{Lip}(f)=1} \int f(\lambda)(\mu(\lambda) - \widetilde{\mu}(\lambda))d\lambda \\ &= \frac{1}{N} \sup_{f \in \mathcal{L}, \mathrm{Lip}(f)=1} \mathrm{trace}(f(H) - f(\widetilde{H})) \\ &\leq \sup_{f \in \mathcal{L}, \mathrm{Lip}(f)=1, \|v\|=1} v^T (f(H) - f(\widetilde{H}))v. \end{aligned}$$

By Theorem 3.8 from Higham [23], the perturbation on $f(H)$ is bounded by the Fréchet derivative,

$$\|f(H) - f(\widetilde{H})\|_2 \leq \mathrm{Lip}(f)\|\delta H\|_F + o(\|\delta H\|_F).$$

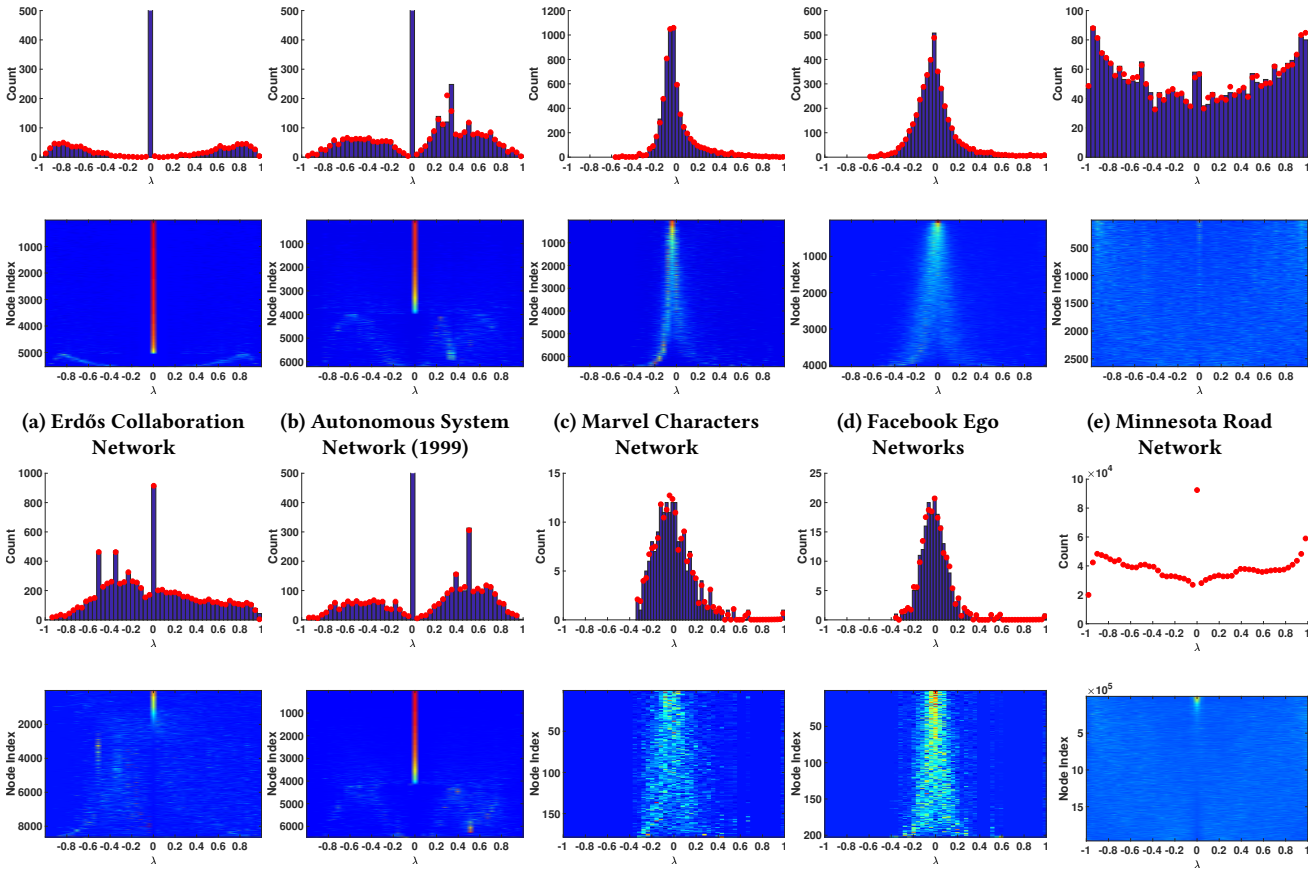
□

5 EXPERIMENTS

5.1 Gallery of DOS/PDOS

We first present our spectral histogram approximation from DOS/PDOS on a wide variety of graphs, including collaboration networks, online social networks, road networks and autonomous systems (dataset details are in the appendix). For all examples, we apply our methods to the normalized adjacency matrices using 500 Chebyshev moments and 20 Hadamard probe vectors. Afterwards, the spectral density is integrated into 50 histogram bins. In figure 4, the DOS approximation is on the first row, and the PDOS approximation is on the second. When a spike exists in the spectrum, we apply motif filtering, and DOS is zoomed appropriately to show the remaining part. For PDOS, we stack the spectral histograms for all nodes vertically, sorted by their projected weights on the leading left singular vector. Red indicates that a node has high weight at certain parts of the spectrum, whereas blue indicates low weight.

We observe many distinct shapes of spectrum in our examples. The eigenvalues of denser graphs, such as the Marvel characters



(a) Erdős Collaboration Network (b) Autonomous System Network (1999) (c) Marvel Characters Network (d) Facebook Ego Networks (e) Minnesota Road Network

(f) HepTh Collaboration Network (g) Autonomous System Network (2000) (h) Harry Potter Characters Network (i) Twitter Ego Networks (j) California Road Network

Figure 4: DOS(top)/PDOS(bottom) histograms for the normalized adjacency of 10 networks from five domains. For DOS, blue bars are the true spectrum, and red points are from KPM (500 moments and 20 Hadamard probes). For PDOS, the spectral histograms of all nodes are aligned vertically. Red indicates high weight around an eigenvalue, and blue indicates low weight. The true spectrum for the California Road Network (4j) is omitted, as it is too large to compute exactly (1,965,206 nodes).

network (4c: average degree 52.16) and Facebook union of ego networks (4d: average degree 43.69), exhibit decay similar to thepower-law around $\lambda = 0$. There has been study on the power-law distribution in the eigenvalues of the adjacency and the Laplacian matrix, but it only focuses on the leading eigenvalues rather than the entire spectrum [18] for large real-world datasets. Relatively sparse graphs (4a: average degree 3.06,; 4b: average degree 4.13) often possess spikes, especially around $\lambda = 0$, which reflect a larger set of loosely-connected boundary nodes. It is much more evident in the PDOS spectral histograms, which allow us to pick out the nodes with dominant weights at $\lambda = 0$ and those that contribute most to local structures. Finally, though the road network is quite sparse (ave. deg 2.50), its regularity results in a lack of special features, and most nodes contribute evenly to the spectrum according to PDOS.

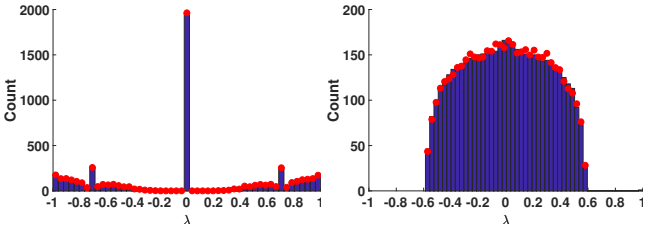
5.2 Computation time

In this experiment, we show the scaling of our methods by applying them to graphs of varying size of nodes, edges, and sparsity patterns. Rather than computation power, the memory cost of loading a graph with 100M-1B edges is more often the constraint. Hence, we report runtimes for a Python version on a Google Cloud instance with 200GB memory and an Intel Xeon E5 v3 CPU at 2.30GHz.

The datasets we use are obtained from the SNAP repository [31]. For each graph, we compute the first 10 Chebyshev moments using KPM with 20 probe vectors. Most importantly, the cost for each moment is independent of the total number of moments we compute. Table 1 reports number of nodes, number of edges, average degree of nodes, and the average runtime for computing each moment. We can observe that the runtime is in accordance with the theoretical complexity $O(N_z(|V| + |E|))$. For the Friendster social network with about 1.8 billion edges, computing each moment takes about 1000 seconds to compute, which means we could obtain a rough approximation to its spectrum within a day. As the dominant cost is matrix-matrix multiplication and we use several probe vectors, our approach has ample opportunity for parallel computation.

Table 1: Average time to compute each Chebyshev moment (with 20 probes) for graphs from the SNAP repository.

Network	# Nodes	# Edges	Avg. Deg.	Time (s)
Facebook	4,039	88,234	43.69	0.007
AstroPh	18,772	198,110	21.11	0.028
Enron	36,692	183,831	10.02	0.046
Gplus	107,614	13,673,453	254.12	1.133
Amazon	334,863	925,872	5.53	0.628
Neuron	1,018,524	24,735,503	48.57	9.138
RoadNetCA	1,965,206	2,766,607	2.82	2.276
Orkut	3,072,441	117,185,083	76.28	153.7
LiveJournal	3,997,962	34,681,189	17.35	14.52
Friendster	65,608,366	1,806,067,135	55.06	1,017



(a) $m = 1$ **(b)** $m = 5$

Figure 5: Spectral histogram for scale-free model with 5000 **nodes and different** m**. Blue bars are the real spectrum, red points are from KPM (**500 **moments and** 20 **probes).**

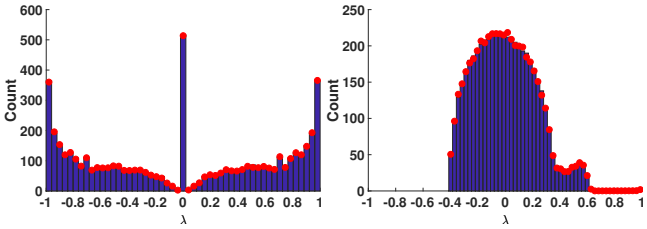
5.3 Model Verification

In this experiment, we investigate the spectrum for some of the popular graph models, and whether they resemble the behavior of real-world data. Two of the most popular models used to describe real-world graphs are the scale-free model [3] and the small-world model [47]. Farkas et al. [19] has analyzed the spectrum of the adjacency matrix; we instead consider the normalized adjacency.

The scale-free model grows a random graph with the preferential attachment process, starting from an initial seed graph and adding one node and m edges at every step. Figure 5 shows spectral histograms for this model with 5000 nodes and different choices of m. When $m = 1$, the generated graph has abundant local motifs like many sparse real-world graphs. By searching in PDOS for the nodes that have high weight at the two spikes, we find node-doubles ($\lambda = 0$) and singly-attached chains ($\lambda = \pm 1/\sqrt{2}$). When $m = 5$, the graph is denser, without any particular motifs, resulting in an approximately semicircular spectral distribution.

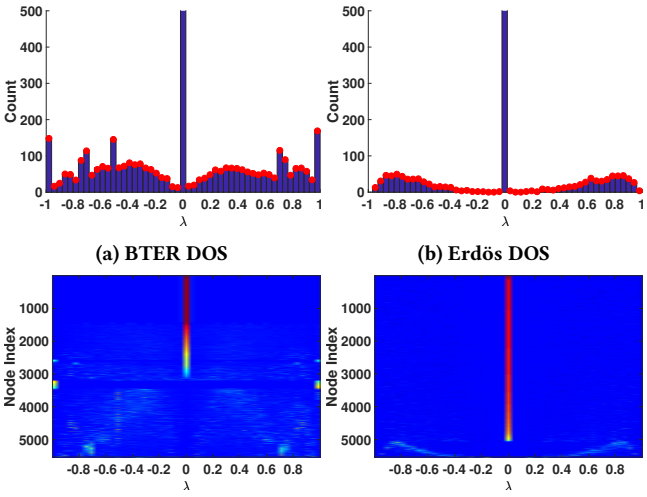
The small-world model generates a random graph by re-wiring edges of a ring lattice with a certain probability p. Here we construct these graphs on 5000 nodes with $p = 0.5$; the pattern in spectrum is insensitive for a wide range of p. In Figure 6, when the graph is sparse with 5000 edges, the spectrum has spikes at 0 and ±1, indicating local symmetries, bipartite structure, and disconnected components. With 50000 edges, localized structures disappear and the spectrum has narrower support.

Finally, we investigate the Block Two-Level Erdös-Rényi (BTER) model [43], which directly fits an input graph. BTER constructs a similar graph by a two-step process: first create a collection of Erdös-Rényi subgraphs, then interconnect those using a Chung-Lu



(a) $|E| = 5k$ **(b)** $|E| = 50k$

Figure 6: Spectral histograms for small-world model with 5000 **nodes and re-wiring probability** $p = 0.5$**, starting with** 5000 **(6a) and** 50000 **(6b edges. Blue bars are the real spectrum, red points are from KPM (**5000 **moments and** 20 **probes).**



(a) BTER DOS **(b) Erdös DOS**

(c) BTER PDOS **(d) Erdős PDOS**

Figure 7: Comparison of spectral histogram between Erdős Collaboration Network and the BTER model. Both DOS and PDOS are computed with 500 **moments and** 20 **probe vectors.**

model [8]. Seshadhri et al. showed their model accurately captures the observable properties of the given graph, including the eigenvalues of the adjacency matrix. Figure 7 compares the DOS/PDOS of the Erdös collaboration network and its BTER counterpart. Unlike the original graph, most 0 eigenvalues in BTER graph come from isolated nodes. The BTER graph also has many more isolated edges ($\lambda = \pm 1$), singly-attached chains ($\lambda = \pm 1/\sqrt{2}$)), and singly-attached triangles ($\lambda = -1/2$). We locate these motifs by inspecting nodes with high weights at respective part of the spectrum.

6 DISCUSSION

In this paper, we make the computation of spectral densities a practical tool for the analysis of large real-world network. Our approach borrows from methods in solid state physics, but with adaptations that improve performance in the network analysis setting by special handling of graph motifs that leave distinctive spectral fingerprints. We show that the spectral densities are stable to small changes in the graph, as well as providing an analysis of the approximation error in our methods. We illustrate the efficiency of our approach by treating graphs with tens of millions of nodes and billions of edges using only a single compute node. The method

provides a compelling visual fingerprint of a graph, and we show how this fingerprint can be used for tasks such as model verification.

Our approach opens the door for the use of complete spectral information in large-scale network analysis. It provides a framework for scalable computation of quantities already used in network science, such as common centrality measures and graph connectivity indices (such as the Estrada index) that can be expressed in terms of the diagonals and traces of matrix functions. But we expect it to serve more generally to define new families of features that describe graphs and the roles nodes play within those graphs. We have shown that graphs from different backgrounds demonstrate distinct spectral characteristics, and thus can be clustered based on those. Looking at LDOS across nodes for role discovery, we can identify the ones with high similarity in their local structures. Moreover, extracting nodes with large weights at various points of the spectrum uncovers motifs and symmetries. In the future, we expect to use DOS/LDOS as graph features for applications in graph clustering, graph matching, role classification, and other tasks.

Acknowledgments. We thank NSF DMS-1620038 for supporting this work.

REFERENCES

[1] Haim Avron and Sivan Toledo. 2011. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM (JACM)* 58, 2 (2011), 8.

[2] Anirban Banerjee. 2008. *The spectrum of the graph Laplacian as a tool for analyzing structure and evolution of networks.* Ph.D. Dissertation.

[3] Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. *science* 286, 5439 (1999), 509–512.

[4] Costas Bekas, Effrosyni Kokiopoulou, and Yousef Saad. 2007. An estimator for the diagonal of a matrix. *Applied Numerical Mathematics* 57, 11-12 (2007), 1214–1229.

[5] Mikhail Belkin and Partha Niyogi. 2001. Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic (NIPS'01)*. MIT Press, Cambridge, MA, USA, 585–591.

[6] Isaac Chavel. 1984. *Eigenvalues in Riemannian geometry*. Vol. 115. Academic press.

[7] Jeff Cheeger. 1969. A lower bound for the smallest eigenvalue of the Laplacian. In *Proceedings of the Princeton conference in honor of Professor S. Bochner*.

[8] Fan Chung and Linyuan Lu. 2002. Connected components in random graphs with given expected degree sequences. *Annals of combinatorics* 6, 2 (2002), 125–145.

[9] Fan Chung and Linyuan Lu. 2006. *Complex graphs and networks*. Number 107 in CBMS Regional Conference Series in Mathematics. American Mathematical Soc.

[10] Fan RK Chung and Fan Chung Graham. 1997. *Spectral graph theory*. Number 92. American Mathematical Soc.

[11] David Cohen-Steiner, Weihao Kong, Christian Sohler, and Gregory Valiant. 2018. Approximating the Spectrum of a Graph. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1263–1271.

[12] Dragoš Cvetkovic, Slobodan Simic, and Peter Rowlinson. 2009. *An introduction to the theory of graph spectra*. Cambridge University Press.

[13] D. M. Cvetković, M. Doob, and H. Sachs. 1998. *Spectra of Graphs: Theory and Applications* (third ed.). Wiley.

[14] Tim Davis, WW Hager, and IS Duff. 2014. SuiteSparse. *URL: faculty. cse. tamu. edu/davis/suitesparse. html* (2014).

[15] William E Donath and Alan J Hoffman. 2003. Lower bounds for the partitioning of graphs. In *Selected Papers Of Alan J Hoffman: With Commentary*. World Scientific, 437–442.

[16] dpmartin42. 2014. Networks. https://github.com/dpmartin42/Networks/commits/master.

[17] Francois Ducastelle and Françoise Cyrot-Lackmann. 1970. Moments developments and their application to the electronic charge distribution of d bands. *Journal of Physics and Chemistry of Solids* 31, 6 (1970), 1295–1306.

[18] Nicole Eikmeier and David F Gleich. 2017. Revisiting Power-law Distributions in Spectra of Real World Networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 817–826.

[19] Illés J Farkas, Imre Derényi, Albert-László Barabási, and Tamas Vicsek. 2001. Spectra of "real-world" graphs: Beyond the semicircle law. *Physical Review E* 64, 2 (2001), 026704.

[20] David Gleich. 2016. Repository of Difficult Graph Experiments and Results (RODGER). https://www.cs.purdue.edu/homes/dgleich/rodger/.

[21] Gene H Golub and Gérard Meurant. 1997. Matrices, moments and quadrature II; how to compute the norm of the error in iterative methods. *BIT Numerical Mathematics* 37, 3 (1997), 687–705.

[22] Carolyn Gordon, David L. Webb, and Scott Wolpert. 1992. One Cannot Hear the Shape of a Drum. *Bull. Amer. Math. Soc.* 27 (1992), 134–138.

[23] Nicholas J Higham. 2008. *Functions of matrices: theory and computation*. Vol. 104. Siam.

[24] B. Huffaker, M. Fomenkov, and k. claffy. 2012. *Internet Topology Data Comparison*. Technical Report. Cooperative Association for Internet Data Analysis (CAIDA).

[25] Michael F Hutchinson. 1990. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation* 19, 2 (1990), 433–450.

[26] Dunham Jackson. 1911. *Über die Genauigkeit der Annäherung stetiger Funktionen durch ganze rationale Funktionen gegebenen Grades und trigonometrische Summen gegebener Ordnung*. Dieterich'schen Universität Buchdruckerei.

[27] John David Jackson. 2006. *Mathematics for Quantum Mechanics: An Introductory Survey of Operators, Eigenvalues, and Linear Vector Spaces*. Dover Publications.

[28] Mark Kac. 1966. Can One Hear the Shape of a Drum? *The American Mathematical Monthly* 73, 4 (1966), 1–23.

[29] Leonid Vasilevich Kantorovich and Gennady S Rubinstein. 1958. On a space of completely additive functions. *Vestnik Leningrad. Univ* 13, 7 (1958), 52–59.

[30] George Karypis and Vipin Kumar. 1998. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. on Scientific Computing* 20, 1 (1998).

[31] Jure Leskovec and Andrej Krevl. 2014. SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data.

[32] Bruno Lévy. 2006. Laplace-Beltrami eigenfunctions towards an algorithm that "understands" geometry. In *Shape Modeling and Applications, 2006. SMI 2006. IEEE International Conference on*. IEEE, 13–13.

[33] Jan R Magnus and Heinz Neudecker. 1988. Matrix differential calculus with applications in statistics and econometrics. *Wiley series in probability and mathematical statistics* (1988).

[34] H. P. McKean. 1972. Selberg's trace formula as applied to a compact Riemann surface. *Communications on Pure and Applied Mathematics* 25, 3 (1972), 225–246.

[35] Frank McSherry. 2001. Spectral partitioning of random graphs. In *focs*. IEEE, 529.

[36] M. Mihail. 1989. Conductance and convergence of Markov chains-a combinatorial treatment of expanders. In *30th Annual Symposium on Foundations of Computer Science*. IEEE. https://doi.org/10.1109/sfcs.1989.63529

[37] Bojan Mohar. 1989. Isoperimetric numbers of graphs. *Journal of Combinatorial Theory, Series B* 47, 3 (1989), 274–291.

[38] Ravi Montenegro, Prasad Tetali, et al. 2006. Mathematical aspects of mixing times in Markov chains. *Foundations and Trends® in Theoretical Computer Science* 1, 3 (2006), 237–354.

[39] Andrew Y Ng, Michael I Jordan, and Yair Weiss. 2002. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*. 849–856.

[40] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank citation ranking: Bringing order to the web*. Technical Report. Stanford InfoLab.

[41] B. N. Parlett. 1984. The Software Scene in the Extraction of Eigenvalues from Sparse Matrices. *SIAM J. Sci. Statist. Comput.* 5, 3 (sep 1984), 590–604. https://doi.org/10.1137/0905042

[42] Alex Pothen, Horst D. Simon, and Kan-Pu Liou. 1990. Partitioning Sparse Matrices with Eigenvectors of Graphs. *SIAM J. Matrix Anal. Appl.* 11, 3 (1990), 430–452.

[43] Comandur Seshadhri, Tamara G Kolda, and Ali Pinar. 2012. Community structure and scale-free collections of Erdős-Rényi graphs. *Physical Review E* 85, 5 (2012), 056109.

[44] Alistair Sinclair and Mark Jerrum. 1989. Approximate counting, uniform generation and rapidly mixing Markov chains. *Information and Computation* 82, 1 (jul 1989), 93–133. https://doi.org/10.1016/0890-5401(89)90067-9

[45] Lloyd N Trefethen. 2013. *Approximation theory and approximation practice*. Vol. 128. Siam.

[46] Luca Trevisan. 2012. Max cut and the smallest eigenvalue. *SIAM J. Comput.* 41, 6 (2012), 1769–1786.

[47] Duncan J Watts and Steven H Strogatz. 1998. Collective dynamics of 'small-world' networks. *nature* 393, 6684 (1998), 440.

[48] Alexander Weiße, Gerhard Wellein, Andreas Alvermann, and Holger Fehske. 2006. The kernel polynomial method. *Reviews of modern physics* 78, 1 (2006).

[49] Hermann Weyl. 1911. Über die asymptotische Verteilung der Eigenwerte. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse* 1911 (1911), 110–117.

[50] Eugene P. Wigner. 1958. On the distribution of the roots of certain symmetric matrices. *Annals of Mathematics* (1958), 325–327.

A DATA SOURCE

The datasets used in this paper mainly come from the SNAP [31] and RODGER repositories [20]. Table 2 is a list of the networks from these two sources.

Table 2: List of datasets and the corresponding source.

Network	Full Name	Source
Facebook	Facebook Ego Networks	SNAP
Gplus	Google+ Ego Networks	SNAP
Twitter	Twitter Ego Networks	SNAP
LiveJournal	LiveJournal Online Social Network	SNAP
Friendster	Friendster Online Social Network	SNAP
Orkut	Orkut Online Social Network	SNAP
Amazon	Amazon Product Network	SNAP
Enron	Enron Email Communication Network	SNAP
AstroPh	Arxiv Astro Physics Collaboration Network	SNAP
HepTh	Arxiv High Energy Physics Theory Collaboration Network	SNAP
RoadNetCA	California Road Network	SNAP
AS-733	Autonomous System Network	SNAP
AS-CAIDA	CAIDA Autonomous System Network	SNAP
Neuron	Megascale Cell-Cell Similarity Network	SNAP
Erdös	Erdös Collaboration Network	RODGER
Marvel Chars	Marvel Characters Network	RODGER

In addition, we used the Minnesota Road Network from the SuiteSparse Matrix Collection [14], and the Harry Potter Characters Network from an open source repository [16].

B CODE RELEASE

Code for reproducible experiments and figures are available at https://github.com/kd383/NetworkDOS.