MULTISCALE MODEL. SIMUL.
Vol. 22, No. 3, pp. 925–955




# RANKING EDGES BY THEIR IMPACT ON THE SPECTRAL COMPLEXITY OF INFORMATION DIFFUSION OVER NETWORKS*

JEREMY KAZIMER†, MANLIO DE DOMENICO‡, PETER J. MUCHA§, AND DANE TAYLOR¶

**Abstract.** Despite the numerous ways now available to quantify which parts or subsystems of a network are most important, there remains a lack of centrality measures that are related to the complexity of information flows and are derived directly from entropy measures. Here, we introduce a ranking of edges based on how each edge's removal would change a system's von Neumann entropy (VNE), which is a spectral-entropy measure that has been adapted from quantum information theory to quantify the complexity of information dynamics over networks. We show that a direct calculation of such rankings is computationally inefficient (or unfeasible) for large networks, since the possible removal of $M$ edges requires that one compute all the eigenvalues of $M$ distinct matrices. To overcome this limitation, we employ spectral perturbation theory to estimate VNE perturbations and derive an approximate edge-ranking algorithm that is accurate and has a computational complexity that scales as $\mathcal{O}(MN)$ for networks with $N$ nodes. Focusing on a form of VNE that is associated with a transport operator $e^{-\beta L}$, where $L$ is a graph Laplacian matrix and $\beta > 0$ is a diffusion timescale parameter, we apply this approach to diverse applications including a network encoding polarized voting patterns of the 117th U.S. Senate, a multimodal transportation system including roads and metro lines in London, and a multiplex brain network encoding correlated human brain activity. Our experiments highlight situations where the edges that are considered to be most important for information diffusion complexity can dramatically change as one considers short, intermediate, and long timescales $\beta$ for diffusion.

**Key words.** network science, information diffusion, perturbation theory, spectral complexity, von Neumann entropy, edge ranking

**MSC codes.** 94A17, 05C82, 60J60, 28D20, 68P30

**DOI.** 10.1137/22M153135X

## 1. Introduction.

Centrality analysis [10, 83]—i.e., determining the importance of substructures including nodes [11, 36, 48], edges [47, 74, 95, 116], and subgraphs [37, 114]—is a fundamental pursuit of network science. Quantifying their importance supports diverse applications ranging from the ranking of webpages [14, 69], athletes/athletic teams [16, 87, 97], and academics/academic institutions [19, 29, 99, 112],

*Received by the editors October 31, 2022; accepted for publication (in revised form) May 9, 2024; published electronically July 9, 2024.
https://doi.org/10.1137/22M153135X
**Funding:** The work of the first and fourth authors was supported by National Science Foundation grants DMS-2052720 and EDT-1551069 and by Simons Foundation grant 578333. The second author was supported by the Istituto Nazionale di Fisica Nucleare grant "LINCOLN" and the European Union funding within the MUR PNRR "National Center for HPC, BIG DATA AND QUANTUM COMPUTING" (Project CN00000013 CN1). The work of the third author was supported by National Science Foundation grant BCS-2140024.

†Department of Mathematics, University at Buffalo, State University of New York (SUNY), Buffalo, NY 14260 USA (jeremykazimer@gmail.com).
‡CoMuNe Lab, Department of Physics and Astronomy "Galileo Galilei," University of Padua, 35131, Italy, and Istituto Nazionale di Fisica Nucleare, Sez. Padova, Italy (manlio.dedomenico@gmail.com).
§Department of Mathematics, Dartmouth College, Hanover, NH 03755 USA (peter.j.mucha@dartmouth.edu).
¶School of Computing & Department of Mathematics and Statistics, University of Wyoming, Laramie, WY 82072 USA, and Department of Mathematics, University at Buffalo, State University of New York (SUNY), Buffalo, NY 14260 USA (dane.taylor@uwyo.edu).

to the identification of potential intervention targets for dynamical processes over networks: examples include congestion points for transportation systems [54, 58], influencers in social networks [67, 68, 72], vaccination strategies [96, 100], as well as points of fragility for critical infrastructures [56, 95] and biological networks [2, 59]. Given the broad impact of this interdisciplinary field, new methodological advances continue to be developed including the extension of centralities to generalized types of networks (e.g., temporal networks [86, 112], multilayer and multiplex networks [25, 26, 27, 104, 113, 118], hypergraphs and simplicial complexes [8, 38, 119]), comparing various centralities for specific applications [61, 108], deriving centralities that cater to particular dynamical systems [95, 115], and building stronger theoretical foundations using tools from statistics [23], mathematics [35], and machine learning [51].

Herein, we aim to further align the study of centrality with information theory and entropy measures, and in particular, to contribute to the growing interface between network science and quantum information theory [4, 24, 25, 43, 45, 76, 88]. For a recent overview of the topic, we refer the reader to [44]. Our approach is partly motivated by node entanglement [46], which was recently proposed as a node centrality measure stemming from the study of a network's von Neumann entropy (VNE) [123]. VNE was originally developed as an information-theoretic measure to quantify disorder within quantum-mechanical systems, and it has played a crucial role in the development of quantum information theory [123, 126]. More recently, it has been extended as a spectral-entropy measure that quantifies the structural complexity of graphs [13, 25, 76], has been generalized to preserve the subadditivity property [24] as later clarified in section 2.2, and has been applied to a variety of interconnected systems, from molecular biology [42] to neuroscience [7]. Of particular relevance is previous research showing that VNE is a natural measure to quantify the complexity of the interplay between structure and information diffusion over a network [44, 45]. We also note in passing that there exist other centrality measures [18, 75, 85, 94, 120, 121] that also utilize entropy in some way, but which are not derived from VNE nor relate to the complexity of information diffusion. Generally speaking, the literature relating edge centrality measures to entropy is lacking, and in particular, no centrality measure for edges has been previously derived using VNE.

In this work, we develop a VNE-based measure for edge importance. (Although our methods easily extend to nodes and subgraphs, we will not explicitly use them so here.) We develop a framework that ranks edges based on how VNE would change if each edge were separately removed. We interpret these rankings from the perspective of information diffusion over networks [44, 45] so that the node rankings are directly associated with the spectral complexity of information diffusion. The top-ranked edge is the one whose removal would most increase VNE, thereby most increasing the spectral complexity of information diffusion. As such, our approach complements the numerous existing centralities that relate to information-spreading dynamics over networks including betweenness [41], Katz centrality [64], communicability [34, 36], and many others [9, 35, 67, 68, 72]. These existing methods typically aim to identify nodes/edges that are important either as sources for broadcasting information or as crucial bottlenecks for information pathways, and as such, they do not provide a notion of centrality as it relates to the complexity/heterogeneity of *information diffusion*. Our work highlights how the extension of some concepts used in quantum information theory to network science, and centrality analysis in particular, can provide complementary insights into the importance of network substructures (e.g., edges) with respect to a system's functional complexity and how their roles change as one considers different timescales for information diffusion.

Our methodology is also motivated by prior centrality analyses [80, 95, 114, 116] that utilize spectral perturbation theory to more efficiently estimate the impact of node, edge, or subgraph removals on the perturbed spectral properties of networks (or functions thereof [115]). Specifically, as a spectral-entropy measure, the computation of VNE requires calculation of the full set of eigenvalues for a graph's Laplacian matrix. (See section 2 for our general definition.) Computing all the eigenvalues of a large matrix is well known to be computationally expensive, and so naively recomputing the eigenvalues to obtain the new VNE that occurs after an edge removal is computationally demanding and is infeasible for large networks. (See section 3.3 for numerical evidence and a discussion.) Thus motivated, we employ spectral perturbation theory to efficiently approximate changes to VNE and approximate the subsequent ranking of edges based on those perturbations. We show that the computational complexity of the approximate rankings in the limit of large $N$ reduces to at most $\mathcal{O}(N)$ per edge, allowing the framework to effectively scale to large networks.

We investigate the true and approximate rankings of edges in terms of VNE change for three empirical networks: a voting-similarity network for the 117th U.S. Senate, a multimodal transportation network encoding roads and metro lines, and a multiplex brain network with two layers that represent correlated brain activity at different frequency bands. We focus on a version of VNE based on a transport operator $e^{-\beta L}$, where $L$ is a graph Laplacian matrix and $\beta > 0$ is a *diffusion timescale parameter.* Interestingly, we find that our measure for a network's most important edges is significantly influenced by $\beta$ in that the top-ranked edges at a short timescale are often very different from those at an intermediate and/or long timescale. This result is interesting because it highlights the multiscale nature of importance for fundamental units of a network. Our numerical experiments reveal three different types of network structures that systematically cause such a reordering effect: community structure in the U.S. Senate voting network, the relative speed between roads and metro lines for the transportation network, and the strength of coupling between network layers for the multiplex brain network. These experiments also highlight the potential broad utility of our proposed entropy-based rankings for diverse applications across the social, physical, and biological sciences.

The remainder of this paper is organized as follows: we present background information in section 2, methodology in section 3, numerical experiments in section 4, and a discussion in section 5. A codebase that reproduces our findings can be found at [66].

## 2. Background information.

We first provide background information Laplacian matrices and their spectral properties (section 2.1), VNE as a measure of spectral complexity for information diffusion (section 2.2), and prior centrality measures that stem from spectral perturbation theory (section 2.3).

### 2.1. Laplacian matrices and the diffusion equation.

Let $G(\mathcal{V}, \mathcal{E})$ be a graph with a set $\mathcal{V} = \{1, \ldots, N\}$ of nodes and a set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ of undirected edges, each of which can have a positive weight $W_{ij} > 0$. We assume that there are no self-edges. The graph can be equivalently defined by a symmetric adjacency matrix $A$ with entries $A_{ij} = W_{ij}$ if $(i, j) \in \mathcal{E}$ and $A_{ij} = 0$ otherwise. We define $M = \frac{1}{2}\sum_{ij} A_{ij}$ as the total weight of edges. (The factor of $1/2$ arises since each edge appears twice in the matrix $A$.) For unweighted graphs, $A_{ij} = A_{ji} = 1$ for each edge $(i, j)$ so that $M$ equals the number of edges. We further define $D = \mathrm{diag}[d_1, \ldots, d_N]$, where each $d_i = \sum_j A_{ij}$ encodes the weighted degree (i.e., strength) of node $i$. The unnormalized Laplacian matrix (also called *combinatorial Laplacian*) is given by a size-$N$ square matrix

$$L = D - A. \tag{2.1}$$

The matrix $L$ is important to many diverse applications including graph partitioning [39], analysis of spanning trees [78], synchronization of nonlinear dynamical systems [89, 102, 115], electricity flow [31], manifold learning [6, 21], harmonic analysis [22], graph sparsification [106], and neural networks [15, 128]. Of particular relevance is that one can define the diffusion equation over a graph by

$$\frac{d}{dt}\mathbf{x}(t) = -\hat{\beta} L \mathbf{x}(t), \tag{2.2}$$

where $\mathbf{x}(t) = [x_1(t), \ldots, x_N(t)]$, $x_i(t)$ is the density at node $i$ at time $t$, and $\hat{\beta} > 0$ is a diffusion rate.

Theory development for many Laplacian-matrix-related applications often stems from studying the spectral properties (i.e., eigenvalues and eigenvectors) of $L$. As a real symmetric matrix, $L$ is diagonalizable, $L = U\Lambda U^T$, where $\Lambda = \text{diag}[\lambda_1, \ldots, \lambda_N]$ is a diagonal matrix whose diagonal entries encode the eigenvalues $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_N$, and their corresponding eigenvectors, $\mathbf{u}^{(k)}$, make up the orthonormal columns of $U = [\, \mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(N)}]$. We assume that the network is connected (i.e., $L$ is irreducible) so that $\lambda_2 > 0$. By construction, $L\,\mathbf{1} = 0$, where $\mathbf{1} = [1, \ldots, 1]^T$, which implies $\lambda_1 = 0$ is an eigenvalue and $\mathbf{u}^{(1)} = N^{-1/2}\mathbf{1}$ is its associated normalized eigenvector. To facilitate later analyses, we also define the vector of eigenvalues: $\vec{\lambda} = [\lambda_1, \ldots, \lambda_N]$. Using a spectral decomposition, (2.2) has the general solution

$$\mathbf{x}(t) = e^{-\hat{\beta} t L}\mathbf{x}(0) = U e^{-\hat{\beta} t \Lambda} U^T \mathbf{x}(0) = \sum_{k=1}^{N} \alpha_k e^{-\hat{\beta} t \lambda_k} \mathbf{u}^{(k)}, \tag{2.3}$$

where $\alpha_k = \langle \mathbf{u}^{(k)}, \mathbf{x}(0) \rangle$ is the projection of the initial condition onto the $k$th eigenvector (each of which is an invariant subspace of the diffusion dynamics). Herein, we will often combine the diffusion rate and time parameter into a single *timescale parameter* $\beta = \hat{\beta} t$.

Before continuing, we highlight that there exist normalized versions of Laplacians that are also widely studied. For example, $\tilde{L} = D^{-1}L$ is an asymmetric normalized Laplacian, and under the substitution $\tilde{L} \mapsto L$, (2.3) would describe a continuous-time Markov chain [40]. Similarly, $\hat{L} = D^{-1/2} L D^{-1/2}$ defines a symmetric normalized Laplacian that shares the same eigenvalues as $\tilde{L}$ and is broadly important for machine learning applications. In this work, we will focus on $L$, noting that our methodology and findings can be easily extended to applications involving these other matrices.

**2.2. Information diffusion and von Neumann entropy (VNE).** While the diffusion equation given by (2.2) is often associated with heat flow, it can also be used as a model for other types of diffusion including the propagation of quanta of information over complex networks [7, 43, 46]. We begin by using $\beta = \hat{\beta} t$ to define $Z_\beta(L) \equiv \text{Trace}(e^{-\beta L}) = \sum_{k=1}^{N} e^{-\beta \lambda_k}$, which normalizes a *propagator matrix* $e^{-\beta L}$ to yield an associated *density matrix*

$$\rho_\beta(L) = \frac{e^{-\beta L}}{Z_\beta(L)} = \sum_{k=1}^{N} s_k (\mathbf{u}^{(k)})^T (\mathbf{u}^{(k)}), \tag{2.4}$$

where $s_k = e^{-\beta \lambda_k} / Z_\beta(L)$. These expressions use the eigenvalues $\lambda_k$ and eigenvectors $\mathbf{u}^{(k)}$ of $L$, which were defined in the previous section.

We note that in the classical interpretation of diffusion, if a particle is first observed at node $i$ and later observed at node $j$, then it is believed to have taken some well-defined (although possibly unknown) path from $i$ to $j$. In contrast, the quantum

transport of a quantum particle that is emitted at node $i$ is considered to be delocalized (i.e., exists across all paths) until it is later detected at node $j$. In either case, one can interpret a vector $\mathbf{x}(t)$ to be an *information field* at timescale $t \geq 0$ for information dynamics that are initialized with an initial field $\mathbf{x}(0) = \sum_{i=1}^{N} p_i \mathbf{e}_i$, where $p_i$ is the probability that information is "seeded" at node $i$ and $\mathbf{e}_i$ is a unit vector in which all entries are zeros except for the $i$th entry, which is a one. We assume $\sum_i p_i = 1$. Each entry $[\rho_\beta(L)]_{ij}$ gives the probability of transport of information quanta from node $i$ to $j$ at timescale $\beta$. Moreover, each eigenspace-restricted transport operator $s_k(\ \mathbf{u}^{(k)})^T(\mathbf{u}^{(k)})$ defines a $k$th *information stream* over which information quanta can be transported. By construction, $\mathrm{Trace}(\rho_\beta(L)) = \sum_k s_k = 1$ so that each $s_k$ defines the probability of transport along the $k$th information stream at timescale $\beta$. For both classical and quantum diffusion it is natural to examine whether transport is localized onto a small set of information streams (i.e., a few eigenmodes) or whether it is dispersed across many information streams (i.e., many eigenmodes). Such an inquiry is said to examine the *spectral complexity* of information diffusion. Here, we focus on classical diffusion according to (2.2) but seek to utilize the toolset of quantum information theory to measure the spectral complexity of information diffusion, and we will later use it to measure the importance (i.e., centrality) of edges. To this end, we quantify the spectral complexity of information diffusion using von Neumann entropy (VNE). VNE was introduced by John von Neumann as a measure for quantum information [123] and can quantify, for example, the departure of a quantum-mechanical system from its pure state. Applications include using VNE to identify entangled spin-orbital bound states [127], examining nonequilibrium thermodynamics of bosons [79], and developing complementary analyses for quantum phenomena [57]. Due to its widespread applicability and inherent ability to capture uncertainty for entangled quantum states, VNE has become a cornerstone for modern quantum information theory [126].

Recently, this formalism was extended to studying structural information for graphs [13, 25, 76], although its more reliable generalization that preserves the subadditivity property of VNE was proposed later in [24] and further developed in [45]. (See [44] for a review). Applications have included the study of global trade networks [25] and the functional connectivity of brain networks [7, 84]. We will utilize the subadditive-preserving definition of VNE to develop an entropy-based centrality measure for edges in graphs.

We begin by defining a generalized notion of VNE for networks that incorporates and extends earlier versions.

DEFINITION 2.1 (von Neumann entropy for graphs). *Let $G(\mathcal{V}, \mathcal{E})$ be a graph and $L$ be an associated Laplacian matrix. Assume $L$ has the diagonalization $L = U\Lambda U^{-1}$, where $\Lambda = \mathrm{diag}(\vec{\lambda})$, $\vec{\lambda} = [\lambda_1, \ldots, \lambda_N]^T$ with $\lambda_i \in \mathbb{R}$, and $U$ contains the associated (right) eigenvectors as columns. Further, let $\rho(L)$ be a matrix-valued analytic function that admits the diagonalization $\rho(L) = U\mathrm{diag}[f_1(\vec{\lambda}), \ldots, f_N(\vec{\lambda})]U^{-1}$ and satisfies $1 = \mathrm{Tr}(\rho(L)) = \sum_i f_i(\vec{\lambda})$. We then define a general form for VNE of graph $G(\mathcal{V}, \mathcal{E})$ by*

$$\begin{aligned} h(L) &= -\mathrm{Tr}[\rho(L)\log_2 \rho(L)] \\ &= -\sum_{k=1}^{N} f_k(\vec{\lambda}) \log_2 f_k(\vec{\lambda}). \end{aligned} \tag{2.5}$$

*By convention, we define $0\log_2(0) = 0$.*

*Remark* 2.1. Because $1 = \sum_k f_k(\vec{\lambda})$, one can interpret each $f_k(\vec{\lambda})$ as a probability, and then VNE coincides with Shannon entropy for the set $\{f_k(\vec{\lambda})\}$ of probabilities.

*Remark* 2.2. We define a matrix-valued function $\rho(L)$ that satisfies $\text{Trace}(\rho(L)) = 1$ to be a "trace-normalized matrix transformation." In principle, matrix $L$ need not be restricted to Laplacian matrices, and it could represent any graph-encoding matrix that is diagonalizable and has real-valued eigenvalues.

The earliest application of VNE to graphs that we know of can be attributed to Braunstein, Ghosh, and Severini [13], wherein the authors restricted their attention to the choice of function $f_k(\vec{\lambda}) = \lambda_k / \sum_j \lambda_j$ for possibly weighted graphs, and we will refer to the associated VNE as *uniform VNE.* For unweighted graphs, $\sum_j \lambda_j = 2M$ gives twice the number of edges. However, it has been shown that this definition does not preserve the desired property of subadditivity for entropies [25]. More recently, De Domenico and Biamonte [24] proposed using $\rho(L) = \rho_\beta(L)$, as defined in (2.4), in which case $f_k(\vec{\lambda}) = s_k$, as defined for (2.4). Importantly, this definition does satisfy the subadditivity property, which states that $h(L^{(1)} + L^{(2)}) \leq h(L^{(1)}) + h(L^{(2)})$ for two graphs with equally sized Laplacian matrices $L^{(1)}, L^{(2)} \in \mathbb{R}^{N \times N}$.

Our definition of VNE above allows $L$ to represent either a normalized or unnormalized Laplacian matrix. Here, we focus on the choice of the combinatorial Laplacian matrix $L$ given by (2.1), and we refer to the corresponding VNE measure as *diffusion-kernel VNE.* We note in passing, however, that VNE has also been studied using a left normalized (also called random-walk) Laplacian matrix $\mathcal{L} = D^{-1}L$ [43], and we will refer to that form as *continuous-time random-walk VNE.* Regardless of the choice of matrix $L$, large VNE values indicate that diffusion transport is dispersed across many information streams (i.e., many eigensubspaces), whereas small VNE values indicate that it is localized to one, or a few, information streams (i.e., concentrated onto a smaller eigensubspace). See [7, 44, 45] for the exploration of other relationships between VNE and information diffusion dynamics, including discussions on trapped fields and information field diversity.

**2.3. Spectral perturbation theory for centrality analysis.** Given that diffusion-kernel VNE measures the spectral complexity of information diffusion, the remainder of this paper will focus on VNE-based analyses of networks. Specifically, we propose to rank edges (although our methods easily extend to nodes and subgraphs) according to how each edge's removal would change the graph's VNE.

Our work is largely motivated by prior centrality measures that quantify importance by considering how structural modifications lead to perturbations for the spectral properties of graph-encoding matrices. Of particular relevance is recent work [46] that proposed a node ranking by examining how VNE changes occur after node removals, and *node entanglement* was defined as a centrality measure inspired by the concept of quantum entanglement for quantum-mechanical systems. Herein, we will develop a complementary edge centrality measure that ranks edges based on how each edge's removal would change the graph's VNE. Importantly, centrality measures relating to spectral changes can be computationally expensive if many eigenvalues and eigenvectors are involved in the measures' definitions. In particular, VNE requires knowledge of all the eigenvalues, and so the direct recomputation of VNE for the Laplacian matrices of graphs after the removal of nodes or edges would be impractical for large graphs.

Given this limitation, herein we propose to build on existing techniques that leverage spectral perturbation theory to approximate centrality measures and rankings in a way that is more computationally efficient. For example, the *dynamical importance* [95] of a node or edge is defined as the decrease that would occur for the adjacency matrix's spectral radius if that node or edge is removed (see also [116]). In addition, one

can define the importance of a node by computing how much the PageRank or other centrality measures change upon the removal of that node [32]. Other related work includes considering the impact of structural modifications on the spectra of Laplacian matrices [76, 80, 105] as well as functions of such eigenvalues and eigenvectors [115]. Such centrality measures can be efficiently approximated by developing linear approximations for how structural perturbations of graphs give rise to spectral perturbations. We also refer the reader to [112, 113] for the use of more advanced matrix-perturbation techniques (namely singular perturbation theory) that have arisen in the context of centrality measures for multiplex and temporal networks. We note in passing that these perturbative analyses of networked-coupled dynamical systems are closely related to the notions of eigenvalue and eigenvector elasticities [62, 63].

Spectral perturbation theory for network modifications often stems from the following first-order approximation.

THEOREM 2.2 (perturbation of simple eigenvalues [5]). *Let $X$ be a symmetric $N \times N$ matrix with eigenvalues $\{\lambda_j\}$ and normalized eigenvectors $\{\mathbf{u}^{(j)}\}$. Consider an eigenvalue $\lambda_i$ that is simple in that it has algebraic multiplicity one: $\lambda_i \neq \lambda_j$ for any other $j$ and $\lambda_i$ has a one-dimensional eigenspace spanned by the eigenvector $\mathbf{u}^{(i)} \in \mathbb{R}^N$. Further, consider a fixed symmetric perturbation matrix $\Delta X$, and let $X(\epsilon) = X + \epsilon \Delta X$. We denote the eigenvalues and eigenvectors of $X(\epsilon)$ by $\lambda_j(\epsilon)$ and $\mathbf{u}^{(j)}(\epsilon)$, respectively. It then follows that*

$$\lambda_i(\epsilon) = \lambda_i + \epsilon \lambda_i'(0) + \mathcal{O}(\epsilon^2), \tag{2.6}$$

*where $\lambda_i'(0)$ is the derivative of $\lambda_i(\epsilon)$ with respect to $\epsilon$ at $\epsilon = 0$ and is given by*

$$\lambda_i'(0) = (\mathbf{u}^{(i)})^T \Delta X \mathbf{u}^{(i)}. \tag{2.7}$$

*Here, $\mathcal{O}(\epsilon^2)$ indicates that the difference between the right- and left-hand sides has an asymptotic $\epsilon \to 0$ scaling behavior that is upper bound by a constant multiple of $\epsilon^2$.*

*Proof.* For the proof, see [5]. ∎

*Remark* 2.3. The first-order approximation given by (2.6) is accurate when the perturbation is sufficiently small, $|\epsilon \lambda_i'(0)| \ll |\lambda_i|$.

To study how edge perturbations change the spectral properties of a Laplacian matrix $L$ given by (2.1), we can set $X = L$ and study a perturbation $\Delta L$ encoding the changed edge(s).

PROPOSITION 2.3 (weighted edge perturbations for Laplacian matrices). *When the weighted and undirected network is modified by adding an edge $(p, q)$ of weight $A_{pq}$, the Laplacian perturbation matrix takes the form*

$$\Delta L_{ij}^{(pq)} = \begin{cases} A_{pq}, & (i,j) \in \{(p,p),(q,q)\}, \\ -A_{pq}, & (i,j) \in \{(p,q),(q,p)\}, \\ 0 & \text{otherwise.} \end{cases} \tag{2.8}$$

*Similarly, when the network is modified by removing an edge $(p, q)$, the corresponding Laplacian perturbation matrix is $-\Delta L_{ij}^{(pq)}$. (We emphasize that in this notation we use $A_{pq}$ to be the (nonzero) weight of the edge when present—that is, the weight of the newly added edge or the weight of the edge prior to its removal.)*

Versions of Proposition 2.3 have appeared in various works, usually under the assumption of unweighted networks (see, e.g., [76, 115], in which case $A_{ij} \in \{0, 1\}$).

This form for $\Delta L$ can be combined with Theorem 2.2 to provide first-order approximations for the impact of network modifications on the eigenvalues of a Laplacian matrix.

LEMMA 2.4 (first-order spectral impact of edge additions and removals [80]). *Let $\Delta L^{(pq)}$ and $-\Delta L^{(pq)}$, respectively, denote the perturbation matrix for a graph Laplacian under the addition and removal of a weighted edge $(p,q)$ with weight $A_{pq}$ as defined in Proposition* 2.3. *Then the simple eigenvalues $\lambda_i(\epsilon)$ of the perturbed matrix $L \pm \epsilon \Delta L^{(pq)}$ are given by* (2.6) *with*

$$\lambda_i'(0) = \pm(\mathbf{u}^{(i)})^T \Delta L^{(pq)} \mathbf{u}^{(i)} = \pm A_{pq}(u_p^{(i)} - u_q^{(i)})^2. \tag{2.9}$$

In the next section, we extend this spectral approximation theory to diffusion-kernel VNE to develop VNE-based edge-ranking algorithms that are computationally efficient and can scale to large networks. It is also worth noting that the above-mentioned spectral perturbation theory has been previously used to study graph VNE [76], although that work did not investigate centrality and focused on a formulation in which $s_k = \lambda_k/2M$. Finally, we note that the above results easily extend to describing the first-order effects due to the modification of sets of edges [115], thereby allowing predictions for node and subgraph modifications (although that is not our focus here).

**3. VNE perturbations measure the importance of edges.** We now present our main theoretical and algorithmic results. Specifically, we introduce algorithms that rank edges according to their contribution to the spectral complexity of information diffusion over networks, as measured by diffusion-kernel VNE. In section 3.1, we develop these rankings and discuss their limitation to small graph sizes. In section 3.2, we develop an approximate-ranking algorithm that efficiently scales to large graphs and is based on spectral perturbation theory. In section 3.3, we present experiments to compare these two rankings, which are based on the true and approximate values for how VNE would be perturbed by edge removals. (Further experiments are deferred to section 4.)

**3.1. Edge rankings by VNE increases upon edge removals.** Given a graph $G(\mathcal{V},\mathcal{E})$ with $N$ nodes and $|\mathcal{E}|$ undirected edges with weights $\{A_{ij}\}$, we seek to rank the edges so that the most important edge $(p,q) \in \mathcal{E}$ is the one that would maximize the VNE of a residual graph that results from removing edge $(p,q)$. More precisely, let $L$ be the unnormalized graph Laplacian of $G(\mathcal{V},\mathcal{E})$ given by (2.1), and let $-\Delta L^{(pq)}$ be the change to $L$ given by (2.8) that would occur upon the removal of edge $(p,q) \in \mathcal{E}$. Further, let $h(L)$ be the graph's VNE, as given by (2.5). Then the change to VNE that would occur upon the removal of edge $(p,q)$ is given by

$$Q_{pq} = h(L - \Delta L^{(pq)}) - h(L). \tag{3.1}$$

In principle, $Q_{pq}$ is not necessarily positive, although our experiments suggest that $Q_{pq} > 0$ for most edges. Considering the set $\{Q_{pq}\}$ of VNE perturbations for $(p.q) \in \mathcal{E}$, the top-ranked edge is a solution to the following optimization problem:

$$(p,q) = \underset{(p,q)\in\mathcal{E}}{\operatorname{argmax}}\, Q_{pq}. \tag{3.2}$$

DEFINITION 3.1 (edge rankings by VNE increases). *Given a graph $G = (\mathcal{V},\mathcal{E})$ with unnormalized Laplacian $L$, consider the changes $Q_{pq}$ to VNE given by* (3.1), *which would occur upon the removals of edges $(p,q) \in \mathcal{E}$. Then, we define the rankings*

$$R_{pq} = 1 + |\hat{\mathcal{E}}|, \text{ where } \hat{\mathcal{E}} = \{(n,m) \in \mathcal{E} : Q_{nm} > Q_{pq}\}, \tag{3.3}$$

*so that $R_{pq} \in \{1,\dots,|\mathcal{E}|\}$ gives the rank of each edge $(p,q) \in \mathcal{E}$.*

**Algorithm 3.1 Edge Rankings by VNE Increases.**

**Require:** Graph $G(\mathcal{V}, \mathcal{E})$ with nodes $\mathcal{V}$, weighted edges $\mathcal{E}$, and unnormalized Laplacian $L$.

**Ensure:** Cardinality $|\mathcal{E}| > 0$

1: Compute set $\{\lambda_i\}$ of eigenvalues for $L$.
2: Compute VNE $h(L)$ according to (2.5).
3: **for** $(p, q) \in \mathcal{E}$ **do**
4: Compute the perturbation matrix $\Delta L^{(pq)}$ according to (2.8).
5: Compute the change $Q_{pq}$ according to (3.1) using $L$ and $\Delta L^{(pq)}$.
6: **end for**
7: Use (3.3) on the set $\{Q_{pq}\}$ to obtain the set $\{R_{pq}\}$ of rankings.

In practice, we implement these rankings efficiently using the NumPy function "argsort" in Python.

*Remark* 3.1. One can rank edges according to perturbations to VNE given by (2.5) for any choice of the function $\rho(\lambda)$. We will focus herein on the diffusion-kernel VNE in which $\rho(\lambda)$ is given by (2.4).

*Remark* 3.2. Note that the $Q_{pq}$ values for different edges $(p, q)$ can be the same, in which case the rankings are not unique. Such scenarios can be handled in a variety of ways. We resolve edges with tied rankings arbitrarily. That is, if $k$ edges are tied for rank $R$, then we arbitrarily assign them ranks $R$, $R+1, \ldots, R+k$.

Before continuing, we highlight that the direct computation of rankings $R_{pq}$ will be infeasible for large graphs due to the large computational cost. For example, in section 3.3 we estimate that the runtime to compute the set $\{Q_{pq}\}$ for a graph with $N = 8000$ nodes would take approximately 1300 hours (about 54 days) to complete on a standard desktop. More specifically, computing all eigenvalues $\{\lambda_i\}$ of a size-$N$ matrix $L$ is an expensive task having computational complexity that typically scales proportionally to $N^3$ (e.g., if using the QR method [49]). In order to compute the VNE change $Q_{pq}$ given by (3.1) for an edge $(p, q)$, one needs to first compute the perturbed eigenvalues $\{\lambda'_i\}$ for a perturbed Laplacian $L' = L - \Delta L^{(p,q)}$. Since this must be done for each edge, constructing the set $\{Q_{pq}\}$ for $(p, q) \in \mathcal{E}$ has computational complexity that is expected to scale proportionally to $|\mathcal{E}|N^3$. That said, developing numerical methods to compute the eigenvalues for large sparse matrices remains an active research field, and in section 5 we survey some leading approaches that may be useful to speed up Algorithm 3.1.

Nevertheless, it is impractical for large networks to numerically compute the eigenvalues for $|\mathcal{E}|$ distinct matrices of size $N$. Thus motivated, in the next section we develop an approximate ranking algorithm that is much more efficient and can be readily applied to larger graphs.

**3.2. Perturbation theory for VNE and efficient edge ranking.** Implementing Algorithm 3.1 can be computationally infeasible for large graphs, and here we develop approximate rankings based on the spectral perturbation theory that we presented in section 2.3. Recall that we define VNE (see Definition 2.1) using a spectral map $\mathbf{f}: \mathbb{R}^N \mapsto \mathbb{R}^N$ that is defined entrywise by

$$\mathbf{f}(\vec{\lambda}) = [f_1(\vec{\lambda}), \ldots, f_N(\vec{\lambda})]^T, \tag{3.4}$$

where $\vec{\lambda} = [\lambda_1, \ldots, \lambda_N]^T$ is a vector of eigenvalues. For uniform VNE, one has $f_k(\vec{\lambda}) = \lambda_k/(2M)$, which depends only on the $k$th eigenvalue since $\sum_k \lambda_k = 2M$ is a conserved quantity equal to total edge weight. In contrast, for diffusion-kernel VNE one has that $f_k(\vec{\lambda}) = s_k = e^{-\beta\lambda_k}/Z(\beta)$, as defined for (2.4). In this latter case, each $f_k(\vec{\lambda})$ depends on all the eigenvalues due to the denominator term $Z(\beta) = \sum_k e^{-\beta\lambda_k}$.

We now approximate how structural modifications change VNE to first order.

LEMMA 3.2 (first-order perturbation of VNE). *Let $L$ be a diagonalizable Laplacian matrix given by* (2.1) *for an undirected graph, and let $L' = L + \epsilon\Delta L$ denote its perturbation for a small scalar $\epsilon$. Further, let $h(L)$ be the graph's VNE given by Definition* 2.1. *Letting $H(\epsilon) \equiv h(L + \epsilon\Delta L)$ be the $\epsilon$-perturbed VNE, then $H(\epsilon)$ has the first-order approximation*

$$H(\epsilon) = H(0) + \epsilon H'(0) + \mathcal{O}(\epsilon^2), \tag{3.5}$$

*where $H(0) = h(L)$ is the VNE of the original graph. Moreover, when the eigenvalues $\lambda_i$ of $L$ are simple, then the derivative*

$$H'(0) = \sum_{i,j} \frac{\partial h}{\partial f_i} \frac{\partial f_i}{\partial \lambda_j} \frac{\partial \lambda_j}{\partial \epsilon}\bigg|_{\epsilon=0}$$

*can be constructed using partial derivatives with $\partial\lambda_j/\partial\epsilon|_{\epsilon=0} = \lambda_j'(0) = (\mathbf{u}^{(j)})^T \Delta L \mathbf{u}^{(j)}$ given by* (2.7) *in Theorem* 2.2. *Here, $\mathbf{u}^{(j)}$ is the eigenvector associated with $\lambda_j$.*

Lemma 3.2 can be generally applied to different choices for the density function $\rho(L)$ described in Definition 2.1. For example, it can recover a previously shown result for uniform VNE.

THEOREM 3.3 (first-order perturbation of uniform VNE [76]). *Consider a uniform density function $f_i(\vec{\lambda}) = \rho(\lambda_i) = \lambda_i/\sum_j \lambda_j$ and the full set of assumptions in Lemma* 3.2. *Then the perturbed VNE is given by* (3.5) *with*

$$H'(0) = -\frac{1}{2M} \sum_i (\mathbf{u}^{(i)})^T \Delta L \mathbf{u}^{(i)} \left[ \log_2 \left( \frac{\lambda_i}{\sum_j \lambda_j} \right) + \frac{1}{\ln(2)} \right]. \tag{3.6}$$

In [76], Theorem 3.3 was developed to study the effects of graph rewiring on uniform VNE, and in particular, how the distribution of VNE across a random-graph ensemble that is obtained via edge rewiring can converge to that for well-known random-graph ensembles, including ensembles associated with the Erdős–Rényi $G_{NM}$ and configuration models. Turning our attention back to diffusion-kernel VNE, we obtain the following.

THEOREM 3.4 (first-order perturbation of diffusion-kernel VNE). *Consider the diffusion-kernel density function $f_i(\vec{\lambda}) = s_i$, as defined according to* (2.4), *and the full set of assumptions in Lemma* 3.2. *Then the perturbed VNE is given by* (3.5) *with*

$$H'(0) = -\beta \sum_{i=1}^{N} f_i(\vec{\lambda}) \left[ \log_2(f_i(\vec{\lambda})) + \frac{1}{\ln(2)} \right] \left[ -\lambda_i'(0) + \sum_j^N f_j(\vec{\lambda}) \lambda_j'(0) \right], \tag{3.7}$$

*where $\lambda_i'(0) = (\mathbf{u}^{(i)})^T \Delta L \mathbf{u}^{(i)}$.*

*Proof.* For the proof, see Appendix A. ∎

In principle, a Laplacian perturbation matrix $\Delta L$ can encode a broad family of structural modifications to a graph. For example, Lemma 2.4 in section 2.3 showed that $\lambda_i'(0) = \pm(\mathbf{u}^{(i)})^T \Delta L^{(p,q)}\ \mathbf{u}^{(i)} = \pm A_{pq}(u_p^{(i)} - u_q^{(i)})^2$ for the addition (+) or removal (-) of an edge $(p,q)$ having weight $A_{pq}$. As shown in [115], a similar result can encode the simultaneous addition/removal for sets of edges. Thus, in principle one can compute (3.6) and (3.7) very efficiently for a given edge modification or a set of edge modifications. See Appendix B for an experiment that provides numerical validation for Theorem 3.4 by estimating the change to VNE induced by removing $k$ edges for $0 \leq k \leq 20$.

With this approximation theory in hand, we now return our attention to the ranking of edges based on the change $Q_{pq} = h(L - \Delta L^{(pq)}) - h(L)$ to VNE that would occur after the individual removal of each edge $(p,q) \in \mathcal{E}$. Combining Theorem 3.4 with Lemma 2.4, we approximate

$$Q_{pq} \approx \tilde{Q}_{pq} \equiv H'(0), \tag{3.8}$$

where $H'(0)$ is appropriately defined for the particular type of VNE (i.e., as represented by the spectral mapping $f_i(\vec{\lambda})$) and $\lambda_j'(0) = -A_{pq}(u_p^{(j)} - u_q^{(j)})^2$. Herein, we will focus on diffusion-kernel VNE in which $H'(0)$ is given by (3.7), which uses $f_i(\vec{\lambda}) = e^{-\beta\lambda_i}/\sum_j e^{-\beta\lambda_j}$. In principle, one can use a similar approach to approximate edge rankings based on VNE perturbations for other choices of $f_i(\vec{\lambda})$. Finally, we also highlight that due to the linearity of first-order approximations, one can also estimate the change to VNE that would occur due to the removal of a subset $\mathcal{E}^1 \subset \mathcal{E}$ of edges by $H'(0) = \sum_{(p,q)\in\mathcal{E}^1} Q_{pq}$.

Given the approximate perturbations $\{\tilde{Q}_{pq}\}$, we now define an approximate ranking of edges.

DEFINITION 3.5 (edge rankings by first-order approximate VNE increases). *Given a graph $G = (\mathcal{V}, \mathcal{E})$ with unnormalized Laplacian $L$, consider the approximate changes $\tilde{Q}_{pq}$ to VNE given by* (3.8).

*Then, we define the rankings*

$$\tilde{R}_{pq} = 1 + |\hat{\mathcal{E}}|, \text{ where } \hat{\mathcal{E}} = \{(n,m) \in \mathcal{E} : \tilde{Q}_{nm} > \tilde{Q}_{pq}\}, \tag{3.9}$$

*so that $\tilde{R}_{pq} \in \{1, \ldots, |\mathcal{E}|\}$ gives the approximate rank of each edge $(p,q) \in \mathcal{E}$.*

We summarize the computation of this ranking in Algorithm 3.2.

**Algorithm 3.2 Edge Rankings by First-Order-Approximate VNE Increases.**

**Require:** Graph $G(\mathcal{V}, \mathcal{E})$ with nodes $\mathcal{V}$, weighted edges $\mathcal{E}$, and unnormalized Laplacian $L$.
**Ensure:** Cardinality $|\mathcal{E}| > 0$
1: Compute set $\{\lambda_i\}$ of eigenvalues for $L$.
2: Compute VNE $h(L)$ according to (2.5).
3: **for** $(p,q) \in \mathcal{E}$ **do**
4: Compute the perturbation matrix $\Delta L^{(pq)}$ according to (2.8).
5: Compute the approximate change $\tilde{Q}_{pq}$ according to (3.8) using $L$ and $\Delta L^{(pq)}$.
6: **end for**
7: Use (3.9) on the set $\{\tilde{Q}_{pq}\}$ to obtain the set $\{\tilde{R}_{pq}\}$ of approximate rankings.

Before continuing, we highlight that one possible limitation of this approach is that Lemma 3.2, Theorem 3.3, and Theorem 3.4 define $H'(0)$ in a way that assumes that the eigenvalues $\lambda_j$ of $L$ are simple by virtue of Theorem 2.2. We propose to neglect the contributions of repeated eigenvalues as a simple heuristic. Moreover, we predict that if eigenvalue repetition occurs, it is more likely to arise for large eigenvalues (i.e., for which $e^{-\beta\lambda_j}$ is small), as opposed to the smaller, more important eigenvalues for which the $e^{-\beta\lambda_j}$ terms are larger. Therefore, we expect that the neglect of the $\lambda_i'(0)$ terms for repeated eigenvalues will have a particularly small effect on an estimate $H'(0)$ for diffusion-kernel VNE and the subsequent rankings of edges. However, while Algorithm 3.1 does not make any assumption about repeated eigenvalues, Algorithm 3.2 does make such an assumption which limits its usability. That said, we have yet to encounter repeated eigenvalues in our experiments presented here, but such situations are known to occur for some networks.

**3.3. Comparison of Algorithms 3.1 and 3.2.** Recall that our main motivation for developing approximate rankings was that it is can be computationally expensive (or infeasible) to directly compute the set $\{Q_{pq}\}$, which requires one to compute the exact change to VNE that occurs after removing each edge $(p, q)$. More specifically, upon each edge removal, we must recompute all eigenvalues of a size-$N$ matrix. Computing all the eigenvalues of a matrix is known to be computationally expensive, and such an algorithm must be implemented $M$ separate times for $M$ distinct matrices if one considers the removal of $M$ edges. The QR algorithm, for example, requires $\propto N^3$ flops [49], implying that the computation time grows proportionally to $N^3$ for a network with $N$ nodes.[1] (Of course, speeding the computation of spectra for large sparse matrices remains an active research field, and we highlight several leading approaches in section 5.) In contrast, the approximate rankings utilize spectral perturbation theory and the eigenvalues of a size-$N$ matrix only must be computed once. Then for each edge, one computes $\tilde{Q}_{pq} = H'(0)$ using (3.7), which has $\mathcal{O}(N)$ complexity and is a much more efficient approach. We also note for both algorithms that there is an additional start-up cost from having to compute the spectra of the original graph Laplacian.

In Figure 1, we compare Algorithms 3.1 and 3.2 in two ways. First, we show in Figure 1(A) that the approximate rankings $\{\tilde{R}_{pq}\}$ are much more efficient to compute than the true rankings $\{R_{pq}\}$ for an example graph. Second, we show in Figure 1(B) that these two sets of rankings can be very similar $\tilde{R}_{pq} \approx R_{pq}$. Their similarity is somewhat expected, since $\tilde{Q}_{pq} \approx Q_{pq}$ by construction.

In Figure 1(A), we support our predictions for the computational complexity for both algorithms by studying the scaling behavior for the algorithms' runtimes for graphs of increasing size $N$. We considered Erdős–Rényi $G_{NM}$ graphs with $N$ nodes and $M = N\langle d\rangle/2$ edges, where $\langle d\rangle = 10$ is the mean degree. We then generated a sequence of graphs with increasing size $N$, and for each graph we empirically estimated the runtime required to compute the rankings $R_{pq}$ and $\tilde{R}_{pq}$. For the approximate rankings $\{\tilde{R}_{pq}\}$, we computed them directly using Algorithm 3.2. However, we found that it was computationally infeasible to directly run Algorithm 3.1 for these graphs with sizes $N > 2000$, and so we instead estimated the algorithms' runtime by measuring the time required to compute $Q_{pq}$ for a single edge. Then we approximated

[1]Notably, we implemented our algorithms using the NumPy Python package, and the eigenpair computations are based on LAPACK [55], for which the asymptotic runtime complexity has a lower bound $\Omega(N^2)$ and upper bound $\mathcal{O}(N^3)$ [28]. Computations were conducted on a Dell Precision 3650 computer that has a 2.80 GHz Intel Xeon W-1390 processor with 64 GB RAM.

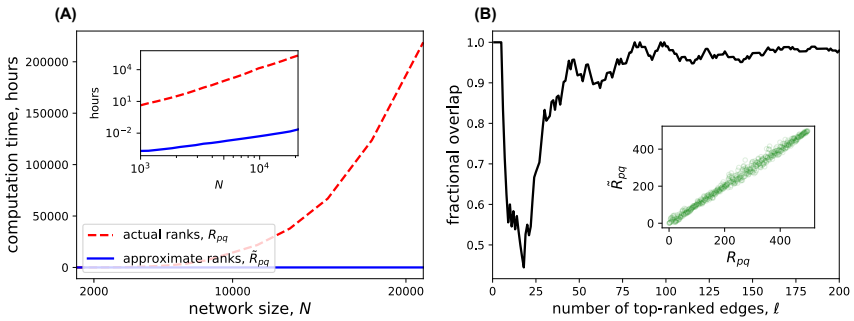


FIG. 1. Comparison of Algorithms 3.1 and 3.2. (A) *We estimate the runtimes for* $\beta = 1$ *that are required to execute these algorithms for Erdős–Rényi* $G_{NM}$ *graphs with increasing size with* $N$ *nodes and* $M = 5N$ *edges, so that mean degree remains fixed as* $N$ *increases. The inset is a logarithmic scale to reveal their runtimes' scaling behaviors (see text). Note that Algorithm* 3.1 *would potentially take hundreds of hours for each* $N$*, so we extrapolated these times from that required to compute for a single edge, instead of running the algorithm to completion. In contrast, Algorithm* 3.2 *completed in under a minute for all values of* $N$ *plotted.* (B) *We compare the fractional overlap* $|\mathcal{E}^{(\ell)} \cap \tilde{\mathcal{E}}^{(\ell)}|/\ell$ *for the* $\ell$ *top-ranked edges for the two rankings* $R_{pq}$ *and* $\tilde{R}_{pq}$ *in a* $G_{NM}$ *graph with* $N = 100$ *nodes and* $M = 1000$ *edges at* $\beta = 1$*. The inset shows that they are strongly correlated:* $R_{pq} \approx \tilde{R}_{pq}$ *for all edges* $(p, q) \in \mathcal{E}$*. We computed a Pearson correlation coefficient of* 0.996.

the total runtime by multiplying this duration by $M$, since the algorithm requires us to compute $Q_{pq}$ for all edges. We considered $N \in [2000, 20{,}000]$ and found that in all cases Algorithm 3.2 completes in less than a minute, whereas Algorithm 3.1 would take hundreds of hours to perform all of these computations (i.e., were we to attempt running it in its entirety). The subpanel in Figure 1(A) shows the runtimes in a log-log scale, and we applied least-squares linear fits to empirically estimate the scaling to be $\propto MN^{2.813}$ for Algorithm 3.1 and $\propto MN^{0.519}$ for Algorithm 3.2. We note that these observing scaling behaviors are slightly less than our predicted scaling behaviors: $\propto MN^3$ for Algorithm 3.1 and $\propto MN$ for Algorithm 3.2. Notably, our experiments were limited to graphs with $N = 20{,}000$ nodes, and we expect that the consideration of larger graphs would yield observed scaling rates closer to these asymptotic predictions.

In Figure 1(B), we provide evidence to support our claim that the approximate and true rankings are similar: $\tilde{R}_{pq} \approx R_{pq}$. Here, we computed the rankings $R_{pq}$ and $\tilde{R}_{pq}$ for an Erdős–Rényi $G_{NM}$ graph [33] with $N = 100$ nodes and $M = 1000$ edges. For each set of rankings, we considered $\ell$ top-ranked edges and measured the *fractional overlap,* also known as the *precision at* $\ell$ metric [20], between these two sets, or more precisely, the size of their overlap. That is, letting $\mathcal{E}^{(\ell)}$ and $\tilde{\mathcal{E}}^{(\ell)}$ be the sets of top-ranked edges according to $R_{pq}$ and $\tilde{R}_{pq}$, respectively, we plot $|\mathcal{E}^{(\ell)} \cap \tilde{\mathcal{E}}^{(\ell)}|/\ell$ versus the set size $\ell$. Note that this is a fraction that lies between 0 and 1, since the sets of top-ranked edges have the same cardinality: $\ell = |\mathcal{E}^{(\ell)}| = |\tilde{\mathcal{E}}^{(\ell)}|$. Observe in Figure 1(B) that this fraction is very large for many values of $\ell$, implying that the rankings $\{R_{pq}\}$ and $\{\tilde{R}_{pq}\}$ are very similar for this graph. This is further shown in the subplot, where we provide a scatter plot that compares $\tilde{R}_{pq}$ versus $R_{pq}$ for each edge $(p, q) \in \mathcal{E}$.

In Appendix C, we extend and recapitulate the experimental findings shown in Figure 1(B) by obtaining similar results for two empirical networks that will be later described in section 4. Given the strong similarity between rankings $\{R_{pq}\}$ and $\{\tilde{R}_{pq}\}$, and the observation that computing $\{R_{pq}\}$ directly using Algorithm 3.1 is inefficient

(or infeasible) for large graphs, for the remainder of the paper, we will focus our attention on studying the approximate rankings $\{\tilde{R}_{pq}\}$ given by Algorithm 3.2.

**4. Three empirical case studies reveal structural/dynamical mechanisms that influence the importance of edges.** Here, we apply our VNE-based measures for edge importance to three empirical network data sets: (section 4.1) a voting-similarity network for the U.S. Senate, (section 4.2) a multimodal transportation system, and (section 4.3) a multiplex brain network. In each empirical network, we will show that the edges that are deemed most important can drastically change by considering different timescale parameters $\beta$. In so doing, we will examine the crucial role that is played by the interplay between $\beta$ and some network property: (section 4.1) community structure due to political polarization, (section 4.2) edge weights that encode different speeds along roads and metro lines, and (section 4.3) the strength of coupling between layers of a multiplex network. In section 4.4, we compare our proposed centrality measure to other edge centralities for these three empirical case studies.

**4.1. U.S. Senate voting-similarity network with community structure.** In our first experiment, we study the importance of edges according to Algorithm 3.2 for a network that encodes voting similarity among U.S. Senators, and we will compare the rankings of *interparty edges* that connect Senators in different parties to the rankings of *intraparty edges* between Senators in the same party. More generally, we ask the following: which edges are most important in a graph that contains community structure, the edges between communities or the ones inside of communities? As we shall show, the answer will depend sensitively on the timescale of the dynamics considered.

For brevity, we defer a detailed description of the voting-similarity network that we study to Appendix D. Here, we provide a summary. We created a Python Module called *VoteView-python* [65] that extracts data from the VoteView repository [73] that summarizes the voting records of members of the U.S. Congress. Following techniques similar to those in [81, 82, 124], we constructed a graph with an associated adjacency matrix such that each entry $A_{ij}$ encodes the fraction of bills in which Senators $i$ and $j$ vote identically. We restrict our attention to the 117th U.S. Senate considering bills from January 3, 2021 until June 30, 2022 (noting that the 117th Congress had not yet concluded when we conducted this experiment.) We also thresholded the matrix, setting $A_{ij}$ to zero for any fraction less than 0.4. The resulting procedure yielded a graph with $N = 100$ nodes and $|\mathcal{E}| = 2,656$ undirected, weighted edges. Due to strong party polarization, the graph contains two large-scale communities that indicate, respectively, the Republican and Democratic parties. (Independents are incorporated into the community associated with the Democrats, in agreement with how those two Senators currently caucus.)

In Figure 2, we study the effect of $\beta$ on the rankings $\tilde{R}_{pq}$ given by Algorithm 3.2 for intralayer and interlayer edges for the voting-similarity network. In Figure 2(A), we visualize the voting-similarity network and use edge color (in grayscale) to depict the $\tilde{R}_{pq}$ values for three choices for $\beta \in \{10^{-2}, 10^{-1}, 10^{0}\}$. Senators' party affiliations (i.e., Republican versus Democrats and independents) are indicated by the node colors (red versus blue). Observe that the resulting network contains two well-separated communities due to party polarization. That said, there is an arrow pointing to a set of Senators {Susan Collins (R), Lisa Murkowski (R), Rob Portman (R), Mitt Romney (R), Shelley Capito (R), Lindsey Graham (R)} whose voting patterns are not as strongly polarized along party lines. These Senators are found to have many edges to

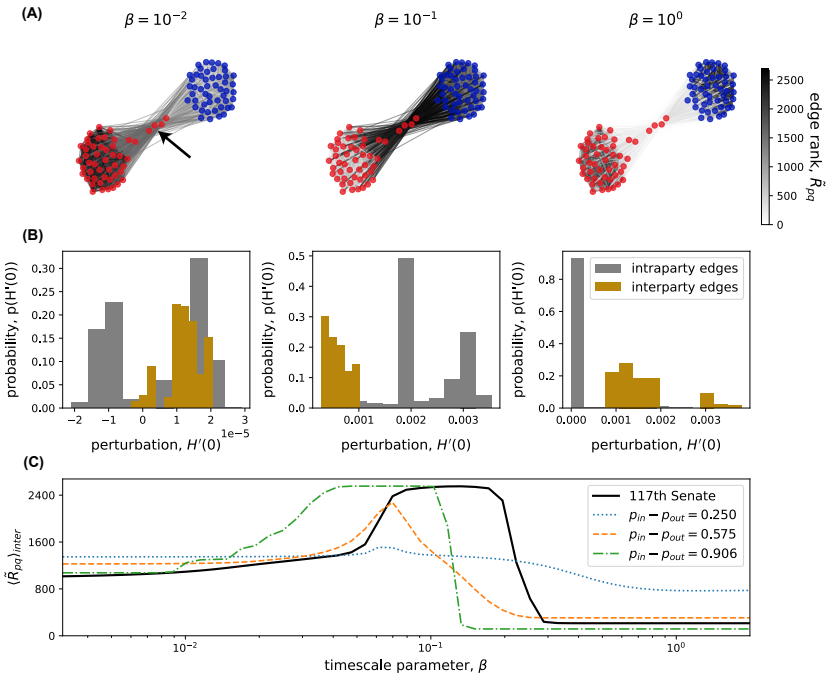


FIG. 2. Rankings of inter- and intraparty edges for a polarized political network. (A) *We visualize the rankings* $\tilde{R}_{pq}$ *of edges given by Algorithm* 3.2 *using grayscale for a graph that encodes voting similarity among the* 117*th U.S. Senate. The three columns depict three choices of timescale parameter* $\beta$*. Nodes represent Senators, and the node colors red and blue indicate, respectively, the two major political parties: Republicans and Democrats (including independents).* (B) *For the same three values of* $\beta$*, we plot empirically measured distributions* $p(H'(0))$ *of the approximate change* $H'(0)$ *to diffusion-kernel VNE given by* (3.7)*. In each panel, we show two distributions: one in which* $p(H'(0))$ *is measured across intraparty edges, and one in which* $p(H'(0))$ *is measured across interparty edges.* (C) *The solid black curve depicts the mean rank* $\langle \tilde{R}_{pq} \rangle_{inter}$ *across interlayer edges versus* $\beta$*. The colored curves indicate the values of* $\langle \tilde{R}_{pq} \rangle_{inter}$ *for a comparable random-graph model with two communities and different amounts of connectivity between communities (see text). Our main finding is that the interparty edges have the top rankings for larger* $\beta$*, the lowest rankings for intermediate* $\beta$*, and intermediate rankings for smaller* $\beta$*.*

both Republicans and Democrats, and as such, their node degrees are approximately 12x higher than those for other Senators.

Observe in the rightmost column of Figure 2(A) for $\beta = 1$ that the top-ranked edges are the interparty edges that connect Senators in different political parties. Interestingly, for $\beta = 10^{-1}$ (center column) the interparty edges are the ones with the lowest rankings, and for $\beta = 10^{-2}$ (left column) the rankings of interparty and intraparty edges are similar. This nonlinear effect of $\beta$ on $\tilde{R}_{pq}$ is further supported in panels (B) and (C). In Figure 2(B), we display a probability distribution $P(H'(0))$ of our approximate perturbations $H'(0)$ given by (3.7) for the same three values of $\beta$. Particularly, we separate the distributions according to interparty edges (gold) and intraparty edges (gray). For small $\beta$, these two distributions have a similar support. For medium $\beta$, we observe that the perturbations to VNE due to removing intraparty edges are larger than those for interparty edges. Similar to panel (A), we observe for large $\beta$ that the ordering flips—that is, the largest perturbations are now due to the removal of interparty edges as opposed to intraparty edges. These findings corroborate our network visualization in panel (A). Finally, in Figure 2(C) we use

a solid black curve to plot the average ranking $\langle \tilde{R}_{pq} \rangle_{inter}$ of interparty edges as a function of $\beta$. As expected based on our findings in panels (A) and (B), $\langle \tilde{R}_{pq} \rangle_{inter}$ takes on an intermediate value for small $\beta$, is very large for a medium value of $\beta$, and then is very small for large $\beta$.

To gain deeper insight, in Figure 2(C) we also plot $\langle \tilde{R}_{pq} \rangle_{inter}$ values for a comparable random-graph model that yield graphs with two communities, each containing 50 nodes (see the colored dotted, dot-dashed, and dashed curves). Specifically, we study a stochastic block model with $N = 100$ nodes and with two communities of equal size. Nodes $\{1, \ldots, 50\}$ are assigned to community 1, and nodes $\{51, \ldots, 100\}$ are assigned to community 2. Then, we create *intracommunity edges* between pairs of nodes in the same community uniformly at random with probability $p_{in}$, and *intercommunity edges* are created between pairs of nodes in different communities uniformly at random with probability $p_{out}$. We fix the mean $(p_{in} + p_{out})/2$ so that the expected number of edges, $N(N-1)(p_{in} + p_{out})/2$, matches that for the voting-similarity network. We then consider three choices for the difference $(p_{in} - p_{out}) \in \{0.25, 0.575, 0.911\}$, which is a measure for structural polarization. We note that the last value, 0.911, closely matches the empirically measured difference in edge probability for intraparty and interparty edges for the voting-similarity network. And in fact, stochastic block models are a popular graphical model for political polarization [101] and other sources for community structure [1, 91]. In Figure 2(C), we plot $\langle \tilde{R}_{pq} \rangle_{inter}$ (now corresponding to intercommunity edges) for three choices of $(p_{in} - p_{out})$. Observe for the larger two values of $(p_{in} - p_{out})$ that as $\beta$ increases, the average $\langle \tilde{R}_{pq} \rangle_{inter}$ exhibits a peak and then becomes very small. This phenomenon is very similar to what we observe for the voting-similarity network (black solid curve). Thus, this behavior for rankings $\tilde{R}_{pq}$ appears to occur due to the presence of community structure (that is, not due to some other possible structural property of the empirical voting-similarity network).

**4.2. Multimodal transportation system with roads and metro lines that have different speeds.** In our next case study, we study the importance of edges for a multimodal transportation network that encodes roads and metro lines in London [111]. In this data (which is available at [109]), 2217 vertices represent intersections and there are two types of edges (15 metropolitan lines and 2854 roads) to encode these two modes of transportation. Notably, the metro lines connect 11 major stations in London, and their locations are mapped to the nearest road intersection.

Given these two types of edges, in this section we study which has higher rankings, roads or metro lines, and we investigate how these rankings $\tilde{R}_{pq}$ change as one varies the relative speed between roads and metro lines. That is, in addition to studying how the rankings $\tilde{R}_{pq}$ change for different values of the timescale parameter $\beta$, we will also introduce *balancing parameter* $\chi \in [0, 1]$ that controls the extent to which metro lines are faster or slower than roads. More specifically, we define an undirected, weighted transportation network with an adjacency matrix

$$\mathbf{A} = \chi \mathbf{A}_{metro} + (1 - \chi)\, \mathbf{A}_{road}, \tag{4.1}$$

where $\mathbf{A}_{metro}$ and $\mathbf{A}_{road}$ are adjacency matrices in which edges only encode metro lines and roads, respectively. Note that there is no diffusion across metro lines (or roads) in the limit $\chi \to 0$ (or $\chi \to 1$). Finally, we note that it is important to study different relative speeds of roads and metro lines, since they have been shown, for example, to significantly influence the locations where congestions can occur [107].

In Figure 3(A), we provide a visualization of the multimodal transportation network with node locations reflecting their geospatial coordinates (i.e., latitudes and

longitudes). Thick and thin lines represent metro lines and roads, respectively, and the edge colors reflect their rankings $\tilde{R}_{pq}$ for the choices $\chi = 0.6$ and $\beta = 10^{-1}$. In Figure 3(B), we present a similar network visualization, except that we now consider a larger timescale parameter: $\beta = 10^{0}$. By comparing panels (A) and (B), observe for $\beta = 10^{-1}$ that the top-ranked edges are metro lines. In contrast, metro lines have the lowest rankings for $\beta = 10^{0}$.

In Figure 3(C), we plot the average ranking $\langle \tilde{R}_{pq} \rangle_{metro}$ across metro lines versus the timescale parameter $\beta$. The three thick colored curves reflect three choices for $\chi$. The thin curves depict the $\tilde{R}_{pq}$ values for each individual metro line. In Figure 3(D), we show similar information by plotting $\langle \tilde{R}_{pq} \rangle_{metro}$ versus $\chi$ for a few choices of $\beta \in [0.01, 3]$. Together, panels (C) and (D) highlight that the metro lines have the top rankings when $\beta$ is sufficiently small and $\chi$ is sufficiently large. That is, transport across metro lines must be sufficiently faster than that across roads, and one only considers diffusion at a short timescale.

From a topological perspective, metro lines introduce long-range, "short-cut" connections across a road network that is largely a two-dimensional geometric substrate [111]. We find it interesting that the importance of such long-range edges is so sensitive to timescale. We hypothesize that this phenomenon may be related to the

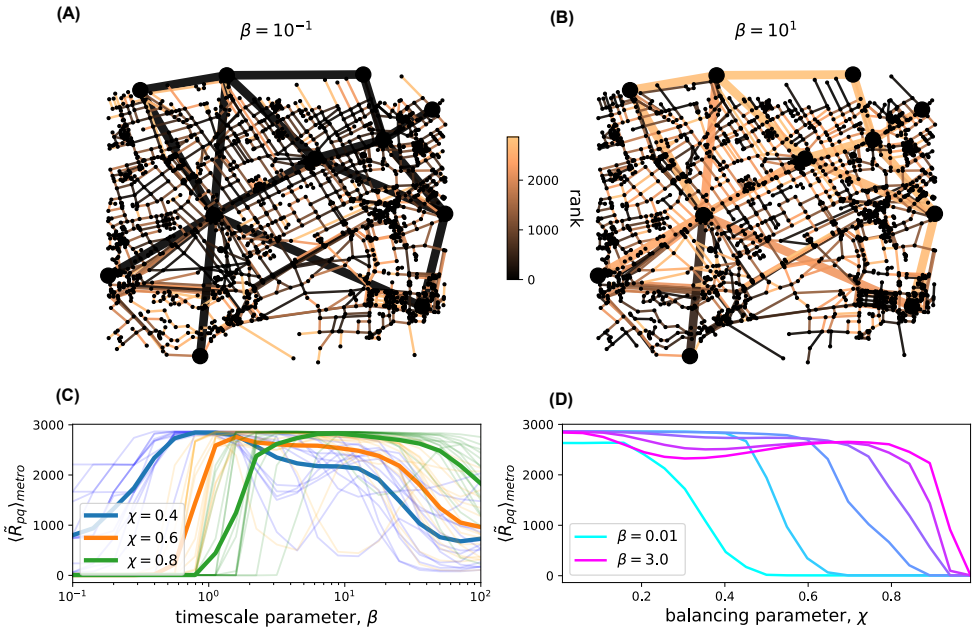


FIG. 3. Transportation network with metro lines and roads having different speeds. (A) *Visualization of a London transportation system in which metro lines and roads are depicted by thick and thin lines, respectively. Edge colors depict their rankings* $\tilde{R}_{pq}$ *for a small timescale* $\beta = 10^{-1}$. (B) *Same information as panel* (A), *except for a larger timescale* $\beta = 10^{0}$. *Observe that the metro lines have the highest rankings in* (A), *but the lowest rankings in* (B). (C) *Thick curves depict the average edge ranking* $\langle \tilde{R}_{pq} \rangle_{metro}$ *across metro lines versus* $\beta$ *for three choices of the tuning parameter* $\chi$ *(which controls whether diffusion is faster over metro lines or roads). Each thin curve indicates how the ranking* $\tilde{R}_{pq}$ *for each metro line varies with* $\beta$. (C) *Similar information as in panel* (C), *except we now plot* $\langle \tilde{R}_{pq} \rangle_{metro}$ *versus* $\chi$, *and different curves indicate several choices of* $\beta$. *In general, we find that metro lines (which introduce long-range "short-cuts" across the road network) are the top-ranked edges provided that* $\chi$ *is sufficiently large and* $\beta$ *is sufficiently small.*

following observation: if someone moves between two spatially distant nodes across a small timescale $\beta$, then they must have traversed a metro line. This is no longer true at a large timescale $\beta$, since there are many paths that one could take. Thus, one might expect short-cut edges to have a greater impact on diffusion dynamics (and rankings derived therefrom) at shorter timescales versus longer timescales.

**4.3. Multiplex brain network with interlayer coupling.** In our final case study, we study the importance of edges for a multiplex brain network in which nodes represent brain regions and different *network layers* encode coordinated brain activity patterns at different frequency bands. Our focus in this section is to investigate how the edge rankings $\tilde{R}_{pq}$ change as one varies the strength $\omega \geq 0$ of coupling between layers (as well as the timescale parameter $\beta$).

We study an empirical network obtained from [52] that was constructed from functional magnetic resonance imaging (fMRI) data in which nodes encode regions of interest (ROI) in human brains and edges encode a measure for spectral coherence between fMRI signals that are aggregated across each ROI [53]. In particular, the frequency content of an fMRI signal for each ROI was decomposed into different frequency bands, and then coherence relationships were identified for each frequency band. The result was a set of *network layers* in which each layer encodes spectral-coherence relationships at a particular frequency band.

Focusing on the frequency bands of $\delta$ waves (2–4 Hz) and $\theta$ waves (4.5–7.5 Hz), we considered two adjacency matrices $\mathbf{A}^{(\delta)}$ and $\mathbf{A}^{(\theta)}$, respectively. Both matrices are size $N = 148$ and encode two types of relationships among a common set of nodes $\mathcal{V} = \{1, \ldots, N\}$. That is, a nonzero matrix entry $A_{ij}^{(\delta)}$ encodes a relation between brain regions $i$ and $j$ at the $\delta$ frequency band, whereas $A_{ij}^{(\theta)}$ encodes a similar spectral-coherence relationship, but at the $\theta$ frequency band. To focus our study on sparse networks, we also applied a threshold to these matrices as described in Appendix E. This resulted in each layer containing $M = 1652$ undirected, weighted edges.

Given these two network layers encoded by $\mathbf{A}^{(\delta)}$ and $\mathbf{A}^{(\theta)}$, which we refer to as *intralayer adjacency matrices,* we then constructed a two-layer multiplex network by coupling them together. Specifically, we consider a *supra-adjacency matrix*

$$\mathbb{A} = \begin{bmatrix} \mathbf{A}^{(\delta)} & 0 \\ 0 & \mathbf{A}^{(\theta)} \end{bmatrix} + \omega \begin{bmatrix} 0 & \mathbf{I} & \mathbf{I} & 0 \end{bmatrix}, \tag{4.2}$$

where $\mathbf{I}$ is the identity matrix and $\omega \geq 0$ is an *interlayer coupling strength* that tunes how strongly the layers are coupled together. Coupling matrices in this way is called uniform, diagonal coupling.

As visualized in Figure 4(A), coupling network layers following (4.2) introduces *interlayer edges* (red dotted lines) that connect each node $i$ in the first layer to itself in the second layer. (The two network layers are indicated by the gray shaded regions, and the set of nodes is the same for both layers.) That is, there is one interlayer edge connecting each node in one layer to itself in the other layer. In contrast, we will refer to the edges encoded by $\mathbf{A}^{(\delta)}$ and $\mathbf{A}^{(\theta)}$ as *intralayer edges.* Notably, interlayer edges have a weight given by $\omega$, whereas the possible weights for intralayer edges are encoded by the matrix entries in $\mathbf{A}^{(\delta)}$ and $\mathbf{A}^{(\theta)}$.

Similar to our study in section 4.1 (where we examined the rankings of inter-party edges), we will now pay particular attention to the rankings $\tilde{R}_{pq}$ of interlayer edges. We ask the following: When do the interlayer edges have the top rankings? In Figure 4(B), we plot the average ranking $\langle \tilde{R}_{pq} \rangle_{inter}$ across interlayer edges versus the timescale parameter $\beta$. Different curves indicate different choices of the interlayer

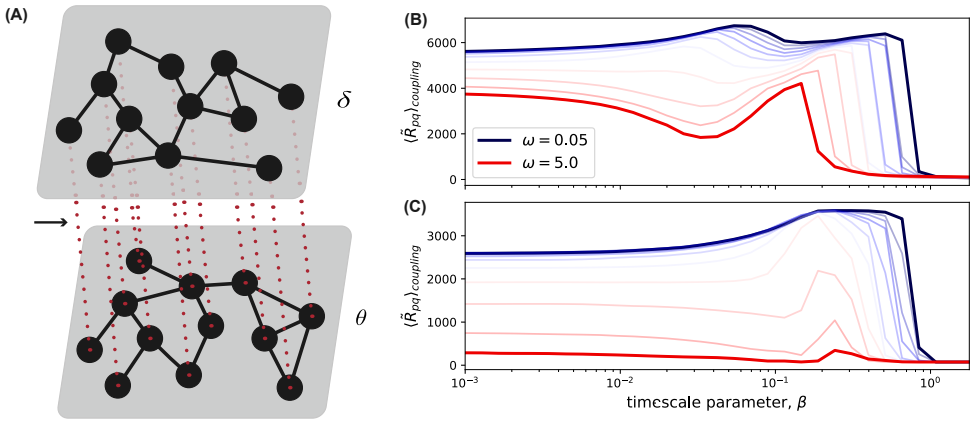


FIG. 4. Multiplex network representing coherent fMRI signals among brain regions. (A) *Toy illustration of a multiplex network with two layers representing two frequency bands: δ and θ waves. The solid black and red dotted lines represent, respectively, intralayer and interlayer edges. Interlayer edges are weighted by a coupling strength* $\omega \geq 0$. (B) *Average ranking* $\langle \tilde{R}_{pq} \rangle_{inter}$ *across interlayer edges versus* $\beta$ *for several choices of* $\omega \in [0.05, 5]$. (C) *Similar information as in panel* (B), *except we now consider a synthetic model in which each layer is an Erdős–Rényi* $G_{NM}$ *random graph in which* $N$ *and* $M$ *match the values for the empirical network layers. Observe a similar trend in panels* (A) *and* (B): *interlayer edges have the top rankings when* $\beta$ *is very large, and these rankings take on their lowest values when* $\beta$ *has some intermediate value.*

coupling strength $\omega \in [0.05, 5]$. Observe that the interlayer edges have the top rankings only when $\beta$ is very large. Moreover, for intermediate values of $\beta$, there exists a peak that corresponds to when the interlayer edges obtain their lowest rankings. Interestingly, these two phenomena are reminiscent of our findings for interparty edges that we presented in section 4.1. By comparing the different curves for different $\omega$, we can observe that the peak is sharper for larger $\omega$ values, whereas it is more of a plateau for smaller $\omega$ (and can even have two optima). In Appendix E, we show that qualitatively similar results occur for 24 other multiplex brain networks taken from [53].

In Figure 4(C), we support these findings by studying $\langle \tilde{R}_{pq} \rangle_{inter}$ for a random-graph model in which the two layers are given by Erdős–Rényi $G_{NM}$ networks for $N = 148$ and $M = 7052$ in which the number of edges $M$ is chosen so that the numbers of intralayer edges in each layer matches those for $\mathbf{A}^{(\delta)}$ and $\mathbf{A}^{(\theta)}$. We plot $\langle \tilde{R}_{pq} \rangle_{inter}$ versus $\beta$ for this generative model in Figure 4(C) for different choices of $\omega$, where one can observe that the behavior is qualitatively similar to that which was shown in Figure 4(B). Despite this similarity, there are some differences between the curves in panels (B) and (C), which could result because the empirical network contains rich structural features (e.g., heterogeneous weights for intralayer edges and nonrandom edges) that are lacking from the simple model that we study in panel (C).

**4.4. Comparison to other edge centralities for the case studies.** Here, we provide additional insight about our VNE-based edge centrality by comparing Algorithm 3.2 to three other edge centralities: current flow [12], betweenness [83], and an extension of node degree centrality whereby a centrality score is computed for each edge $(i, j)$ by adding the degrees of nodes $i$ and $j$. We make this comparison for three networks from the three case studies studied above.

In Figure 5(A), we plot Pearson correlation coefficients relating rankings according to Algorithm 3.2 to rankings according to each of the three other centralities. Results are shown for a U.S. Senate voting-similarity network (section 4.1). Observe for the

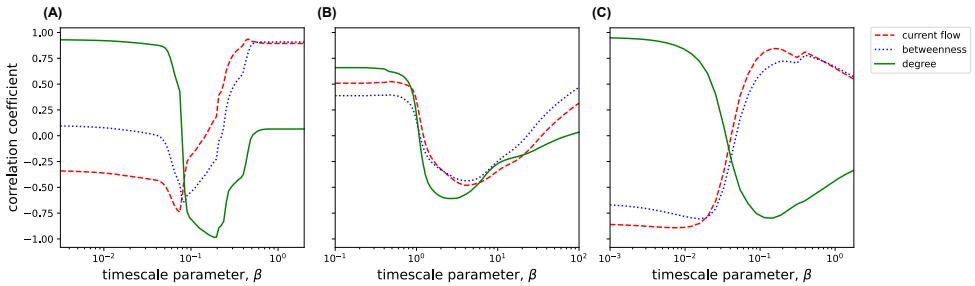


FIG. 5. Comparison of Algorithm 3.2 to other edge centralities. (A) *For the U.S. Senate voting-similarity network from section* 4.1*, we plot the Pearson correlation coefficient between rankings according to Algorithm* 3.2 *with varying* $\beta$ *and three other centralities: edge current flow (red dashed), betweenness (blue dotted), and a degree-based centrality (solid green).* (B) *Similar results but for the multimodal transportation network from section* 4.2 *under the choice of* $\chi = 0.6$. (C) *Similar results but for the multiplex brain network from section* 4.3 *under the choice of* $\omega = 1$.

smaller values of $\beta$ that rankings according to Algorithm 3.2 are strongly positively correlated with rankings under edge degree centrality but are uncorrelated (or weakly correlated) with rankings according to either edge current flow or betweenness. In contrast, for the larger values of $\beta$, rankings according to Algorithm 3.2 become uncorrelated with rankings according to edge degree centrality, but they are strongly correlated with rankings according to edge betweenness and current flow.

In Figure 5(B), we plot the same information as in panel (A), but we now consider a multimodal transport network (section 4.2). Observe that rankings according to Algorithm 3.2 are weakly correlated for most choices of $\beta$, and these can be either positively or negatively correlated depending on $\beta$. Interestingly, the correlations between Algorithm 3.2 and the other edge centralities are very similar at each value of $\beta$.

In Figure 5(C), we again plot the same information, but we now consider a multiplex brain network (section 4.3). Observe for the smaller values of $\beta$ that rankings according to Algorithm 3.2 are strongly correlated with edge degree, but they are strongly negatively correlated with rankings according to both edge current flow and betweenness. Further, for the larger values of $\beta$, rankings according to Algorithm 3.2 are weakly negatively correlated with rankings according to edge degree, but they are strongly positively correlated with rankings according to edge current flow and betweenness.

From these observations, we conclude that common notions of centrality can be similar to our proposed centrality measure that quantifies the spectral complexity of diffusion, but only for some networks and only when the diffusion timescale lies within some range of values. Importantly, our proposed VNE-based edge rankings of edges are notably different from these known rankings for other values of $\beta$, particularly, mid-range values of $\beta$. We find that the intricacies of such rankings and their correlations are generally difficult to relate to network properties, and we leave this pursuit open to future work.

**5. Discussion.** In this paper, we have proposed a centrality measure derived from von Neumann entropy (VNE) to identify and rank network edges according to their impact on the spectral complexity of information diffusion over a graph. A codebase that reproduces our findings can be found at [66]. Our work complements prior work [46] that uses VNE to obtain a node centrality measure and which does not use first-order approximation theory to be scalable for large graphs. In section 3.1,

we developed a measure for the importance of each edge $(p, q) \in \mathcal{E}$ by considering how VNE would change upon its removal. We rank the edges according to these VNE perturbations, and the resulting top-ranked edge is the one such that its removal would most increase VNE (and subsequently, most increase the spectral complexity of information diffusion at timescale $\beta$). Our approach complements existing centrality measures that are related to information spreading [35, 36, 41, 64, 67, 68, 72] but which do not consider the spectral complexity of information diffusion, as measured by VNE. Because the direct computation of VNE perturbations are computationally expensive (or infeasible) for large-scale graphs, in section 3.2 we developed an approximate ranking of edges based on first-order approximation theory that can efficiently predict these VNE perturbations. That said, we note that the computational challenges are not fully resolved for VNE-based analyses of graphs. First, VNE requires one to compute a full set of matrix eigenvalues, which has a general scaling of $\mathcal{O}(N^3)$ for graphs with $N$ nodes. In principle, this poor computational scaling could be mitigated by incorporating spectral approximation techniques such as those that rely on message passing [17], subgraph motifs [92, 93], kernel polynomial methods [30, 125], random matrix theory [90], and randomized numerical linear algebra [77].

Also, for our goal of approximating how VNE is perturbed due to structural modifications to a network, it may be practical to approximate VNE perturbations using only a subset of eigenvalue perturbations, alleviating the need to compute all eigenvalues. That said, the number of eigenvalues required to maintain some level of approximation accuracy would vary greatly depending on the diffusion timescale $\beta$ that is considered. For example, when $\beta$ is very large, one would expect that $H'(0)$ given by (3.7) could be accurately approximated using only a few of the smallest eigenvalues $\lambda_i$ (i.e., since $e^{-\beta\lambda_i}$ would be negligible for larger eigenvalues). Here, we intentionally explored a wide range of timescales $\beta$, which can also be computationally expensive. Thus, it may also be beneficial to develop methods that explore the $\beta$-parameter space more systematically by incorporating, e.g., one's external knowledge about some dynamical, structural, or preferential criteria. One possible direction would be to leverage spectral theory for block-structure matrices including those arising for multiplex networks [50, 103, 110, 112, 113].

Despite these challenges, we utilized our approximation theory for perturbed diffusion-kernel VNE to study the entropic importance of edges via three case studies with empirical networks to investigate structural/dynamical mechanisms that can systematically influence the entropic importance of edges. In section 4.1, we studied a network encoding voting similarity in the 117th U.S. Senate and studied the importance of interparty edges (i.e., those that connect across the two large-scale communities that result from party polarization). In section 4.2, we studied a multimodal transportation network and studied the importance of edges representing metro lines as opposed to those representing roads. In section 4.3, we studied a multiplex brain network and studied the importance of interlayer edges that couple network layers versus intralayer edges that connect nodes within a particular layer. In each of these case studies, we found that the edges that are deemed to be the most important drastically change by considering different diffusion timescales (i.e., different values of the parameter $\beta$).

Notably, our proposed techniques for measuring the importance of edges with respect to VNE are expected to have many applications beyond our three case studies. As an entropy-based measure, we expect our techniques to help reveal new ways to leverage information theory to study diverse types of biological, social, and physical systems. Because diffusion-kernel VNE has a well-defined interpretation for the diffusion equation [see eq. (2.2)], it can also contribute to the related literature on

Laplacian-based algorithms for data analysis and machine learning [6, 15, 21, 60, 70]. As one example, our ranking of edges could contribute to the development of Laplacian-based algorithms for graph sparsification [106] as well as link prediction [71, 117]. However, there are limiting cases for our proposed technique such as signed networks and directed networks, where the spectra can take on complex values for which VNE is not yet defined. In addition, since our choice of VNE is defined by the diffusion process, it may be inappropriate to apply to networks with disconnected components, since diffusion can be trapped onto a component depending on the initial condition.

That said, our investigation of the multiscale aspects of entropic importance in this paper (i.e., whereby the most important edges depend sensitively on the diffusion timescale $\beta$ that is considered) also opens up new lines of scientific inquiry. For instance, we observed "ranking regimes" in which the entropic importance of edges remains relatively insensitive to $\beta$ as well as "regime transitions" in which rankings drastically change as $\beta$ varies. (See [112, 113] for further discussion on multiscale centrality regimes and transitions.) Our case studies explored three structural scenarios in which such behavior can arise, and it would be beneficial to develop a deeper understanding in future research. Moreover, while we focused herein on a formulation of VNE that is related to information diffusion, our methods could be extended to other applications by considering other dynamical systems and their related matrices. For example, by considering a normalized Laplacian matrix $\tilde{\mathbf{L}} = \mathbf{D}^{-1}\mathbf{L}$, VNE-based methods would more appropriately describe information theory related to continuous-time random walks [24] as opposed to diffusion. One could also consider spectral entropies relating to adjacency matrices and nonbacktracking matrices [3]. However, one benefit from defining VNE according to the diffusion equation is that it has a closed-form, analytical solution using spectral theory, and so spectral perturbation theory can be used as it is in this paper. Extending to different dynamical processes may pose a challenge if no closed-form analytical solution exists. In such situations, methods such as trace estimators [122] might help. Another direction could involve using Hodge Laplacian matrices [98, 129] to extend VNE and VNE-related centrality measures to simplicial complexes (which are a higher-order generalization of graphs). Common to all of these scenarios, it would be beneficial to consider how structural modifications impact VNE, how these VNE perturbations can be used for centrality analysis, and how spectral perturbation theory can support the development of computationally efficient methodologies.

**Appendix A. Proof of Theorem 3.4.** Equation (3.6) establishes that the first-order derivative of VNE can be expanded in terms of partial derivatives

$$H'(0) = \sum_{i,j} \frac{\partial h}{\partial f_i} \frac{\partial f_i}{\partial \lambda_j} \frac{\partial \lambda_j}{\partial \epsilon} \bigg|_{\epsilon=0},$$

where we define VNE as $h(\mathbf{f}) = -\sum_i f_i(\vec{\lambda}) \log_2(f_i(\vec{\lambda}))$. Our proof only requires that we solve for these terms. It was shown in [76] that the first terms are given by

$$\frac{\partial h}{\partial f_i} = -\log_2\left(f_i(\vec{\lambda})\right) - \frac{1}{\ln(2)}. \tag{A.1}$$

Considering the diffusion-kernel VNE given by

$$f_i(\vec{\lambda}) = \frac{e^{-\beta\lambda_i}}{\sum_k e^{-\beta\lambda_k}},$$

we obtain the derivatives $\partial f_i/\partial \lambda_j$ separately for $j = i$ and $j \neq i$. For $j \neq i$, we find

$$\begin{aligned}\frac{\partial f_i}{\partial \lambda_j} &= \frac{\left(\sum_k e^{-\beta\lambda_k}\right)(0) - \left(e^{-\beta\lambda_i}\right)\left(-\beta e^{-\beta\lambda_j}\right)}{\left[\sum_k e^{-\beta\lambda_k}\right]^2} \\ &= 0 + \beta \frac{e^{-\beta\lambda_i}}{\sum_k e^{-\beta\lambda_k}} \frac{e^{-\beta\lambda_j}}{\sum_k e^{-\beta\lambda_k}} \\ &= \beta f_i(\vec{\lambda}) f_j(\vec{\lambda}). \end{aligned} \tag{A.2}$$

For $j = i$, we find

$$\begin{aligned}\frac{\partial f_i}{\partial \lambda_i} &= \frac{\left(\sum_k e^{-\beta\lambda_k}\right)\left(-\beta e^{-\beta\lambda_i}\right) - \left(e^{-\beta\lambda_i}\right)\left(-\beta e^{-\beta\lambda_i}\right)}{\left[\sum_k e^{-\beta\lambda_k}\right]^2} \\ &= -\beta \left[ \frac{\left(\sum_k e^{-\beta\lambda_k}\right)\left(e^{-\beta\lambda_i}\right)}{\left[\sum_k e^{-\beta\lambda_k}\right]^2} - \frac{\left(e^{-\beta\lambda_i}\right)\left(e^{-\beta\lambda_i}\right)}{\left[\sum_k e^{-\beta\lambda_k}\right]^2} \right] \\ &= -\beta f_i(\vec{\lambda}) \left(1 - f_i(\vec{\lambda})\right). \end{aligned} \tag{A.3}$$

Combining these results, we obtain

$$\begin{aligned}\sum_j \frac{\partial f_i}{\partial \lambda_j} \frac{\partial \lambda_j}{\partial \epsilon} &= -\beta f_i(\vec{\lambda}) \left[ \left(1 - f_i(\vec{\lambda})\right) \frac{\partial \lambda_i}{\partial \epsilon} - \sum_{j \neq i}^{N} f_j(\vec{\lambda}) \frac{\partial \lambda_j}{\partial \epsilon} \right] \\ &= -\beta f_i(\vec{\lambda}) \left[ \frac{\partial \lambda_i}{\partial \epsilon} - \sum_{j=1}^{N} f_j(\vec{\lambda}) \frac{\partial \lambda_j}{\partial \epsilon} \right]. \end{aligned} \tag{A.4}$$

Finally, we use that

$$\left. \frac{\partial \lambda_i}{\partial \epsilon} \right|_{\epsilon=0} = \lambda_i'(0) = (\mathbf{u}^{(i)})^T \Delta L \mathbf{u}^{(i)} \tag{A.5}$$

as given by (2.7). We combine these terms to recover (3.7).

**Appendix B. Numerical validation of Theorem 3.4.** In Figure 6, we provide numerical validation for Theorem 3.4. First, we computed the VNE $h(L)$ of a random graph that was sampled from the Erdős–Rényi $G_{NM}$ model with $N = 100$ nodes and $M = 2000$ edges. In Figure 6(A), we plot the perturbed VNE $h(L + \Delta L)$ (blue dotted curve) after removal of $k$ edges with $k \in \{0, \ldots, 20\}$. Edges were selected for removal uniformly at random. For comparison, the red dashed curve depicts our first-order approximation $h(L + \Delta L) \approx h(L) + H'(0)$ with $H'(0)$ given by (3.7) with $\Delta L$ defined according to Proposition 2.3. Note for the deletion of several edges that $\Delta L = -\sum_{(p,q)} \Delta L^{(pq)}$, where the sum is taken over the set of removed edges [115]. Observe in Figure 6(A) that, as expected, the first-order approximation is more accurate when the perturbation is smaller (i.e., smaller $k$).

In Figure 6(B), we support our claim that the approximation $h(L+\epsilon\Delta L) \approx h(L)+\epsilon H'(0)$ is first-order accurate by showing that the approximation error has a second-order scaling behavior. Specifically, we plot the approximation error $E(\epsilon) \equiv |h(L + \epsilon\Delta L) - [h(L) + \epsilon H'(0)]|$ versus $\epsilon$ and find that the error decays as $\mathcal{O}(\epsilon^2)$. Note that in a log-log scale, this scaling corresponds to a linear relationship $\log(E(\epsilon)) \propto 2\log(\epsilon)$. In Figure 6(B), we plot $\log(E(\epsilon))$ versus $\log(\epsilon)$ for three choices of $k \in \{1, 10, 20\}$, and we computed a least-squares linear fit to each curve. We empirically observed the

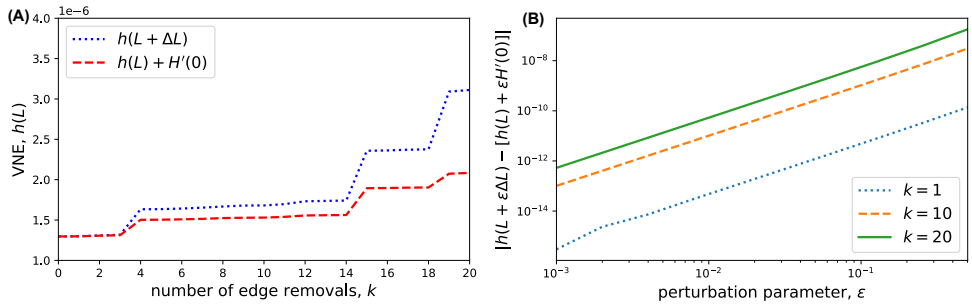


FIG. 6. Numerical validation of first-order perturbation theory for diffusion-kernel VNE. (A) *Comparison of* $h(L+\Delta L)$ *and our first-order prediction,* $h(L)+H'(0)$*, with* $H'(0)$ *given by Theorem* 3.4*, for a perturbed graph. We consider an Erdős–Rényi* $G_{NM}$ *random graph with* $N=100$ *nodes and* $M=1500$ *unweighted edges, and each perturbation matrix* $\Delta L$ *encodes the removal of* $k$ *edges as described in* [115] *(which generalized Lemma* 2.4 *to edge sets). Observe that the first-order approximation becomes less accurate for larger perturbations (i.e., larger* $k$*).* (B) *Numerical validation that the approximation error* $E(\epsilon)\equiv|h(L+\epsilon\Delta L)-[h(L)+\epsilon H'(0)]|$ *vanishes in the limit of small* $\epsilon$ *as* $\mathcal{O}(\epsilon^2)$*, that is,* $\log(E(\epsilon))\propto 2\log(\epsilon)$*. We plot* $E(\epsilon)$ *for the same network as in panel* (A) *in a log-log scale and compute a least-squares linear fit to obtain empirically measured slopes* $\{2.042, 2.022, 2.036\}$*, respectively, for three choices of* $k\in\{1,10,20\}$*. These are all nearly equal to the predicted slope of 2, supporting our claim that Theorem* 3.4 *has second-order error.*

slopes to be approximately 2.014, 2.022, and 2.026 for these three lines, respectively. Because these are all very close to our analytically predicted slope of 2, we can be confident that our first-order approximation does in fact have second-order error.

**Appendix C. Further comparison of Algorithms 3.1 and 3.2.** Here, we extend the results shown in Figure 1(B) by repeating the experiment for two empirical networks: a network encoding voting-pattern similarity among the U.S. Senate and a multiplex network derived from fMRI data. See sections 4.1 and 4.3, respectively, for their descriptions. We also highlight that we were unable to conduct this experiment for the transit network described in section 4.2, since we found it infeasible to compute Algorithm 3.1 for that network. Specifically, the transit network contains approximately 22 times more nodes than the other two empirical networks. Since the runtime of Algorithm 3.1 scales like $MN^3$ for $M$ edges and $N$ nodes, we estimate that Algorithm 3.1 for the transit network could take up to 10,000 times longer than for the other two empirical networks. (Recall that the poor scaling of Algorithm 3.1 was our main motivation for developing Algorithm 3.2.)

In Figures 7(A) and (B), we present results that are similar to those shown in Figure 1(B) but which are now obtained for two empirical networks: (A) a U.S. Senate voting network and (B) a multiplex brain network. For both networks, we plot the fractional overlap among the top-$\ell$ ranked edges for the two algorithms with $\beta=1$ and varying $\ell$. Observe for both networks that both algorithms identify a similar set of top-ranked edges. The insets show scatter plots that directly compare the algorithms rankings. Observe that the approximate edge rankings $\{\tilde{R}_{pq}\}$ and true edge rankings $\{R_{pq}\}$ are very similar for all edges in both networks. These results recapitulate our findings that were previously discussed for Figure 1(B).

**Appendix D. Further description of the voting-similarity network in section 4.1.** Here, we describe our construction of an empirical network that encodes voting similarity among persons in the 117th U.S. Senate. We first downloaded voting-pattern data from VoteView [73] to create a tensor with entries $\gamma_{ijk}=1$ when Senators

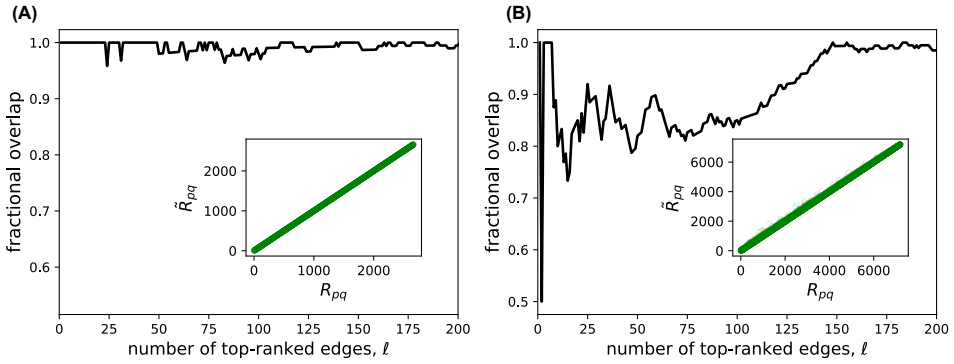


FIG. 7. Comparison of Algorithms 3.1 and 3.2 for U.S. Senate and multiplex brain networks. *Extending the results in Figure* 1(B), *we now compare the two algorithms for two empirical networks:* (A) *a U.S. Senate voting network and* (B) *a multiplex brain network with interlayer coupling* $\omega = 1$. *See sections* 4.1 *and* 4.3, *respectively, for their descriptions. In both panels, we plot the fractional overlap among the top-*$\ell$ *ranked edges for the two algorithms with* $\beta = 1$ *and varying* $\ell$. *The insets show scatter plots that directly compare the rankings from the algorithms.*

$i$ and $j$ vote identically on bill $k$ and $\gamma_{ijk} = 0$ otherwise. Letting $b_{ij}$ denote the number of bills in which both $i$ and $j$ vote, we define a weighted adjacency matrix with entries $\tilde{A}_{ij} = \frac{1}{b_{ij}} \sum_k \gamma_{ijk} \in [0,1]$ indicating for each pair of Senators the fraction of bills on which they vote identically. Equivalently, we only sum over bills $k$ in which both $i$ and $j$ cast a vote. Finally, we obtain a sparse network with an adjacency matrix $\mathbf{A}$ by applying a threshold $\tau$ to $\tilde{\mathbf{A}}$. That is, we define

$$A_{ij} = \begin{cases} \tilde{A}_{ij}, & \tilde{A}_{ij} > \tau, \\ 0 & \text{otherwise.} \end{cases} \tag{D.1}$$

After exploring a wide range of $\tau$ values, we selected $\tau = 0.40$, which is a minimum voting-together fraction of 0.40. The resulting graph is a weighted voting-similarity network [81, 82, 124] that is undirected, and each edge $(i, j)$ has a weight $A_{ij} > 0.4$.

We note that there are other ways to construct voting-similarity networks, including, e.g., by restricting attention to nonunanimous roll calls [124] as opposed to considering all bills (as we have done). Note that for convenience, we removed Kamala Harris and Kelly Loeffler from the data set because they participated in the 117th Senate for less than a month before resigning or being replaced. We also relabeled the independents to be Democrats since the current independents caucus with the Democrats and because, for simplicity, we would like to focus this experiment on a network having two communities of approximately equal size.

Recall from section 4.1 that we would like to compare the rankings $\tilde{R}_{pq}$ of edges within and between communities, i.e., intraparty and interparty edges. As such, we selected $\tau = 0.4$ so that the resulting network has few edges between the two communities. In this case, we found 474 interparty edges and 2182 intraparty edges. This occurs due to party polarization; that is, the vast majority of edges are intraparty edges that connect Senators having the same party affiliation, whereas relatively few edges are interparty edges. Since most Senators are Republicans or Democrats (and the current independents largely act as Democrats in their roll call voting), the resulting network has two large communities. As shown in Figure 8(A), these communities manifest as "blocks" in the adjacency matrix $\mathbf{A}$. In Figure 8(B), we depict the

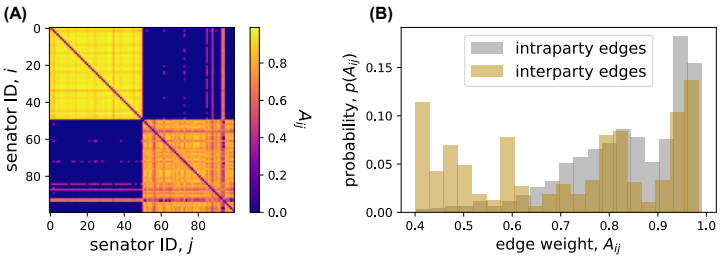


FIG. 8. Network encoding voting similarity among the 117th U.S. Senate. (A) *Visualization of the adjacency matrix* $\mathbf{A}$*, where* $A_{ij}$ *encodes the fraction of bills in which Senators* $i$ *and* $j$ *vote identically (ignoring fractions less than* 0.4*). By sorting node IDs by the Senator's political affiliations so that nodes* $\{1,\ldots,50\}$ *are Republicans and nodes* $\{51,\ldots,100\}$ *are either Democrats or independents, the resulting two-community structure manifests as two large "blocks" in the matrix.* (B) *Histogram depicts the probability distributions for edge weights* $\{A_{ij}\}$*, which we measure separately across intraparty and interparty edges.*

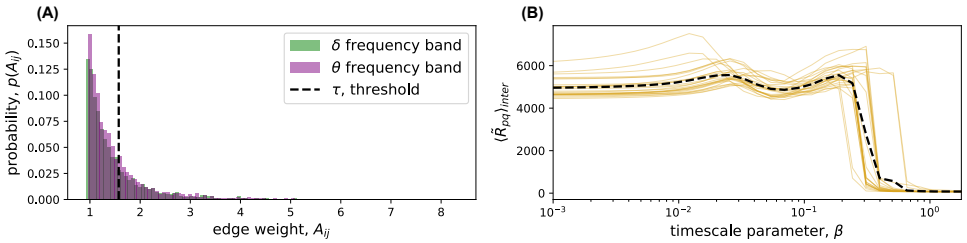


FIG. 9. Multiplex network encoding spectral coherence among brain regions. (A) *Histograms depicting the probability distributions of edge weights* $\{A_{ij}^{(\delta)}\}$ *and* $\{A_{ij}^{(\theta)}\}$ *for two network layers encoding spectral-coherence relations among the fMRI activity brain regions at two frequency bands.* (B) *We plot the mean ranking* $\langle \tilde{R}_{pq} \rangle_{inter}$ *of interlayer edges versus timescale parameter* $\beta$ *for the coupling strength* $\omega = 1$*. Different curves represent the* 25 *healthy human brains in the data set* [52, 53]*. The black dashed line depicts the mean curve.*

distributions of edge weights $\{A_{ij}\}$, which we measure separately across intraparty and interparty edges.

**Appendix E. Further description of the multiplex brain network in section 4.3.** Here, we provide additional discussion for our experiments with multiplex brain networks. We first downloaded adjacency matrices that encode network layers from [52]. See [53] for a paper discussing this data and related experiments. Each adjacency matrix encodes spectral coherence for a distinct particular frequency band. Diagonal entries in the matrices are set to zero to ensure that there are no self-edges. For each network layer, nodes represent brain regions and weighted edges represent coherence values. The full data includes six network layers for each person, and there are data for 25 healthy persons and 25 persons with Alzheimer's disease. We focused on two network layers that represent the frequency bands of $\delta$ and $\theta$ waves by examining two adjacency matrices $\mathbf{A}^{(\delta)}$ and $\mathbf{A}^{(\theta)}$ for the first person in the data set.

In Figure 9(A), we plot histograms of the coherence values $A_{ij}^{(\delta)}$ and $A_{ij}^{(\theta)}$ encoded in these two matrices. Because we would like to study sparse networks, we choose a threshold of $\tau = 1.52$ and remove edges having a weight less than $\tau$. Observe that the two distributions $p(A_{ij}^{(\delta)})$ and $p(A_{ij}^{(\theta)})$ are similar, and so the two "sparsified" network layers have similar weight distributions and a similar number of edges.

In Figure 9(B), we present a plot that is similar to Figure 4(B). Here, we plot the mean ranking $\langle \tilde{R}_{pq} \rangle_{inter}$ of interlayer edges versus timescale parameter $\beta$ for the

coupling strength $\omega = 1$. The different curves represent the 25 different healthy human brains in the study, and the dashed black curve gives the mean curve. Observe that all curves are qualitatively similar, suggesting that the effect of $\beta$ on edges' entropic importance is robust across the brain data.

## REFERENCES


[1] E. Abbe, *Community detection and stochastic block models: Recent developments*, J. Mach. Learn. Res., 18 (2017), pp. 6446–6531.

[2] S. Allesina and M. Pascual, *Googling food webs: Can an eigenvector measure species' importance for coextinctions?*, PLoS Comput. Biol., 5 (2009), e1000494.

[3] N. Alon, I. Benjamini, E. Lubetzky, and S. Sodin, *Non-backtracking random walks mix faster*, Commun. Contemp. Math., 9 (2007), pp. 585–603.

[4] K. Anand, G. Bianconi, and S. Severini, *Shannon and von Neumann entropy of random networks with heterogeneous expected degree*, Phys. Rev. E, 83 (2011), 036109.

[5] K. E. Atkinson, *An Introduction to Numerical Analysis*, John Wiley & Sons, 2008.

[6] M. Belkin and P. Niyogi, *Laplacian eigenmaps and spectral techniques for embedding and clustering*, in Advances in Neural Information Processing Systems 14, MIT Press, 2001, pp. 585–591.

[7] B. Benigni, A. Ghavasieh, A. Corso, V. d'Andrea, and M. De Domenico, *Persistence of information flow: A multiscale characterization of human brain*, Netw. Neurosci., 5 (2021), pp. 831–850.

[8] A. R. Benson, *Three hypergraph eigenvector centralities*, SIAM J. Math. Data Sci., 1 (2019), pp. 293–312, https://doi.org/10.1137/18M1203031.

[9] G. Bertagnolli, C. Agostinelli, and M. De Domenico, *Network depth: Identifying median and contours in complex networks*, J. Complex Netw., 8 (2020), cnz041.

[10] S. P. Borgatti, *Centrality and network flow*, Soc. Networks, 27 (2005), pp. 55–71.

[11] U. Brandes, *A faster algorithm for betweenness centrality*, J. Math. Sociol., 25 (2001), pp. 163–177.

[12] U. Brandes and D. Fleischer, *Centrality measures based on current flow*, in Annual Symposium on Theoretical Aspects of Computer Science, Springer, 2005, pp. 533–544.

[13] S. L. Braunstein, S. Ghosh, and S. Severini, *The Laplacian of a graph as a density matrix: A basic combinatorial approach to separability of mixed states*, Ann. Comb., 10 (2006), pp. 291–317.

[14] S. Brin and L. Page, *The anatomy of a large-scale hypertextual web search engine*, Comput. Networks ISDN Syst., 30 (1998), pp. 107–117.

[15] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, *Geometric deep learning: Going beyond Euclidean data*, IEEE Signal Process. Mag., 34 (2017), pp. 18–42.

[16] T. Callaghan, P. J. Mucha, and M. A. Porter, *Random walker ranking for NCAA division* I-A *football*, Amer. Math. Monthly, 114 (2007), pp. 761–777.

[17] G. T. Cantwell and M. E. J. Newman, *Message passing on networks with loops*, Proc. Natl. Acad. Sci. USA, 116 (2019), pp. 23398–23403, https://doi.org/10.1073/pnas.1914893116.

[18] V. Chellappan and K. M. Sivalingam, *Application of entropy of centrality measures to routing in tactical wireless networks*, in 2013 19th IEEE Workshop on Local & Metropolitan Area Networks (LANMAN), IEEE, 2013, pp. 1–6.

[19] A. Clauset, S. Arbesman, and D. B. Larremore, *Systematic inequality and hierarchy in faculty hiring networks*, Sci. Adv., 1 (2015), https://www.science.org/doi/10.1126/sciadv.1400005.

[20] P. Clough and M. Sanderson, *Evaluating the performance of information retrieval systems using test collections*, Inform. Res., 18 (2013), pp. 18–12.

[21] R. R. Coifman and S. Lafon, *Diffusion maps*, Appl. Comput. Harmon. Anal., 21 (2006), pp. 5–30.

[22] R. R. Coifman and M. Maggioni, *Diffusion wavelets*, Appl. Comput. Harmon. Anal., 21 (2006), pp. 53–94.

[23] C. De Bacco, D. B. Larremore, and C. Moore, *A physical model for efficient ranking in networks*, Sci. Adv., 4 (2018), eaar8260.

[24] M. De Domenico and J. Biamonte, *Spectral entropies as information-theoretic tools for complex network comparison*, Phys. Rev. X, 6 (2016), 041062.

[25] M. De Domenico, V. Nicosia, A. Arenas, and V. Latora, *Structural reducibility of multilayer networks*, Nat. Commun., 6 (2015), 6864.

[26] M. De Domenico, A. Solé-Ribalta, E. Omodei, S. Gómez, and A. Arenas, *Ranking in interconnected multilayer networks reveals versatile nodes*, Nat. Commun., 6 (2015), 6868.
[27] D. R. DeFord and S. D. Pauls, *A new framework for dynamical models on multiplex networks*, J. Complex Netw., 6 (2018), pp. 353–381.
[28] J. W. Demmel, O. A. Marques, B. N. Parlett, and C. Vömel, *Performance and accuracy of LAPACK's symmetric tridiagonal eigensolvers*, SIAM J. Sci. Comput., 30 (2008), pp. 1508–1526, https://doi.org/10.1137/070688778.
[29] Y. Ding, E. Yan, A. Frazho, and J. Caverlee, *PageRank for ranking authors in co-citation networks*, J. Amer. Soc. Inform. Sci. Technol., 60 (2009), pp. 2229–2243.
[30] K. Dong, A. R. Benson, and D. Bindel, *Network density of states*, in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019, pp. 1152–1161.
[31] P. G. Doyle and J. L. Snell, *Random Walks and Electric Networks*, Carus Math. Monogr. 22, Mathematical Association of America, Washington, DC, 1984.
[32] Y. Du, J. Leung, and Y. Shi, *PerturbationRank: A Non-monotone Ranking Algorithm*, Technical report, University of Michigan, Ann Arbor, MI, 2008.
[33] P. Erdös and A. Rényi, *On random graphs,* I, Publ. Math. Debrecen 6 (1959), pp. 290–297.
[34] E. Estrada, N. Hatano, and M. Benzi, *The physics of communicability in complex networks*, Phys. Rep., 514 (2012), pp. 89–119.
[35] E. Estrada and D. J. Higham, *Network properties revealed through matrix functions*, SIAM Rev., 52 (2010), pp. 696–714, https://doi.org/10.1137/090761070.
[36] E. Estrada, D. J. Higham, and N. Hatano, *Communicability betweenness in complex networks*, Phys. A Statist. Mech. Appl., 388 (2009), pp. 764–774.
[37] E. Estrada and J. A. Rodriguez-Velazquez, *Subgraph centrality in complex networks*, Phys. Rev. E, 71 (2005), 056103.
[38] E. Estrada and G. J. Ross, *Centralities in simplicial complexes. applications to protein interaction networks*, J. Theor. Biol., 438 (2018), pp. 46–60.
[39] M. Fiedler, *Laplacian of graphs and algebraic connectivity*, in Combinatorics and Graph Theory (Warsaw, 1987), Banach Center Publ. 25, PWN—Polish Scientific Publishers, Warsaw, 1989, pp. 57–70.
[40] D. Freedman, *Markov Chains*, Springer Science & Business Media, 2012.
[41] L. C. Freeman, *A set of measures of centrality based on betweenness*, Sociometry, 40 (1977), pp. 35–41.
[42] A. Ghavasieh, S. Bontorin, O. Artime, N. Verstraete, and M. De Domenico, *Multiscale statistical physics of the pan-viral interactome unravels the systemic nature of SARS-CoV-2 infections*, Commun. Phys., 4 (2021), pp. 1–13.
[43] A. Ghavasieh and M. De Domenico, *Enhancing transport properties in interconnected systems without altering their structure*, Phys. Rev. Res., 2 (2020), 013155.
[44] A. Ghavasieh and M. De Domenico, *Statistical physics of network structure and information dynamics*, J. Phys. Complexity, 3 (2022), 011001.
[45] A. Ghavasieh, C. Nicolini, and M. De Domenico, *Statistical physics of complex information dynamics*, Phys. Rev. E, 102 (2020), 052304.
[46] A. Ghavasieh, M. Stella, J. Biamonte, and M. De Domenico, *Unraveling the effects of multiscale network entanglement on empirical systems*, Commun. Phys., 4 (2021), pp. 1–10.
[47] M. Girvan and M. E. J. Newman, *Community structure in social and biological networks*, Proc. Natl. Acad. Sci. USA, 99 (2002), pp. 7821–7826.
[48] D. F. Gleich, *PageRank Beyond the Web*, SIAM Rev., 57 (2015), pp. 321–363, https://doi.org/10.1137/140976649.
[49] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 2013.
[50] S. Gomez, A. Diaz-Guilera, J. Gomez-Gardenes, C. J. Perez-Vicente, Y. Moreno, and A. Arenas, *Diffusion dynamics on multiplex networks*, Phys. Rev. Lett., 110 (2013), 028701.
[51] F. Grando, L. Z. Granville, and L. C. Lamb, *Machine learning in network centrality measures: Tutorial and outlook*, ACM Comput. Surv. (CSUR), 51 (2018), 102.
[52] J. Guillon, *Codebase: Brain Network Toolbox*, https://github.com/brain-network/bnt, 2016.
[53] J. Guillon, Y. Attal, O. Colliot, V. La Corte, B. Dubois, D. Schwartz, M. Chavez, and F. De Vico Fallani, *Loss of brain inter-frequency hubs in Alzheimer's disease*, Sci. Rep., 7 (2017), 10879.
[54] R. Guimerà, S. Mossa, A. Turtschi, and L. A. N. Amaral, *The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles*, Proc. Natl. Acad. Sci. USA, 102 (2005), pp. 7794–7799.

[55] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. Del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, *Array programming with NumPy*, Nature, 585 (2020), pp. 357–362, https://doi.org/10.1038/s41586-020-2649-2.

[56] P. Hines and S. Blumsack, *A centrality measure for electrical networks*, in Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008), IEEE, 2008, pp. 185–185.

[57] F. Holik, A. Plastino, and M. Sáenz, *Natural information measures for Cox' approach for contextual probabilistic theories*, Quant. Inf. Comput., 16 (2016), pp. 0115–0133, https://doi.org/10.26421/QIC16.1-2-8.

[58] P. Holme, *Congestion and centrality in traffic flow on complex networks*, Adv. Complex Syst., 6 (2003), pp. 163–176.

[59] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai, *Lethality and centrality in protein networks*, Nature, 411 (2001), pp. 41–42.

[60] L. G. S. Jeub, P. Balachandran, M. A. Porter, P. J. Mucha, and M. W. Mahoney, *Think locally, act locally: Detection of small, medium-sized, and large communities in large networks*, Phys. Rev. E, 91 (2015), 012821.

[61] F. Jordán, Z. Benedek, and J. Podani, *Quantifying positional importance in food webs: A comparison of centrality indices*, Ecol. Model., 205 (2007), pp. 270–275.

[62] C. E. Kampmann, *Feedback loop gains and system behavior*, in Proceedings of the 1996 International System Dynamics Conference, System Dynamics Society, Cambridge, MA, 1996, pp. 21–25.

[63] C. E. Kampmann and R. Oliva, *Loop eigenvalue elasticity analysis: Three case studies*, Syst. Dyn. Rev., 22 (2006), pp. 141–162.

[64] L. Katz, *A new status index derived from sociometric analysis*, Psychometrika, 18 (1953), pp. 39–43.

[65] J. Kazimer, *Voteview Python: A Python Module for Data Extraction and Network Analysis*, 2021, https://github.com/k-z-m-r/voteview-python.

[66] J. Kazimer, *Ranking Edges by their Impact on the Spectral Complexity of Information Diffusion Over Networks*, 2022, https://github.com/k-z-m-r/vne-based-edge-rankings.

[67] D. Kempe, J. Kleinberg, and É. Tardos, *Maximizing the spread of influence through a social network*, in Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003, pp. 137–146.

[68] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, *Identification of influential spreaders in complex networks*, Nat. Phys., 6 (2010), pp. 888–893.

[69] J. M. Kleinberg, *Authoritative sources in a hyperlinked environment*, in The Structure and Dynamics of Networks, M. Newman, A.-L. Barabási, and D. J. Watts, eds., Princeton University Press, Princeton, NJ, 2006, pp. 514–542, https://doi.org/10.1515/9781400841356.514.

[70] K. Kloster and D. F. Gleich, *Heat kernel based community detection*, in Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014, pp. 1386–1395.

[71] J. Kunegis and A. Lommatzsch, *Learning spectral graph transformations for link prediction*, in Proceedings of the 26th Annual International Conference on Machine Learning, ACM, 2009, pp. 561–568.

[72] S. Lehmann and Y.-Y. Ahn, *Complex Spreading Phenomena in Social Systems*, Springer, 2018.

[73] J. B. Lewis, K. Poole, H. Rosenthal, A. Boche, A. Rudkin, and L. Sonnet, *Voteview: Congressional Roll-Call Votes Database*, 2018, https://www.voteview.com/ (accessed 27 July 2018).

[74] H. Li and Z. Zhang, *Kirchhoff index as a measure of edge centrality in weighted networks: Nearly linear time algorithms*, in SODA '18: Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, 2018, pp. 2377–2396.

[75] Y. Li, W. Cai, Y. Li, and X. Du, *Key node ranking in complex networks: A novel entropy and mutual information-based approach*, Entropy, 22 (2020), 52, https://doi.org/10.3390/e22010052.

[76] Z. Li, P. J. Mucha, and D. Taylor, *Network-ensemble comparisons with stochastic rewiring and von Neumann entropy*, SIAM J. Appl. Math., 78 (2018), pp. 897–920, https://doi.org/10.1137/17M1124218.

[77] P.-G. Martinsson and J. A. Tropp, *Randomized numerical linear algebra: Foundations and algorithms*, Acta Numer., 29 (2020), pp. 403–572.

[78] S. B. Maurer, *Matrix generalizations of some theorems on trees, cycles and cocycles in graphs*, SIAM J. Appl. Math., 30 (1976), pp. 143–148, https://doi.org/10.1137/0130017.
[79] T. Mendes-Santos, G. Giudici, R. Fazio, and M. Dalmonte, *Measuring von Neumann entanglement entropies without wave functions*, New J. Phys., 22 (2020), 013044, https://doi.org/10.1088/1367-2630/ab6875.
[80] A. Milanese, J. Sun, and T. Nishikawa, *Approximating spectral impact of structural perturbations in large networks*, Phys. Rev. E, 81 (2010), 046112.
[81] J. Moody and P. J. Mucha, *Portrait of political party polarization*, Network Sci., 1 (2013), pp. 119–121, https://doi.org/10.1017/nws.2012.3.
[82] P. J. Mucha and M. A. Porter, *Communities in multislice voting networks*, Chaos, 20 (2010), 041108.
[83] M. E. J. Newman, *Networks: An Introduction*, Oxford University Press, Oxford, 2010.
[84] C. Nicolini, G. Forcellini, L. Minati, and A. Bifone, *Scale-resolved analysis of brain functional connectivity networks with spectral entropy*, NeuroImage, 211 (2020), 116603.
[85] A. G. Nikolaev, R. Razib, and A. Kucheriya, *On efficient use of entropy centrality for social network analysis and community detection*, Soc. Networks, 40 (2015), pp. 154–162, https://doi.org/10.1016/j.socnet.2014.10.002.
[86] R. K. Pan and J. Saramäki, *Path lengths, correlations, and centrality in temporal networks*, Phys. Rev. E, 84 (2011), 016105.
[87] J. Park and M. E. J. Newman, *A network-based ranking system for US college football*, J. Stat. Mech. Theory Exp., 2005 (2005), P10014.
[88] F. Passerini and S. Severini, *The von Neumann Entropy of Networks*, University Library of Munich, Germany, MPRA Paper, 2008, https://doi.org/10.2139/ssrn.1382662.
[89] L. M. Pecora and T. L. Carroll, *Master stability functions for synchronized coupled systems*, Phys. Rev. Lett., 80 (1998), 2109.
[90] T. P. Peixoto, *Eigenvalue spectra of modular networks*, Phys. Rev. Lett., 111 (2013), 098701.
[91] T. P. Peixoto, *Nonparametric Bayesian inference of the microcanonical stochastic block model*, Phys. Rev. E, 95 (2017), 012317.
[92] V. M. Preciado and A. Jadbabaie, *From local measurements to network spectral properties: Beyond degree distributions*, in 49th IEEE Conference on Decision and Control (CDC), IEEE, 2010, pp. 2686–2691.
[93] V. M. Preciado and A. Jadbabaie, *Moment-based spectral analysis of large-scale networks using local structural information*, IEEE/ACM Trans. Netw., 21 (2012), pp. 373–382.
[94] T. Qiao, W. Shan, G. Yu, and C. Liu, *A novel entropy-based centrality approach for identifying vital nodes in weighted networks*, Entropy, 20 (2018), 261, https://doi.org/10.3390/e20040261.
[95] J. G. Restrepo, E. Ott, and B. R. Hunt, *Characterizing the dynamical importance of network nodes and links*, Phys. Rev. Lett., 97 (2006), 094102.
[96] J. Rushmore, D. Caillaud, R. J. Hall, R. M. Stumpf, L. A. Meyers, and S. Altizer, *Network-based vaccination improves prospects for disease control in wild chimpanzees*, J. R. Soc. Interface, 11 (2014), 20140349.
[97] S. Saavedra, S. Powers, T. McCotter, M. A. Porter, and P. J. Mucha, *Mutually-antagonistic interactions in baseball networks*, Phys. A, 389 (2010), pp. 1131–1141.
[98] M. T. Schaub, A. R. Benson, P. Horn, G. Lippner, and A. Jadbabaie, *Random walks on simplicial complexes and the normalized Hodge* 1*-Laplacian*, SIAM Rev., 62 (2020), pp. 353–391, https://doi.org/10.1137/18M1201019.
[99] B. M. Schmidt and M. M. Chingos, *Ranking doctoral programs by placement: A new method*, PS Political Science & Politics, 40 (2007), pp. 523–529.
[100] L. B. Shaw and I. B. Schwartz, *Enhanced vaccine control of epidemics in adaptive networks*, Phys. Rev. E, 81 (2010), 046120.
[101] S. M. Shekatkar and S. Barve, *Importance of initial conditions in the polarization of complex networks*, Europhys. Lett., 122 (2018), 38002.
[102] P. S. Skardal, D. Taylor, and J. Sun, *Optimal synchronization of complex networks*, Phys. Rev. Lett., 113 (2014), 144101.
[103] A. Sole-Ribalta, M. De Domenico, N. E. Kouvaris, A. Diaz-Guilera, S. Gomez, and A. Arenas, *Spectral properties of the laplacian of multiplex networks*, Phys. Rev. E, 88 (2013), 032807.
[104] A. Solé-Ribalta, M. De Domenico, S. Gómez, and A. Arenas, *Random walk centrality in interconnected multilayer networks*, Phys. D, 323 (2016), pp. 73–79.
[105] Z. Song and D. Taylor, *Coupling asymmetry optimizes collective dynamics over multiplex networks*, IEEE Trans. Network Sci. Eng., 11 (2023), pp. 1524–1541.
[106] D. A. Spielman and N. Srivastava, *Graph sparsification by effective resistances*, SIAM J. Comput., 40 (2011), pp. 1913–1926, https://doi.org/10.1137/080734029.

[107] E. Strano, S. Shai, S. Dobson, and M. Barthelemy, *Multiplex networks in metropolitan areas: Generic features and local effects*, J. R. Soc. Interface, 12 (2015), 20150651.
[108] F. W. Takes and E. M. Heemskerk, *Centrality in the global network of corporate control*, Soc. Netw. Anal. Min., 6 (2016), 97.
[109] D. Taylor, *Data Release: London Transit Network*, 2015, https://sites.google.com/site/danetaylorresearch/data.
[110] D. Taylor, *Multiplex Markov chains: Convection cycles and optimality*, Phys. Rev. Res., 2 (2020), 033164.
[111] D. Taylor, F. Klimm, H. A. Harrington, M. Kramár, K. Mischaikow, M. A. Porter, and P. J. Mucha, *Topological data analysis of contagion maps for examining spreading processes on networks*, Nat. Commun., 6 (2015), 7723, https://doi.org/10.1038/ncomms8723.
[112] D. Taylor, S. A. Myers, A. Clauset, M. A. Porter, and P. J. Mucha, *Eigenvector-based centrality measures for temporal networks*, Multiscale Model. Simul., 15 (2017), pp. 537–574, https://doi.org/10.1137/16M1066142.
[113] D. Taylor, M. A. Porter, and P. J. Mucha, *Tunable eigenvector-based centralities for multiplex and temporal networks*, Multiscale Model. Simul., 19 (2021), pp. 113–147, https://doi.org/10.1137/19M1262632.
[114] D. Taylor and J. G. Restrepo, *Network connectivity during mergers and growth: Optimizing the addition of a module*, Phys. Rev. E, 83 (2011), 066112.
[115] D. Taylor, P. S. Skardal, and J. Sun, *Synchronization of heterogeneous oscillators under network modifications: Perturbation and optimization of the synchrony alignment function*, SIAM J. Appl. Math., 76 (2016), pp. 1984–2008, https://doi.org/10.1137/16M1075181.
[116] H. Tong, B. A. Prakash, T. Eliassi-Rad, M. Faloutsos, and C. Faloutsos, *Gelling, and melting, large graphs by edge manipulation*, in Proceedings of the 21st ACM International Conference on Information and Knowledge Management, 2012, pp. 245–254.
[117] L. Torres, K. S. Chan, and T. Eliassi-Rad, *Glee: Geometric Laplacian eigenmap embedding*, J. Complex Netw., 8 (2020), cnaa007.
[118] F. Tudisco, F. Arrigo, and A. Gautier, *Node and layer eigenvector centralities for multiplex networks*, SIAM J. Appl. Math., 78 (2018), pp. 853–876, https://doi.org/10.1137/17M1137668.
[119] F. Tudisco and D. J. Higham, *Node and edge nonlinear eigenvector centrality for hypergraphs*, Commun. Phys., 4 (2021), 201.
[120] F. Tutzauer, *Entropy as a measure of centrality in networks characterized by path-transfer flow*, Soc. Networks, 29 (2007), pp. 249–265, https://doi.org/10.1016/j.socnet.2006.10.001.
[121] F. Tutzauer and B. S. Elbirt, *Entropy-based centralization and its sampling distribution in directed communication networks*, Commun. Monogr., 76 (2009), pp. 351–375.
[122] S. Ubaru, J. Chen, and Y. Saad, *Fast estimation of* $\mathtt{tr}(f(A))$ *via stochastic Lanczos quadrature*, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 1075–1099, https://doi.org/10.1137/16M1104974.
[123] J. von Neumann, *Mathematische Grundlagen der Quantenmechanik*, Die Grundlehren der mathematischen Wissenschaften 38, Springer-Verlag, 2013.
[124] A. S. Waugh, L. Pei, J. H. Fowler, P. J. Mucha, and M. A. Porter, *Party Polarization in Congress: A Network Science Approach*, preprint, arXiv:0907.3509, 2009.
[125] A. Weisse, G. Wellein, A. Alvermann, and H. Fehske, *The kernel polynomial method*, Rev. Mod. Phys., 78 (2006), pp. 275–306.
[126] M. M. Wilde, *Quantum Information Theory*, Cambridge University Press, 2013.
[127] W.-L. You, A. M. Oleś, and P. Horsch, *Von Neumann entropy spectra and entangled excitations in spin-orbital models*, Phys. Rev. B, 86 (2012), 094412, https://doi.org/10.1103/physrevb.86.094412.
[128] H. Zhu, K. Sun, and P. Koniusz, *Contrastive Laplacian eigenmaps*, in Advances in Neural Information Processing Systems 34, Curran Associates, 2021, pp. 5682–5695.
[129] C. Ziegler, P. S. Skardal, H. Dutta, and D. Taylor, *Balanced Hodge Laplacians optimize consensus dynamics over simplicial complexes*, Chaos, 32 (2022), 023128, https://doi.org/10.1063/5.0080370.