GRAPH DISTANCE FROM THE TOPOLOGICAL VIEW OF NON-BACKTRACKING CYCLES

LEO TORRES*, PABLO SUÁREZ-SERRATO†, AND TINA ELIASSI-RAD‡

Abstract. Whether comparing networks to each other or to random expectation, measuring dissimilarity is essential to understanding the complex phenomena under study. However, determining the structural dissimilarity between networks is an ill-defined problem, as there is no canonical way to compare two networks. Indeed, many of the existing approaches for network comparison differ in their heuristics, efficiency, interpretability, and theoretical soundness. Thus, having a notion of distance that is built on theoretically robust first principles and that is interpretable with respect to features ubiquitous in complex networks would allow for a meaningful comparison between different networks. Here we introduce a theoretically sound and efficient new measure of graph distance, based on the *length spectrum* function from algebraic topology, which compares the structure of two undirected, unweighted graphs by considering their non-backtracking cycles. We show how this distance relates to structural features such as presence of hubs and triangles through the behavior of the eigenvalues of the non-backtracking matrix, and we showcase its ability to discriminate between networks in both real and synthetic data sets. By taking a topological interpretation of non-backtracking cycles, this work presents a novel application of topological data analysis to the study of complex networks.

Key words. graph distance, algebraic topology, length spectrum

AMS subject classifications. Spectral Graph Theory, Length Spectrum, Random Graphs, Metric Spaces, Topological Data Analysis, Geometric Data Analysis

1. Introduction.

As the network science literature continues to expand and scientists compile more examples of real life networked data sets [14, 29] coming from an ever growing range of domains, there is a need to develop methods to compare complex networks, both within and across domains. Many such graph distance measures have been proposed [46, 26, 4, 8, 39, 44, 12, 13, 9], though they vary in the features they use for comparison, their interpretability in terms of structural features of complex networks, computational costs, as well as in the discriminatory power of the resulting distance measure. This reflects the fact that complex networks represent a wide variety of systems whose structure and dynamics are difficult to encapsulate in a single distance score. For the purpose of providing a principled, interpretable, efficient and effective notion of distance, we turn to the *length spectrum* function. The length spectrum function can be defined on a broad class of metric spaces that includes Riemannian manifolds and graphs. The discriminatory power of the length spectrum is well known in other contexts: it can distinguish certain one-dimensional metric spaces up to isometry [15], and it determines the Laplacian spectrum in the case of closed hyperbolic surfaces [30]. However, it is not clear if this discriminatory power is also present in the case of complex networks. Accordingly, we present a study on the following question: **is the length spectrum function useful for the comparison of complex networks?**

We answer this question in the positive by introducing the **Non-Backtracking spectral Distance (NBD)**: a principled, interpretable, efficient, and effective measure that quantifies the distance between two undirected, unweighted networks. NBD

*Network Science Institute, Northeastern University, Boston, MA (leo@leotrs.com).

†Department of Mathematics, UC Santa Barbara USA, and Instituto de Matemáticas, Universidad Nacional Autónoma de México, Ciudad de México, CDMX, (pablo@im.unam.mx).

‡Network Science Institute & College of Computer and Information Science, Northeastern University, Boston, MA, (tina@eliassi.org).

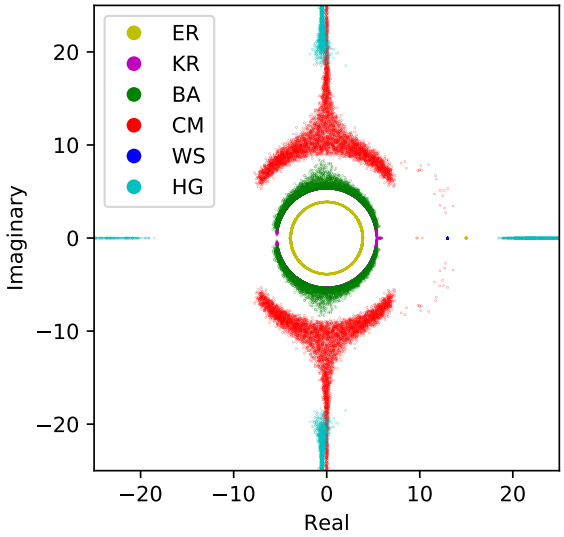

FIG. 1. *(Best viewed in color.) Complex eigenvalues of the non-backtracking matrix of random graphs (see Sec. 2.2 for definition). For different random graph models –Erdös-Rényi (ER) [18, 10], Barabási-Albert (BA) [5], Stochastic Kronecker Graphs (KR) [31, 45], Configuration Model with power law degree distribution (CM; $\gamma = 2.3$) [38], Watts-Strogatz (WS) [51], Hyperbolic Graphs (HG; $\gamma = 2.3$) [27, 2]– we plot the largest $r = 200$ eigenvalues of each of 50 random graphs of each model on the complex plane. Each model generates eigenvalue distributions presenting different geometric patterns. We analyze and exploit these patterns in order to fine-tune the Non-Backtracking spectral Distance, NBD (Sec. 4). To make the plot more readable, we show only eigenvalues close to the origin. All graphs have $n = 5 \times 10^4$ nodes and average degree approximately $\langle k \rangle = 15$.*

has several desirable properties. First, NBD is based on the theory of the length spectrum and the set of non-backtracking cycles of a graph (a non-backtracking cycle is a closed walk that does not retrace any edges immediately after traversing them); these provide the theoretical background of our method. Second, NBD is interpretable in terms of features of complex networks such as existence of hubs and triangles. This helps in the interpretation and visualization of distance scores yielded by NBD. Third, NBD is a computationally efficient method, needing only the computation of a few of the largest eigenvalues of the non-backtracking matrix of a graph. Fourth, NBD is effective at distinguishing real and synthetic networks as we demonstrate with several experiments in Section 5. In studying NBD, we highlight the topological interpretation of the non-backtracking cycles of a graph, present an efficient algorithm to compute the non-backtracking matrix, and discuss the data visualization capabilities of its complex eigenvalues (see Fig. 1). We have also made publicly available source code that implements NBD and the related matrix computations [50].

Perspective. Hashimoto [21] discussed the non-backtracking cycles of a graph (and the associated non-backtracking matrix) in relation to the theory of Zeta functions in graphs. Terras [47] explained the relationship between non-backtracking cycles and the free homotopy classes of a graph (see Sec. 2). More recently, the non-backtracking matrix has been applied to diverse applications such as node centrality [34] and community detection [28], and to the data mining tasks of clustering [42] and embedding [23]. In particular, the application to community detection is of special interest because it was shown that the non-backtracking matrix performs better at spectral clus-

tering than the Laplacian matrix in some cases [28]. Hence, there is recent interest in describing the eigenvalue distribution of the non-backtracking matrix in models such as the Erdős-Rényi random graph and the stochastic block model [11, 20, 52, 43]. Our work differs from other applied treatments of the non-backtracking matrix in that we arrive at its eigenvalues from first principles, as a relaxed version of the length spectrum. Concretely, we use the eigenvalues to compare graphs because the spectral moments of the non-backtracking matrix describe certain aspects of the length spectrum (see Sec. 3). The spectral moments of the adjacency and Laplacian matrices also describe structural features of networks [19, 41].

The rest of this paper is structured as follows. Section 2 provides necessary background information on the length spectrum, non-backtracking cycles, and the non-backtracking matrix. Section 3 explains the connection between these objects, and how we use this connection in our derivation of the Non-Backtracking spectral Distance (NBD). Section 4 describes our scalable algorithm for computing the non-backtracking matrix, as well as some of its spectral properties that help in the interpretation of NBD in terms of hubs and triangles. Section 5 discusses the practical details of computing NBD as well as several case studies and applications. We conclude in Section 6 with a discussion of limitations and future work.

2. Theoretical Background. Here we introduce two theoretical constructions: the length spectrum of a metric space and the set of non-backtracking cycles of a graph. Our analysis pivots on the fact that the latter is a particular subset of the (domain of the) former.

2.1. Length Spectrum. Consider a metric space X and a point $p \in X$. A closed curve that goes through p is called a *loop*, and p is called its *basepoint*. Two loops are *homotopic* to one another relative to a given basepoint when there exists a continuous transformation from one to the other that fixes the basepoint. The *fundamental group* of X with basepoint p is denoted by $\pi_1(X, p)$ and is defined as the first homotopy group of X, i.e., the set of all loops in X that go through p, modulo homotopy. An element of the fundamental group is called a *homotopy class*. If a homotopy class contains the loop c, we denote it as $[c]$. A *conjugacy class* of the fundamental group is a set C of homotopy classes such that for any two classes $[c_1], [c_2] \in C$ there exists a third class $[g]$ that satisfies $[c_1] = [g^{-1}][c_2][g]$. Here, the inverse of the loop g is defined as the same loop traversed in reverse. Closed curves without a distinguished basepoint are called *free loops*, and they correspond to conjugacy classes of $\pi_1(X, p)$. A well-known fact of homotopy theory is that if X is path-connected (i.e., if any two points can be connected through a continuous curve in X) then $\pi_1(X, p)$ is unique up to isomorphism, regardless of basepoint p. In the present work we only consider connected graphs, hence, we simply write $\pi_1(X)$ when there is no ambiguity. For more on homotopy, refer to [36, 22].

The length spectrum $\mathcal{L}$ is a function on $\pi_1(X)$ which assigns to each homotopy class of loops the infimum length among all of the representatives in its conjugacy class.[1] Note, importantly, that the definition of length of a homotopy class considers the length of those loops not only in the homotopy class itself, but in all other conjugate classes. In the case of compact geodesic spaces, such as finite metric graphs, this infimum is always achieved. For a finite graph where each edge has length one, the value of $\mathcal{L}$ on a homotopy class then equals the number of edges contained in the

[1] The definition presented here is also known as *marked* length spectrum. An alternative definition of the (unmarked) length spectrum does not depend on π_1; see for example [30].

length-minimizing loop. That is, for a graph $G = (V, E)$, $v \in V$, if $[c] \in \pi_1(G, v)$ and c achieves the minimum length k in all classes conjugate to $[c]$, we define $\mathcal{L}([c]) = k$.

Our interest in the length spectrum is supported by the two following facts. First, graphs are *aspherical*. More precisely, the underlying topological space of any graph is aspherical, i.e., all of its homotopy groups of dimension greater than one are trivial.[2] Therefore, it is logical to study the only non-trivial homotopy group, the fundamental group $\pi_1(G)$. Second, Constantine and Lafont [15] showed that the length spectrum of a graph determines a certain subset of it up to isomorphism. Thus, we aim to determine when two graphs are close to each other by comparing their length spectra relying on the main theorem of [15]. For completeness, we briefly mention the main result of [15], which is known as *marked length spectrum rigidity*. For a metric space X they define a subset $Conv(X)$, the minimal set to which X retracts by deformation. Let X_1, X_2 be a pair of compact, non-contractible, geodesic spaces of topological dimension one. Their main theorem shows that if the marked length spectra of X_1, X_2 are the same, then $Conv(X_1)$ is isometric to $Conv(X_2)$. Now, when X_1, X_2 are graphs, as is our case, $Conv(X_i), i = 1, 2$, corresponds to the subgraph resulting from iteratively removing nodes of degree 1 from G_i; that is, $Conv(G_i)$ is the 2-core of G_i [7]. Thus, their main theorem states that *when two graphs have the same length spectrum, their 2-cores are isomorphic.*

Given these results, it is natural to use the length spectrum as the basis of a measure of graph distance. Concretely, given two graphs, we aim to efficiently quantify how far their 2-cores are from being isomorphic by measuring the distance between their length spectra. In Section 3, we explain our approach at implementing a computationally feasible solution for this problem.

2.2. Non-Backtracking Cycles. Here we introduce the non-backtracking cycles of a graph, and the associated non-backtracking matrix, and point out the connection between these and the theory of length spectra.

Consider an undirected, unweighted graph $G = (V, E)$. For $e = (u, v) \in E$, define e^{-1} as the same edge traversed in the inverse order, $e^{-1} = (v, u)$. A *cycle* in G is a sequence of edges $e_1e_2...e_k$ such that if $e_i = (u_i, v_i)$ then $v_i = u_{i+1}$ for $i = 1, ..., k-1$ and $v_k = u_1$. Here, k is called *length* of the cycle. A *non-backtracking cycle* (NBC) is one where $e_{i+1} \neq e_i^{-1}$, $i = 1, ..., k-1$ and $e_k \neq e_1^{-1}$; that is, an edge is never followed by its own inverse. Now let $|E| = m$. The associated non-backtracking matrix B is the $2m \times 2m$ matrix where each edge is represented by two rows and two columns, one per orientation: (u, v) and (v, u). For two edges (u, v) and (k, l), B is given by

$$B_{k\to l, u\to v} = \delta_{vk}(1 - \delta_{ul}), \tag{2.1}$$

where δ_{ij} is the Kronecker delta. Thus, there is a 1 in the entry indexed by row (k, l) and column (u, v) when $u \neq l$ and $v = k$, and a 0 otherwise. Intuitively, one can interpret the B matrix as the (unnormalized) transition matrix of a random walker that does not perform backtracks: the entry at row (k, l) and column (u, v) is positive if and only if a walker can move from node u to node v (which equals node k) and then to l, without going back to u.

Importantly, NBCs are topologically relevant because backtracking edges are homotopically trivial [47]. Observe that the matrix B tracks each pair of incident edges

[2] This follows from G being homotopy equivalent to a bouquet of k circles, where k is the rank of the fundamental group of G. The universal covering of a bouquet of circles is contractible, which is equivalent to the space being aspherical. (See, for example, [22].)

that do not comprise a backtrack; indeed, $tr(B^k)$ equals the number of NBCs of length k in the graph. This fact will be fundamental in our later exposition. Observe too that B is not symmetric, and hence its eigenvalues are, in general, complex numbers.

If one is interested in B's eigenvalues rather than in the matrix itself, one may use the so-called Ihara determinant formula [21, 6], which says that the eigenvalues of B different than ± 1 are also the eigenvalues of the $2n \times 2n$ block matrix

$$B' = \begin{pmatrix} A & I - D \\ I & 0 \end{pmatrix} \tag{2.2}$$

where A is the adjacency matrix, D is the diagonal matrix with the node degrees, and I is the identity matrix of the appropriate size.

3. Non-Backtracking spectral Distance (NBD). We want to quantify the dissimilarity between two graphs by measuring the dissimilarity between their length spectra. To do this, there are two main obstacles to overcome: (i) computing the exact length spectrum of a given graph is not computationally feasible as it depends on enumerating the length of every non-backtracking cycle, and (ii) it is not clear how to compare two length spectra functions that come from two distinct graphs, because they are defined on disjoint domains (the fundamental groups of the graphs)[3]. In order to overcome these obstacles, we propose a relaxed version of the length spectrum, which we denote by $\mathcal{L}'$ and whose construction comes in the form of a two-step aggregation of the values of $\mathcal{L}$; see Figure 2 for an overview of this procedure.

3.1. Relaxed Length Spectrum. The first step of this procedure is to focus on the image of the length spectrum rather than the domain (i.e., focus on the collection of lengths of cycles). The second step is to aggregate these values by considering the sizes of the level sets of either length spectrum.

Concretely, when comparing two graphs G, H, instead of comparing $\mathcal{L}_G$ and $\mathcal{L}_H$ directly, we compare the number of cycles in G of length 3 vs. the number of cycles in H of the same length, as well as the number of cycles of length 4, of length 5, etc, thereby essentially considering the length spectra as histograms rather than functions. Theoretically, focusing on the size of the level sets provides a common ground to compare the two functions. In practice, this aggregation allows us to reduce the amount of memory needed to store the length spectra because we no longer keep track of the exact composition of each of the infinitely many (free) homotopy classes. Instead, we only keep track of the frequency of their lengths. According to this aggregation, we define the *relaxed* version of the length spectrum as the set of points $\mathcal{L}' = \{(k, n(k)) : k = 1, 2, ..\}$, where $n(k)$ is the number of conjugacy classes of π_1 (i.e., free homotopy classes) of length k.

The major downside of removing focus from the underlying group structure and shifting it towards (the histogram of values in) the image is that we lose information about the combinatorial composition of each cycle. Indeed, $\pi_1(G)$ holds information about the number of cycles of a certain length k in G; this information is also stored in $\mathcal{L}'$. However, the group structure of $\pi_1(G)$ also allows us to know how many of those cycles of length k are formed by the concatenation of two (three, four, etc.) cycles of different lengths. This information is lost when considering only the sizes of level sets of the image, i.e., when considering $\mathcal{L}'$. Fortunately, our experiments (see Sec. 5) indicate that $\mathcal{L}'$ contains enough discriminatory information to distinguish

[3] In [15], the authors need an isomorphism between the fundamental group of the spaces that are being compared, which is also computationally prohibitive.

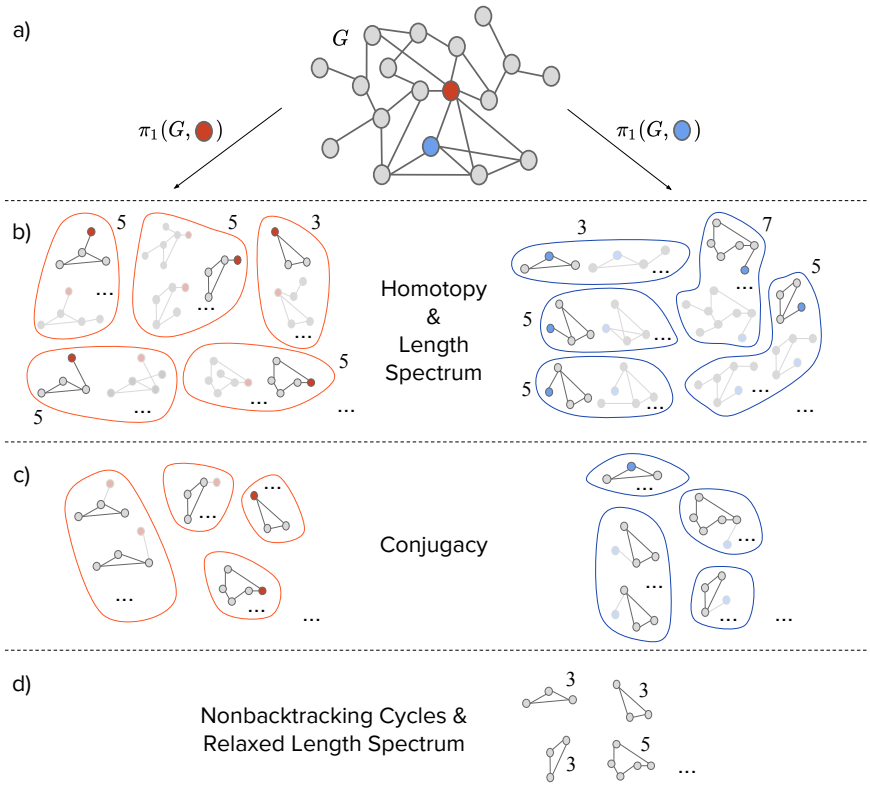

FIG. 2. *Aggregating the values of the length spectrum.* ***a)*** *A graph G with two nodes highlighted in red and blue. These two nodes are used as basepoints to construct two versions of the fundamental group.* ***b)*** *The set of all cycles based at the red node (left) and blue node (right). For either set of cycles, we encircle together those that are homotopy equivalent, thus forming a homotopy class. We annotate each class with its length, and we highlight the representative with minimal length. Note that the lengths of corresponding cycles (those that share all nodes except for the basepoints) can change when the basepoints change.* ***c)*** *We have kept only the highlighted representative in each class in b) and encircled together those that are conjugate. In each conjugacy class, we highlight the (part of) each cycle that corresponds to the free homotopy loop.* ***d)*** *By taking one representative of each conjugacy class, and ignoring basepoints, we arrive at the free homotopy classes, or equivalently, at the set of non-backtracking cycles. We annotate each cycle with its length. We arrive at the same set regardless of the basepoint. The ellipses inside the closed curves mean that there are infinitely many more elements in each set. The ellipses outside the curves mean that there are infinitely many more classes or cycles.*

between real and synthetic graphs effectively. We discuss this limitation further in our concluding remarks in Section 6.

The next step makes use of the non-backtracking cycles (NBCs). We rely on NBCs because the set of conjugacy classes of $\pi_1(G)$ is in bijection with the set of NBCs of G (see e.g., [21, 47]). In other words, to compute the set $\mathcal{L}'$ we need only account for the lengths of all NBCs. Indeed, consider the non-backtracking matrix B of G and recall that $tr(B^k)$ equals the number of NBCs of length k in the graph. This gives us precisely the set $\mathcal{L}' = \{(k, tr(B^k))\}_{k=1}^{\infty}$. Observe further that $tr(B^k)$ equals the sum of all of B's eigenvalues raised to the k-th power. Therefore, the eigenvalues of B contain all the information necessary to compute and compare $\mathcal{L}'$. In this way, we can study the (eigenvalue) spectrum of B, as a proxy for the (length) spectrum

of π_1. Note that the use of $\mathcal{L}'$ presents one possible solution to the problems of how to compute and how to compare two length spectra. We leave the investigation of alternative solutions to future lines of research.

3.2. Properties of NBD. The previous discussion yields a relaxed version of the length spectrum, $\mathcal{L}'$, that can be found efficiently: simply compute the associated matrix B and its eigenvalues. We now need only a way of comparing said eigenvalues. For this purpose, we introduce the following definition of graph distance.

DEFINITION 3.1. *Consider two graphs G, H, and write $\lambda_k = a_k + ib_k \in \mathbb{C}$ for the eigenvalues of the non-backtracking matrix of G and $\mu_k = \alpha_k + i\beta_k$ for those of H, for $k = 1, 2, .., r$, where r is some positive integer. Sort the eigenvalues in decreasing order of magnitude, $|\lambda_1| \geq |\lambda_2| \geq ... \geq |\lambda_r|$, $|\mu_1| \geq |\mu_2| \geq ... \geq |\mu_r|$. We define the* Non-Backtracking spectral Distance (NBD) *between G and H as follows,*

$$d(G, H) = d(\mathcal{L}'_G, \mathcal{L}'_H) = \sqrt{\sum_{k=1}^{r} |a_k - \alpha_k|^2 + |b_k - \beta_k|^2} \tag{3.1}$$

Remark 3.2. Note that Definition 3.1 is the Euclidean distance between two $2r$-dimensional vectors whose entries are the real and imaginary parts of the eigenvalues of the respective non-backtracking matrices. The reason to separate the real and imaginary parts is that they have different interpretations with respect to features of complex networks such as hubs and triangles (see Sec. 4.2).

There are many possible ways of comparing the non-backtracking matrix and its eigenvalues, of which the NBD is only one example. We choose NBD in particular because of the next two propositions.

PROPOSITION 3.3. *NBD is a pseudometric.*

Proof. The function d from Definition 3.1 inherits several desirable properties from the Euclidean distance: non-negativity, symmetry, and, importantly, the triangle inequality. However, the distance between two distinct graphs may be zero when they share all of their eigenvalues. Thus, d is not a metric over the space of graphs but a pseudometric. ∎

The authors of [26] propose three axioms that a measure of graph similarity should satisfy. Here, we present the equivalent axioms for a measure of graph dissimilarity (distance) and show that the NBD satisfies them. The axioms are as follows:

A1. Identity: $d(G, G) = 0$.
A2. Symmetry: $d(G, H) = d(H, G)$.
A3. Divergence: $d(K_n, \bar{K}_n) \to \infty$ as $n \to \infty$, where K_n is the complete graph and $\bar{K}_n$ is the empty graph (a graph with zero edges).

PROPOSITION 3.4. *NBD satisfies axioms A1-A3.*

Proof. Axioms A1 and A2 are satisfied because d is a pseudometric. Axiom A3 is satisfied by observing that the non-backtracking matrix of the empty graph has zero rows, and thus it has no eigenvalues, while the eigenvalues of the complete graph grow as the number of nodes grows. Thus we may accept that d satisfies axiom A3 by convention. If the reader is not satisfied by the fulfillment of an axiom by mere convention, we offer an alternative. We may compare the complete graph K_n to the graph on n nodes with a single edge linking two arbitrary nodes (an *almost* empty graph), in which case its non-backtracking matrix has two rows and two eigenvalues equal to zero. Axiom A3 is still satisfied. ∎

3.3. Using $\mathcal{L}'$ instead of $\mathcal{L}$. Although the Non-Backtracking spectral Distance satisfies all desired axioms and properties, we have deviated from the original definition of the length spectrum in important ways. In fact, as pointed out earlier, $\mathcal{L}'$ is admittedly weaker than $\mathcal{L}$ and thus one may ask if there are theoretical guarantees that the relaxed version of the length spectrum will keep some of the discriminatory power of the original. Indeed, even though the main inspiration for our work is the main result of [15], we can still trust the eigenvalue spectrum of B to be useful when comparing graphs. On the one hand, the spectrum of B has been found to yield fewer isospectral graph pairs when compared to the adjacency and Laplacian matrices in the case of small graphs [17]. On the other hand, B is tightly related to the theory of graph zeta functions [21], in particular the Ihara Zeta function, which is known to determine several graph properties such as girth, number of spanning trees, whether the graph is bipartite, regular, or a forest, etc [16]. Thus, both as a relaxed version of the original length spectrum, but also as an object of interest in itself, we find the eigenvalue spectrum of the non-backtracking matrix B to be an effective means of determining the dissimilarity between two graphs.

4. Non-Backtracking Matrix: Algorithm and Properties. For the rest of this work, we focus on the nonbactracking matrix and its properties. We now present a spectral analysis which will aid in the study of several aspects of NBD. We present an algorithm for computing B, and describe properties of B's eigenvalue distribution in connection with features of complex networks.

4.1. Computing B. Given a graph with n nodes and m undirected edges, define the $n \times 2m$ incidence matrices $P_{x,u\to v} = \delta_{xu}$ and $Q_{x,u\to v} = \delta_{xv}$, and write $C = P^T Q$. Observe that $C_{k\to l,u\to v} = \delta_{vk}$. Therefore,

$$B_{k\to l,u\to v} = C_{k\to l,u\to v}(1 - C_{u\to v,k\to l}) \tag{4.1}$$

Note that an entry of B may be positive only when the corresponding entry of C is positive. Therefore, we can compute B in a single iteration over the nonzero entries of C. Now, C has a positive entry for each pair of incident edges in the graph, thus we find $nnz(C) = O(n\langle k^2\rangle)$, where $\langle k^2\rangle$ is the second moment of the degree distribution, and $nnz(C)$ is the number of non-zero entries in C. Computing P, Q takes $O(m)$ time, so we can compute B in time $O(m + n\langle k^2\rangle)$. For example, in the case of a power-law degree distribution with exponent $2 \le \gamma \le 3$, the runtime of our algorithm falls between $O(m+n)$ and $O(m+n^2)$. Note that if a graph is given in adjacency list format, one can build B directly from the adjacency list in time $\Theta(n\langle k^2\rangle - n\langle k\rangle)$ by generating a sparse matrix with the appropriate entries set to 1 in a single iteration over the adjacency list.

4.2. Spectral Properties. Observe that the sparsity of B grows with the second moment of the degree distribution.

LEMMA 4.1. *Consider the non-backtracking matrix B of a graph G with n nodes and let $nnz(B)$ be the number of non-zero elements therein. Then,*

$$nnz(B) = n\big(\langle k^2\rangle - \langle k\rangle\big), \tag{4.2}$$

where $\langle k\rangle$ and $\langle k^2\rangle$ are the first and second moments of the degree distribution of G, respectively.

Proof. This is seen by using Equation 2.1 to sum over all the elements of B. ∎

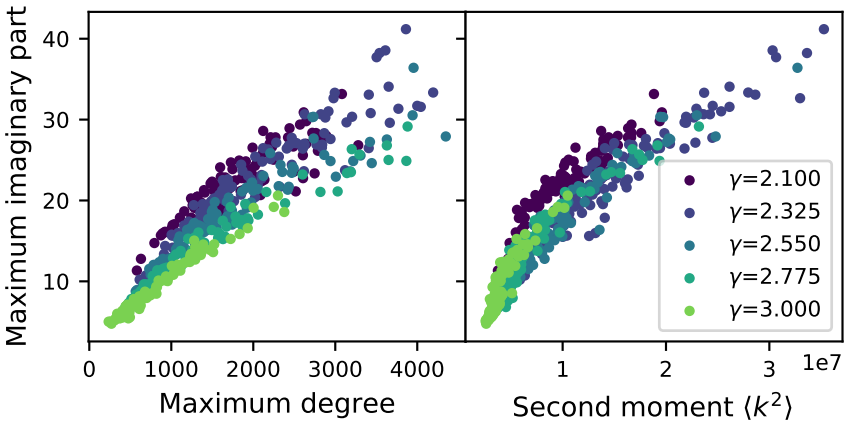

FIG. 3. *Imaginary part of non-backtracking eigenvalues, as a function of heterogeneity of degree distribution in the configuration model. Given a value of γ, we generated a random degree sequence that follows $p_k \sim k^{-\gamma}$ and generated a graph with said degree sequence using the configuration model. The maximum imaginary part of non-backtracking eigenvalues, as well as the sum total of their magnitudes (not pictured), increases as the heterogeneity in degrees increases, i.e., as there are larger hubs or the second moment of the degree distribution $\langle k^2 \rangle$ increases. All graphs have $n = 5 \times 10^4$ nodes and average degree 12; 100 graphs generated for each value of γ.*

Contrast this to $nnz(A) = n\langle k \rangle$, where A is the adjacency matrix of the graph. Experimentally, we have found that the larger $\langle k^2 \rangle$, the larger the variance of B's complex eigenvalues along the imaginary axis (Figure 3).

Next, we turn to B's eigenvalues and their relation to the number of triangles. Write $\lambda_k = a_k + ib_k \in \mathbb{C}$ for the eigenvalues of B, $k = 1, 2, .., 2m$. The number of triangles in a network is proportional to $tr(B^3) = \sum_k Re(\lambda_k^3)$,[4] which, by a direct application of the binomial theorem, equals

$$tr(B^3) = \sum_{k=1}^{2m} a_k(a_k^2 - 3b_k^2). \tag{4.3}$$

In general, the eigenvalues of B with small absolute value tend to fall on a circle in the complex plane [28, 3, 52, 11]. However, if $\sum_k a_k^2$ is large and $\sum_k b_k^2$ is small (implying a large number of triangles), the eigenvalues cannot all fall too close to a circle, since they will need to have different absolute values. Hence, the more triangles in the graph, the less marked the circular shape of the eigenvalues (Figure 1).

Finally, a note of practical importance on the spectrum of B. The multiplicity of the eigenvalue 0 equals the number of edges outside of the 2-core of the graph. For example, a tree, whose 2-core is empty, has all its eigenvalues equal to 0. On the one hand, we could use this valuable information as part of our method to compare two graphs. On the other hand, the existence of zero eigenvalues does not change the value of $tr(B^k), k \geq 0$, and thus leaves $\mathcal{L}'$, the relaxed length spectrum, intact. Moreover, removing the nodes of degree one reduces the size of B (or the sparsity of B', see Sec. 2.2), which makes the computation of non-zero eigenvalues faster.

[4] The imaginary part of this expression vanishes because the complex eigenvalues of a matrix always come in conjugated pairs.

5. Experiments. We discuss practical and algorithmic aspects of computing the Non-Backtracking spectral Distance (NBD), as well as explain how to fine tune it to be more sensitive to triangles and degree distribution. An implementation may be found at [50]. We also present experimental evidence of its discriminatory power when comparing random and real graphs.

5.1. Computing d. Given two graphs G, H and a positive integer r, we compute the distance between the two graphs in three steps; see Algorithm 5.1. First, remove all nodes of degree one from either graph. As mentioned previously, nodes of degree one do not affect the spectrum $\mathcal{L}'$, and their removal makes the computation faster. Note that after removing a node of degree one, another node's degree might decrease from 2 to 1. Thus, we need to iterate this removal until all nodes in the graph have degree at least 2 (this process is called "shaving" in graph mining, and yields the 2-core of the graph). Second, compute the block matrix B' (Sec. 2.2) from either graph and compute its largest r eigenvalues. Third, write these as $\lambda_k = a_k + ib_k$ for G and $\mu_k = \alpha_k + i\beta_k$ for H, where $|\lambda_k| \geq |\lambda_{k+1}|$ and $|\mu_k| \geq |\mu_{k+1}|$ for $k = 1, ..., r-1$, and assign to G the feature vector $v_1 = (\mathbf{a}, \mathbf{b}) = (a_1, a_2, ..., a_r, b_1, b_2, ..., b_r)$, and to H assign $v_2 = (\boldsymbol{\alpha}, \boldsymbol{\beta}) = (\alpha_1, \alpha_2, ..., \alpha_r, \beta_1, \beta_2, ..., \beta_r)$. Finally, compute the distance between G and H as $\|v_1 - v_2\|$, where $\|\cdot\|$ is the Euclidean norm. See Figure 4 (left column) for results of applying this distance measure to random graph models.

Algorithm 5.1 Non-Backtracking spectral Distance

Input: Graphs G, H, positive integer r
Output: real number d, distance between G, H

1: $\tilde{G}, \tilde{H} \leftarrow \text{shave}(G), \text{shave}(H)$
2: $\{\lambda_k\}_{k=1}^r, \{\mu_k\}_{k=1}^r \leftarrow$ largest eigenvalues of B' corresponding to $\tilde{G}, \tilde{H}$
3: $v_1, v_2 \leftarrow (a_1, .., a_r, b_1, .., b_r), (\alpha_1, .., \alpha_r, \beta_1, .., \beta_r)$,
for $\lambda_k = a_k + ib_k$ and $\mu_k = \alpha_k + i\beta_k$, with $k = 1, ..., r$
4: $d \leftarrow \|v_1 - v_2\|$
5: **return** d

5.2. Fine-tuning. One advantage of NBD is that it can be fine tuned to capture the features that we discussed in Section 4.2. For instance, if the number of triangles is of particular interest, one may accentuate the effect of equation 4.3 as follows. If one increases the sum of squares of the real parts and decreases the sum of squares of imaginary parts, one artificially increases the number of triangles in the graph. Hence, to emphasize differences in number of triangles, one can compute the distance using the modified feature vectors

$$v_1' = (\sigma\mathbf{a}, \mathbf{b}/\sigma), v_2' = (\sigma\boldsymbol{\alpha}, \boldsymbol{\beta}/\sigma), \tag{5.1}$$

for some real number $\sigma \geq 1$. We have also observed experimentally that the spread of the imaginary parts of the eigenvalues increases as the second moment of the degree distribution increases. Hence, if the degree distribution is of interest, one may emphasize this effect by using instead the feature vectors

$$v_1' = (|\lambda_1|^\eta a_1, ..., |\lambda_r|^\eta a_r, |\lambda_1|^\eta b_1, ..., |\lambda_r|^\eta b_r)$$
$$v_2' = (|\lambda_1|^\eta \alpha_1, ..., |\lambda_r|^\eta \alpha_r, |\lambda_1|^\eta \beta_1, ..., |\lambda_r|^\eta \beta_r)$$

with $\eta \in \mathbb{R}$ and $\eta > 0$. See Fig. 4 for an example of using these modifications when comparing the random graphs shown in Fig. 1.

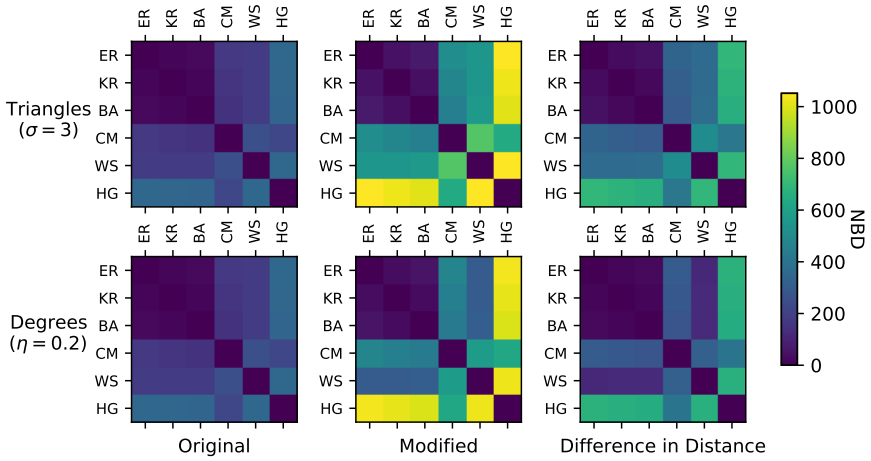

FIG. 4. *Fine-tuning NBD on various random graphs. Graphs are named with the same convention as in Fig. 1. Left column: original (unmodified) NBD between the average eigenvalue vectors of the graphs. Middle column: modified NBD fine-tuned to triangles (top row) and degree distribution (bottom row). Right column: difference between original and modified NBD values. Observe that after fine-tuning to triangles, the distance to HG is increased the most since HG has by far the most triangles across all random graph models used. Similarly, when fine-tuning for degrees, both CM and HG are emphasized since they have-heavy tailed degree distributions. (At this number of nodes, $n = 5 \times 10^4$, the degree distribution of BA is not as heavy-tailed.) See Sec. 5.2 for definitions of tuning parameters σ and η.*

5.3. Case Study 1: Clustering Random Graphs. In the first case study, we compute the NBD between random graphs generated with different random graph models in order to find clusters corresponding to said models. We use a Gaussian mixture which we optimize with the Expectation Maximization (EM) algorithm ([37], Ch. 11). Since the Gaussian probability density function assigns likelihood based on the distance from an arbitrary point to the mean of the distribution, this setup explicitly uses the NBD to perform the learning task. In this case study, our purpose is to showcase the effectiveness of the NBD at distinguishing between random graphs, as well as the fine-tuning mechanisms presented in Section 5.2, in an unsupervised learning setting. We do not perform an exhaustive sweep of parameter space.

The experimental setup is as follows. We generate 50 graphs of each of six different random graph models, for a total of 300 graphs. Each graph has 5×10^4 nodes and approximate average degree $\langle k \rangle = 15$ (see Fig. 1). The random graph models used were Erdős-Rényi (ER) [18, 10], Stochastic Kronecker Graph (KR) [31, 45], Barabási-Albert (BA) [5], Configuration Model with power law degree distribution with exponent $\gamma = 2.3$ (CM) [38], Watts-Strogatz (WS) [51], and Hyperbolic Graph with degree distribution exponent $\gamma = 2.3$ (HG) [27, 2]. We compute the largest $r = 200$ eigenvalues of each graph. For each graph $j = 1, .., 300$, we generate the vector $v_j = (a_1, .., a_r, b_1, .., b_r)$, where $\lambda_k = a_k + ib_k$, $k = 1, ..., r$ are the eigenvalues of the non-backtracking matrix. We use Kernel Principal Component Analysis ([37], Ch. 14) on the set of vectors $\{v_j\}$ to reduce the number of dimensions of the data set to two, for visualization purposes. We used used cosine similarity as kernel. Next, we employ the EM algorithm to estimate data density in 2D space and predict from which Gaussian component each graph is most likely to have come (Fig. 5).

Using the unmodified distance, the results are as follows: three clusters are easily

discernible (CM, HG, WS), while the other three (BA, ER, KR) are not quite so well defined (Figure 5a). However, as explained in Section 5.2, we can use the interpretable geometric features of the eigenvalue distribution to improve this result. We know that ER and BA differ greatly in their degree distribution; specifically, BA will have large second moment of the degree distribution, $\langle k^2 \rangle$, at large number of nodes. However, the number of nodes used here (5×10^4) may not be enough to show this feature. Therefore, we need to emphasize this feature and make the distance measure more sensitive to $\langle k^2 \rangle$ by using Equation 5.2. We find that a value of $\eta = 0.6$ gives the desired result: the cluster corresponding to BA graphs is more easily discernible from ER, KR than before (Figure 5b). Furthermore, we know that KR and ER differ in the number of expected triangles. Thus, using Equation 5.1, we find the parameter $\sigma = 11$ that makes ER and KR graphs distinguishable (Figure 5c). The combination of these two fine-tuned parameters allows us to recover with great accuracy the random models originating the graphs (Figure 5d). The best accuracy achieved across all random initializations of the experiment was 98.66%.

We wish to put this result in the context of other state-of-the-art graph distance methods. For example, the authors of [8] claim their method is able to cluster certain random graphs models with no misclassification errors when the graphs have $N = 50$ nodes. The methods ORTHOP and ORTHFR in [8] are a direct relaxation of the graph isomorphism problem. They try to find a perfect node alignment between two graphs by computing a linear transformation that converts the adjacency matrix of one graph to the adjacency matrix of the other graph. If such a transformation is found and has small matrix norm, the graphs are considered close. The algorithms ORTHOP and ORTHFR differ in the type of transformations that they seek. Since they are relaxations of the isomorphism problem, we expect them to perform quite well in this experiment.

In this context, we wish to study the performance of NBD as the number of nodes varies. Can the NBD come close to ORTHOP and ORTHFR when clustering small graphs? More generally, how far is NBD from identifying the isomorphism class of (the 2-core of) a graph, which was the original promise of the theory of the length spectrum? For this purpose, we execute the same experimental setup as above but on increasingly smaller graphs and compare to an approximation of the graph isomorphism problem, namely ORTHOP and ORTHFR [8]. See Fig. 5e for results. NBD can achieve comparable performance to ORTHOP/ORTHFR at $N = 5 \times 10^4$, while achieving acceptable performance across all other graph sizes (when using fine-tuning parameters). We hypothesize that the drop in performance of NBD in small graphs is due to the fact that small graphs have few eigenvalues (with smaller absolute values than eigenvalues of large graphs), which yields patterns on the complex plane that are difficult to distinguish. However, when the graphs are large, the increase in number of eigenvalues yields geometric patterns that are more easily distinguishable.

5.4. Case Study 2: Dissimilar Samples of the Same Graph. In this case study we take several samples of the same real life network with different sampling algorithms, and measure the distance between them with the purpose of determining which samples were taken with the same algorithm. In doing so we also show the visualization capabilities afforded by the non-backtracking eigenvalues (Fig. 6).

For this experiment, we use the graph of web pages belonging to Stanford University [32]. This graph has $n = 281,903$ nodes and $m = 2,312,497$ edges. We took two samples with each of the following sampling algorithms: node sampling (NS), edge sampling (ES), random walk sampling (RW), random walk sampling with jump

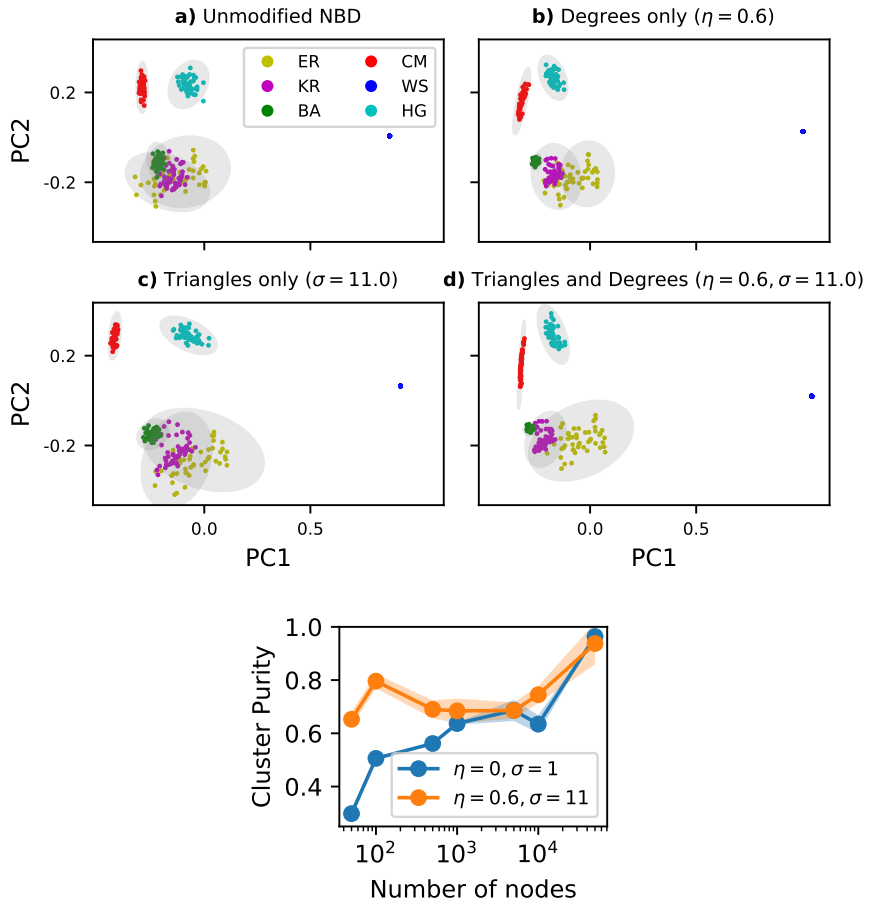

FIG. 5. *Using NBD to cluster random graphs. We compute the largest $r = 200$ eigenvalues random graphs of six different models. Graphs are named with the same convention as in Fig. 1. We plot the first two principal components of the data after applying Kernel PCA with cosine similarity. The clusters (gray ellipses) are found with the Expectation Maximization algorithm optimizing a Gaussian mixture model. We shown results using the unmodified NBD in (**a**), fine tuned to emphasize only degree distribution in (**b**), fine tuned to only triangles in (**c**), and fine tuned to both in (**d**), where all but four data points out of 300 are clustered correctly (cluster purity 98.66%). All graphs have $n = 5 \times 10^4$ nodes and average degree approximately $\langle k \rangle = 15$. See definitions of η, σ in Sec. 5.2. Performance of random graph clustering as number of nodes varies is shown in (**e**), using unmodified NBD and fine-tuned with same parameters as in (d). Authors of [8] claim their methods achieve 100% cluster purity at $n = 50$ nodes (not pictured). Variance is due to stochasticity of random graph models and random initializations of the clustering algorithm.*

(RJ) [1], for a total of eight samples. The samples were taken from random seeds until a minimum of 5% of existing edges were observed. Jump probability for RJ was $p = 0.3$. After visualizing the non-backtracking eigenvalues of each sample graph, we observe there are regions of the complex plane that are consistently occupied by only two of these samples at the same time. However, visualization of the eigenvalues on the complex plane (Fig. 6a) or in a reduced space (Fig. 6b) does not unambiguously determine which samples come from the same algorithm. Therefore, we proceed to apply the Kolmogorov-Smirnov statistical test to determine which samples were taken

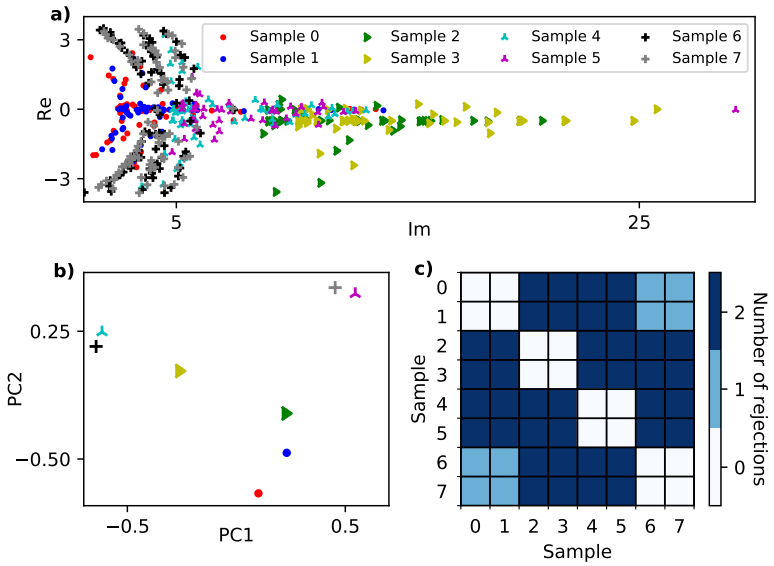

FIG. 6. *(Best viewed in color.) Visualizing distinct samples of the Stanford web page graph [32]. We take eight samples of the same data set and plot the largest $r = 200$ non-backtracking eigenvalues of each sample. Samples taken with the same algorithm share the same marker type. Visual analysis of* ***(a)*** *suggests that the pairs of samples that come from the same algorithm are as follows: samples 2 and 3, samples 0 and 1, samples 6 and 7, and samples 4 and 5. However, Kernel PCA with cosine similarity does not support this hypothesis,* ***(b)****. Hence, we rely on the Kolmogorov-Smirnov statistic to test the hypothesis that each pair of samples comes from the same underlying distribution. In* ***(c)****, we show the number of tests in which we reject the null hypothesis for each pair of samples. Here we confirm that for those pairs of samples identified in* ***(a)****, the null cannot be rejected, while there is enough evidence to reject all other pairs.*

with the same algorithm. We assume that the underlying original network determines a continuous probability distribution over the complex plane with support set $\mathcal{S}$, and that each sampling algorithm determines a distinct probability distribution over $\mathcal{S}$. We assume, further, that the eigenvalues of each sample network are independent observations drawn from these distributions. Hence, to answer the question of which of those samples are taken from the same distribution, we use the Kolmogorov-Smirnov test on each pair of two samples under the null hypothesis that the samples come from the same distribution. All tests performed at 90% significance level and with Bonferroni correction for multiple comparisons of $m = 14$ (each sample is compared to the seven other samples twice: one for the distribution of real parts of the eigenvalues and one for the imaginary part). (Fig. 6c) shows that this test is capable of determining which samples are taken from the same algorithm.

5.5. Case Study 3: Degree-Preserving rewiring. In this case study, we observe the performance of NBD in the presence of structural noise in the graph, introduced through degree-preserving rewiring [35]. Given a graph G, degree-preserving rewiring is a method of obtaining a different graph H at random with the same degree distribution as G. The purpose is to elucidate whether NBD is detecting structural features that can be detected by relying solely upon the degree distribution. If a graph G is close, in NBD terms, to all others with the same degree distribution, then using NBD is not any better than comparing the degree distributions of graphs.

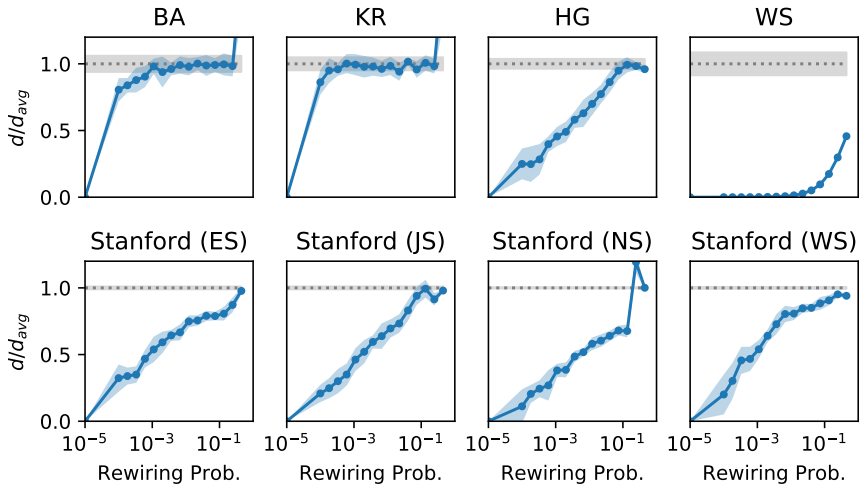

FIG. 7. *Performance of NBD in the presence of structural noise (degree-preserving rewiring). Distances are normalized with respect to the average distance, d_{avg}, between the original graph and an ensemble of graphs with the same degree distribution (dashed line); standard deviation is shown in shaded gray. Normalized distance between the original graph and rewired versions, by percentage of rewired edges (blue). Shaded regions show one standard deviation around the mean. In most cases, NBD is able to distinguish between the original graph and noisy versions across several orders of magnitude of the rewiring probability, in some cases even after 20% of the edges have been rewired. BA and KR are indistinguishable from other graphs with the same degree distribution, which highlights the need to use fine-tuning in applications.*

The experimental setup is as follows. We consider a graph G, and compute its non-backtracking eigenvalues. Then, we generate a sample of random graphs with the same degree distribution as G using the configuration model. We compute the average NBD distance from G to graphs in the sample. If the NBD between G and the sample is small, then NBD is only detecting structure present in the degree distribution and this would be a counter-indication to the use of NBD in this case. Moreover, we introduce structural noise to G by performing degree-preserving randomization [35]. We rewire a fraction of the edges of G and measure the distance between the rewired graph and G. By varying the rewiring parameter (i.e., the probability of rewiring an edge), we expect that the rewired versions of G will move closer and closer to the random sample of configuration model graphs. Hence, we expect the distance to increase from 0 to the average distance to the random sample computed previously.

Fig. 7 shows that NBD is able to distinguish the original graph from noisy versions across a wide range of the rewiring parameter in several cases. However, both BA and KR are indistinguishable from other graphs with the same degree distribution, even with a small rewiring parameter. This partly explains the observation in Case Study 1 that NBD was not able to satisfactorily distinguish between KR and BA graphs before fine-tuning.

5.6. Case Study 4: Enron data set. In this last case study we apply the NBD to the well-known Enron emails data set [24, 33, 48, 49]. From it, we extract a who-emails-whom network, treat it as undirected and unweighted, and aggregate it both daily and weekly; see Fig. 8. The purpose is to recover general common sense features of this data set, such as the periodicity of weekly communications, as well as perform anomaly detection: we expect to see anomalies in the distance measured

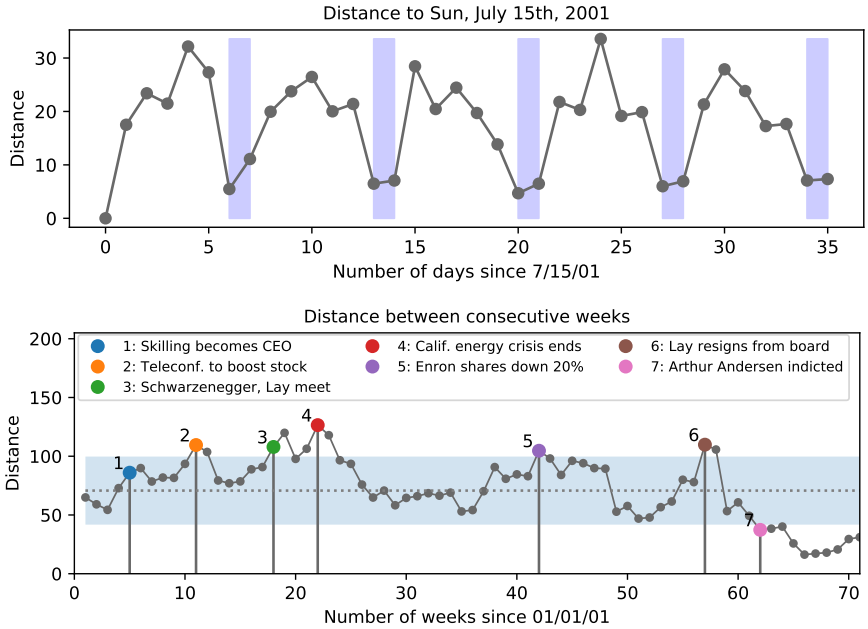

FIG. 8. *(Best viewed in color.) Applying NBD to the Enron data set.* ***Top:*** *Data aggregated into daily graphs and compared to one arbitrarily chosen milestone (Sunday, July 15th, 2001). The periodicity of weekly communications is recovered; that is, graphs corresponding to Saturdays and Sundays are closer to each other than they are to weekdays.* ***Bottom:*** *Data aggregated into weekly graphs and compared to the previous week. We highlight the mean distance (dashed line) and one standard deviation around it (shaded area). Each week that falls outside of the shaded area coincides with a known event during the Enron scandal and subsequent collapse.*

between graphs of this data set whenever a major event in the Enron scandal occurred. We were able to recover both of these features (Fig. 8).

6. Conclusions. We have focused on the problem of deriving a notion of graph distance for complex networks based on the length spectrum function. We add to the repertoire of distance measures the Non-Backtracking spectral Distance (**NBD**): a principled, interpretable, computationally efficient, and effective measure that takes advantage of the fact that one can interpret the non-backtracking cycles of a graph as its free homotopy classes. NBD is principled because it is backed by the theory of the length spectrum, which characterizes the 2-core of a graph up to isomorphism. It is interpretable because we can study its behavior in the presence of structural features such as hubs and triangles, and we can use the resulting geometric features of the eigenvalue distribution to our advantage. It is efficient because it takes no more time than computing a few of the largest eigenvalues of the non-backtracking matrix. Lastly, we have presented extensive experimental evidence to show that it is effective at discriminating between complex networks in various contexts, including visualization, clustering, sampling, and anomaly detection. An implementation of NBD, as well as our algorithm for computing the non-backtracking matrix, is available at [50].

Limitations. There are two major limitations of NBD. First, it relies on the assumption that the non-backtracking cycles contain enough information about the network. Concretely, the usefulness of the NBD will decay as the 2-core of the graph gets smaller. For example, trees have an empty 2-core, and all of its non-backtracking

eigenvalues are equal to zero. In order to compare trees, and more generally, those nodes outside the 2-core of the graph, the authors of [17] propose several different strategies, for example adding a "cone node" that connects to every other node in the graph. However, many real-world networks are not trees and we extensively showcased the utility of NBD on this class of networks in Section 5. Second, definition 3.1 is one possible way to compare two length spectrum functions, but variations are possible. One can possibly improve NBD by choosing a different ordering of eigenvalues (other than by absolute value), or by performing different preprocessing steps to those explored here (such as Kernel PCA with cosine similarity in Sections 5.3 and 5.4).

Future work. There are many other avenues to explore in relation to how to exploit the information stored in the length spectrum and the fundamental group. As mentioned in Sec. 3, the major downside of the relaxed length spectrum $\mathcal{L}'$ is the fact that we lose information stored in the combinatorics of the fundamental group. That is, $\mathcal{L}'$ stores information of the frequency of lengths of free homotopy classes, but no information on their concatenation – i.e., the group operation in $\pi_1(G)$. One way to encapsulate this information is by taking into account not only the frequency of each possible length of non-backtracking cycles, but also the number of non-backtracking cycles of fixed lengths ℓ_1 and ℓ_2 that can be concatenated to form a non-backtracking cycle of length ℓ_3. It remains an open question whether this information can be computed using the non-backtracking matrix for all values of the parameters ℓ_1, ℓ_2, ℓ_3, and if so, how to do it efficiently. One alternative is to rely upon efficient motif counting [40, 25].

In this work, we have focused on introducing and exploiting novel theoretical concepts such as the length spectrum and the fundamental group to the study of complex networks. We hope this work paves the road for more research in topological and geometric data analysis in network science.

Acknowledgments. We thank Evimaria Terzi for her contributions to an earlier version of this work. Torres and Eliassi-Rad were supported by NSF CNS-1314603 and NSF IIS-1741197. Suárez-Serrato was supported by UC-MEXUS (University of California Institute for Mexico and the United States) CN-16-43, DGAPA-UNAM PAPIIT IN102716, and DGAPA-UNAM PASPA program.

REFERENCES

[1] N. K. Ahmed, J. Neville, and R. R. Kompella, *Network sampling: From static to streaming graphs*, TKDD, 8 (2013), pp. 7:1–7:56.
[2] R. Aldecoa, C. Orsini, and D. Krioukov, *Hyperbolic graph generator*, Comput. Phys. Commun., 196 (2015), pp. 492–496.
[3] O. Angel, J. Friedman, and S. Hoory, *The non-backtracking spectrum of the universal cover of a graph*, Trans. Amer. Math. Soc., 367 (2015), pp. 4287–4318.
[4] J. P. Bagrow and E. M. Bollt, *An information-theoretic, all-scales approach to comparing networks*, preprint, arXiv:1804.03665 [cs.SI], (2018).
[5] A.-L. Barabási and R. Albert, *Emergence of scaling in random networks*, Science, 286 (1999), pp. 509–512.
[6] H. Bass, *The Ihara-Selberg zeta function of a tree lattice*, Internat. J. Math., 3 (1992), pp. 717–797.
[7] V. Batagelj and M. Zaversnik, *Fast algorithms for determining (generalized) core groups in social networks*, Adv. Data. Anal. Classi., 5 (2011), pp. 129–145.
[8] J. Bento and S. Ioannidis, *A family of tractable graph distances*, in Proceedings of the 2018 SIAM International Conference on Data Mining (SDM), San Diego, CA, USA., 2018, pp. 333–341.
[9] M. Berlingerio, D. Koutra, T. Eliassi-Rad, and C. Faloutsos, *Network similarity via multiple social theories*, in Advances in Social Networks Analysis and Mining (ASONAM),

Niagara, ON, Canada, 2013, pp. 1439–1440.
[10] B. Bollobás, *Random Graphs*, Cambridge Studies in Advanced Mathematics, Cambridge University Press, 2nd ed., 2001.
[11] C. Bordenave, M. Lelarge, and L. Massoulié, *Non-backtracking spectrum of random graphs: Community detection and non-regular Ramanujan graphs*, in 56th Annual Symposium on Foundations of Computer Science (FOCS), Berkeley, CA, USA., 2015, pp. 1347–1357.
[12] S. Chowdhury and F. Mémoli, *Distances and isomorphism between networks and the stability of network invariants*, preprint, arXiv:1708.04727 [cs.DM], (2017).
[13] S. Chowdhury and F. Mémoli, *The metric space of networks*, preprint, arXiv:1804.02820 [cs.DM], (2018).
[14] A. Clauset, E. Tucker, and M. Sainz, *The Colorado index of complex networks*, 2016, https://icon.colorado.edu/ (accessed 2018-06-19).
[15] D. Constantine and J.-F. Lafont, *Marked length rigidity for one-dimensional spaces*, J. Topol. Anal., (2018). In press.
[16] Y. Cooper, *Properties determined by the Ihara zeta function of a graph*, Electron. J. Combin., 16 (2009), pp. 14, Research Paper 84.
[17] C. Durfee and K. Martin, *Distinguishing graphs with zeta functions and generalized spectra*, Linear Algebra Appl., 481 (2015), pp. 54–82.
[18] P. Erdös and A. Rényi, *On the evolution of random graphs*, Publ. Math. Inst. Hung. Acad. Sci., 5 (1960), p. 17.
[19] E. Estrada, *Spectral moments of the edge adjacency matrix in molecular graphs, 1. definition and applications to the prediction of physical properties of alkanes*, J. Chem. Inf. Comp. Sci., 36 (1996), pp. 844–849.
[20] L. Gulikers, M. Lelarge, and L. Massoulié, *Non-backtracking spectrum of degree-corrected stochastic block models*, in 8th Innovations in Theoretical Computer Science (ITCS), Berkeley, CA, USA., 2017, pp. 44:1–44:27.
[21] K. Hashimoto, *Zeta functions of finite graphs and representations of p-adic groups*, in Automorphic forms and geometry of arithmetic varieties, vol. 15 of Adv. Stud. Pure Math., 1989, pp. 211–280.
[22] A. Hatcher, *Algebraic topology*, Cambridge University Press, 2017.
[23] F. Jiang, L. He, Y. Zheng, E. Zhu, J. Xu, and P. S. Yu, *On spectral graph embedding: A non-backtracking perspective and graph approximation*, in Proceedings of the 2018 SIAM International Conference on Data Mining (SDM), San Diego, CA, USA., 2018, pp. 324–332.
[24] B. Klimt and Y. Yang, *Introducing the enron corpus*, in First Conference on Email and Anti-Spam, (CEAS) 2004, Mountain View, CA, USA., 2004.
[25] T. G. Kolda, A. Pinar, and C. Seshadhri, *Triadic measures on graphs: The power of wedge sampling*, in Proceedings of the 13th SIAM International Conference on Data Mining (ICDM), Austin, Texas, USA., 2013, pp. 10–18.
[26] D. Koutra, N. Shah, J. T. Vogelstein, B. Gallagher, and C. Faloutsos, *DeltaCon: Principled massive-graph similarity function with attribution*, TKDD, 10 (2016), pp. 28:1–28:43.
[27] D. Krioukov, F. Papadopoulos, M. Kitsak, A. Vahdat, and M. Boguñá, *Hyperbolic geometry of complex networks*, Phys. Rev. E, 82 (2010), p. 036106.
[28] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborová, and P. Zhang, *Spectral redemption in clustering sparse networks*, Proc. Natl. Acad. Sci. USA, 110 (2013), pp. 20935–20940.
[29] J. Kunegis, *KONECT: The Koblenz network collection*, in 22nd International World Wide Web Conference, (WWW), Rio de Janeiro, Brazil, 2013, pp. 1343–1350.
[30] C. J. Leininger, D. B. McReynolds, W. D. Neumann, and A. W. Reid, *Length and eigenvalue equivalence*, Int. Math. Res. Not. IMRN, 2007 (2007), p. rnm135.
[31] J. Leskovec, D. Chakrabarti, J. M. Kleinberg, C. Faloutsos, and Z. Ghahramani, *Kronecker graphs: An approach to modeling networks*, J. Mach. Learn. Res., 11 (2010), pp. 985–1042.
[32] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, *Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters*, Internet Math., 6 (2009), pp. 29–123.
[33] R. Marks, *Enron timeline*, 2008, http://www.agsm.edu.au/bobm/teaching/BE/Enron/timeline.html (accessed 2018-06-06).
[34] T. Martin, X. Zhang, and M. E. J. Newman, *Localization and centrality in networks*, Phys. Rev. E, 90 (2014), p. 052808.
[35] S. Maslov and K. Sneppen, *Specificity and stability in topology of protein networks*, Science,

296 (2002), pp. 910–913.
[36] J. R. Munkres, *Topology*, Prentice Hall, 2nd ed., 2000.
[37] K. P. Murphy, *Machine learning: A probabilistic perspective*, The MIT Press, 2012.
[38] M. E. J. Newman, *The structure and function of complex networks*, SIAM Rev., 45 (2003), pp. 167–256.
[39] J.-P. Onnela, D. J. Fenn, S. Reid, M. A. Porter, P. J. Mucha, M. D. Fricker, and N. S. Jones, *Taxonomies of networks from community structure*, Phys. Rev. E, 86 (2012), p. 036104.
[40] A. Pinar, C. Seshadhri, and V. Vishal, *ESCAPE: efficiently counting all 5-vertex subgraphs*, in Proceedings of the 26th International Conference on World Wide Web (WWW), Perth, Australia, 2017, pp. 1431–1440.
[41] V. M. Preciado, A. Jadbabaie, and G. C. Verghese, *Structural analysis of Laplacian spectral properties of large-scale networks*, IEEE Trans. Automat. Contr., 58 (2013), pp. 2338–2343.
[42] P. Ren, R. C. Wilson, and E. R. Hancock, *Graph characterization via Ihara coefficients*, IEEE T. Neural Nerwor., 22 (2011), pp. 233–245.
[43] A. Saade, F. Krzakala, and L. Zdeborová, *Spectral density of the non-backtracking operator on random graphs*, EPL (Europhys. Lett.), 107 (2014), p. 50005.
[44] T. A. Schieber, L. Carpi, A. Díaz-Guilera, P. M. Pardalos, C. Masoller, and M. G. Ravetti, *Quantification of network structural dissimilarities*, Nat. Commun., 8 (2017), p. 13928.
[45] C. Seshadhri, A. Pinar, and T. G. Kolda, *An in-depth analysis of stochastic Kronecker graphs*, J. ACM, 60 (2013), pp. 13:1–13:32.
[46] S. Soundarajan, T. Eliassi-Rad, and B. Gallagher, *A guide to selecting a network similarity method*, in Proceedings of the 2014 SIAM International Conference on Data Mining (SDM), Philadelphia, PA, USA., 2014, pp. 1037–1045.
[47] A. Terras, *Zeta functions of graphs: A stroll through the garden*, Cambridge Studies in Advanced Mathematics, Cambridge University Press, Cambridge, 2011.
[48] The Guardian, *Timeline: Enron*, 2006, https://www.theguardian.com/business/2006/jan/30/corporatefraud.enron (accessed 2018-06-06).
[49] The New York Times, *Timeline: A chronology of enron corp.*, 2006, https://www.nytimes.com/2006/01/18/business/worldbusiness/timeline-a-chronology-of-enron-corp.html (accessed 2018-06-06).
[50] L. Torres, *SuNBEaM: Spectral Non-Backtracking Eigenvalue pseudo-Metric, a topological graph distance*. https://github.com/leotrs/sunbeam, 2018.
[51] D. J. Watts and S. H. Strogatz, *Collective dynamics of small-world networks*, Nature, 393 (1998), p. 440.
[52] P. M. Wood and K. Wang, *Limiting empirical spectral distribution for the non-backtracking matrix of an Erdös-Rényi random graph*, preprint, arXiv:1710.11015 [math.PR], (2017).