

Approximating spectral impact of structural perturbations in large networks

Attilio Milanese,^{1,*} Jie Sun,^{2,†} and Takashi Nishikawa^{2,‡}¹Department of Mechanical & Aeronautical Engineering, Clarkson University, Potsdam, New York 13699-5725, USA²Department of Mathematics & Computer Science, Clarkson University, Potsdam, New York 13699-5815, USA

(Received 9 January 2010; published 23 April 2010)

Determining the effect of structural perturbations on the eigenvalue spectra of networks is an important problem because the spectra characterize not only their topological structures, but also their dynamical behavior, such as synchronization and cascading processes on networks. Here we develop a theory for estimating the change of the largest eigenvalue of the adjacency matrix or the extreme eigenvalues of the graph Laplacian when small but *arbitrary* set of links are added or removed from the network. We demonstrate the effectiveness of our approximation schemes using both real and artificial networks, showing in particular that we can accurately obtain the spectral ranking of small subgraphs. We also propose a local iterative scheme which computes the relative ranking of a subgraph using only the connectivity information of its neighbors within a few links. Our results may not only contribute to our theoretical understanding of dynamical processes on networks, but also lead to practical applications in ranking subgraphs of real complex networks.

DOI: [10.1103/PhysRevE.81.046112](https://doi.org/10.1103/PhysRevE.81.046112)

PACS number(s): 89.75.Hc, 02.10.Ox, 89.75.Fb, 05.10.—a

I. INTRODUCTION

The theory and application of complex networks have been a popular and exciting research topic since the seminal work [1,2] appeared at the end of last century. The study of complex networks interests a variety of fields, such as mathematics, physics, computer science, sociology, and biology, to list a few. (For excellent reviews, see for example Refs. [3–5].) One of the most important problems in complex networks, both from a theoretical and applicative viewpoint, is that of measuring centrality. Various measures of centrality have been proposed based on different quantitative properties of the underlying network. Examples are degree centrality, shortest path, and random walk betweenness, clustering coefficient and eigenvector component [4,6]. Common applications of such measures is the ranking of vertices of the network, in particular in the context of web search engines [7,8]. Many of these centrality measures are connected with the spectral radius of the adjacency matrix of the graph; furthermore, the spectral radius itself is crucial for a class of dynamical processes on networks [5,9–11].

A fundamental question in the study of networks is then how the spectral radius and other invariant network statistics change under structural perturbations, such as the removal or addition of a few links, or the modification of their strength. This question is particularly relevant in the context of evolving networks, since the stepwise changes in such networks are typically small, and developing efficient algorithms to track the corresponding changes in network statistics is crucial in understanding the dynamics of network evolution [12]. The effect of structural perturbations on the spectral

radius as well as other network statistics has significant consequences for the security and robustness of networked systems under component failures and intentional attacks.

Let $G=(V,E,Q)$ be a weighted graph, where V is the set of vertices, E the set of directed edges and Q the set of weights on the edges. For convenience, we label the vertices in V with integers $1, \dots, n$. Let $A \in \mathbb{R}^{n \times n}$ be the adjacency matrix of the graph, where its entry a_{ij} is defined as the weight on the edge going from node i to node j . If A is non-negative and irreducible, then the Perron-Frobenius theorem [9,13] can be used to show that its largest eigenvalue λ is nondegenerate and positive, and there exist positive left and right eigenvectors associated with it. In this case, denote the normalized left and right eigenvectors by u and v , respectively, ($u, v \in \mathbb{R}^n$), so that

$$u^T A = \lambda u^T, \quad A v = \lambda v, \quad \|u\|_2 = \|v\|_2 = 1. \quad (1)$$

The condition of non-negativity is equivalent to requiring non-negative weights on edges; on the other hand, the condition of irreducibility corresponds to the graph being strongly connected. In this paper, the graphs under consideration all share these properties.

In Ref. [11] the *dynamical importance* I_{ij} of an edge from i to j was defined as the amount of relative decrease the removal of this edge causes on λ , i.e., if after removing such an edge λ becomes $\lambda - \Delta\lambda_{ij}$, then $I_{ij} \equiv \Delta\lambda_{ij}/\lambda$. Hence the dynamical importance of an edge quantitatively captures the effect that the removal of such an edge has on the largest eigenvalue of the graph adjacency matrix. An analogous definition was introduced for the dynamical importance of vertices, where I_k is the importance of node k . Approximations of these dynamical importances, based on perturbation techniques, are given in Ref. [11]; in particular, I_{ij} and I_k can be approximated, respectively, by

$$\bar{I}_{ij} = \frac{a_{ij} u_i v_j}{\lambda u^T v} \quad (2)$$

and

*Present address: Technology Department-MS-C-MDA, CERN CH-1211, Geneva (Switzerland); amilanes@cern.ch

†Present address: Department of Physics & Astronomy, Northwestern University, Evanston, IL 60208-3112; sunj@northwestern.edu

‡tnishika@clarkson.edu

$$\bar{I}_k = \frac{u_k v_k}{u^T v}, \quad (3)$$

where u_i and v_i denote the i th component of the vectors u and v , respectively.

In this paper, the notion of dynamical importance is extended to measure the *spectral impact* [14] of a structural perturbation in which an *arbitrary* set of links are removed or added, or their weights are modified. We define the spectral impact (SI) as

$$I_B \equiv \frac{\lambda|_{A+B} - \lambda|_A}{\lambda|_A}, \quad (4)$$

where A denotes the adjacency matrix of the original graph, $\lambda|_A$ is the corresponding spectral radius, while B is the adjacency matrix of the graph consisting of all the n nodes and the links that are removed, added, or modified. Here a positive value of I_B corresponds to an increase in λ , and a negative value to a decrease. Note that in the case of removing a subgraph of A , the entries of B will be nonpositive. In this framework, the edge and vertex importance can be treated as special cases. To enable efficient estimations of the spectral impact, formulas based on the first and improved first-order approximations are presented and discussed in detail. These formulas lead to the observation that the degree centrality, as a local measure, can be viewed as a first-order approximation to the eigenvector centrality.

The rest of the paper is organized as follows. Section II reports first- and second-order perturbation results to approximate the SI using global information on the graph; in particular, the special cases of changing the weight on an edge and removing a node are addressed. In Sec. III, applications to synthetic and real-world graphs are presented. Section IV deals with estimating the SI using local information on the network. While the majority of the results concerns the spectral radius of the adjacency matrix of the graph, Sec. V deals with an extension of our method to the graph Laplacian.

II. APPROXIMATING SPECTRAL IMPACT USING GLOBAL INFORMATION

A. Perturbation results

Suppose a graph is modified so that its adjacency matrix A becomes $A + \epsilon C$. Let λ be the largest eigenvalue of A , and $\lambda + \Delta\lambda$ the largest eigenvalue of $A + \epsilon C$. The change $\Delta\lambda$ can be estimated, when $\epsilon \ll 1$, using a first-order perturbation result [13,15–17], as

$$\Delta\lambda \approx \Delta_1 = \epsilon \lambda'|_A = \frac{u^T \epsilon C v}{u^T v}, \quad (5)$$

where $\lambda'|_A$ denotes the derivative of $\lambda + \Delta\lambda$ as a function of ϵ , evaluated at $\epsilon=0$. Equation (5) shows that—at the first order—the change in the largest eigenvalue is obtained from the derivative $\lambda'|_A$, which depends on both A and C but can be computed using the vectors u and v , and the matrix C . If A is symmetric, $u=v$ and Eq. (5) is of the Rayleigh quotient form [17].

For more accurate approximation, we can use a second-order perturbation result,

$$\Delta\lambda \approx \Delta_1 + \Delta_2 = \epsilon \lambda'|_A + \frac{\epsilon^2}{2} \lambda''|_A. \quad (6)$$

Computing the second derivative term using known perturbation results would require knowledge of all the eigenvalues and eigenvectors of A [15]. This is impractical. Hence, a further approximation is introduced, as

$$\lambda''|_A \approx \frac{\lambda'|_{A+\epsilon C} - \lambda'|_A}{\epsilon}. \quad (7)$$

The term $\lambda'|_{A+\epsilon C}$ takes into account the change in u and v , to $u + \Delta u$ and $v + \Delta v$, respectively. It is proposed here to estimate $v + \Delta v$ by means of one iteration of the power method [18], starting from the unperturbed eigenvector v , as

$$\begin{aligned} v + \Delta v &\approx \frac{(A + \epsilon C)v}{\|(A + \epsilon C)v\|_2} = \frac{\lambda v + \epsilon C v}{\|\lambda v + \epsilon C v\|_2} \\ &= \frac{\lambda v + \epsilon C v}{\sqrt{\lambda^2 v^T v + 2\epsilon \lambda v^T C v + \epsilon^2 v^T C^T C v}} \\ &\approx \frac{\lambda v + \epsilon C v}{\lambda}, \end{aligned} \quad (8)$$

where ϵ terms in the denominator have been neglected. Thus

$$\Delta v \approx \frac{\epsilon}{\lambda} C v. \quad (9)$$

Similarly, it can be found that

$$\Delta u \approx \frac{\epsilon}{\lambda} C^T u. \quad (10)$$

Therefore, the derivative $\lambda'|_{A+\epsilon C}$ can be approximated as

$$\begin{aligned} \lambda'|_{A+\epsilon C} &= \frac{(u + \Delta u)^T C (v + \Delta v)}{(u + \Delta u)^T (v + \Delta v)} \approx \frac{u^T C v}{u^T v} + \frac{u^T C \Delta v}{u^T v} + \frac{\Delta u^T C v}{u^T v} \\ &\approx \frac{u^T C v}{u^T v} + \frac{2\epsilon u^T C C v}{\lambda u^T v}, \end{aligned} \quad (11)$$

having neglected terms containing Δu and Δv in the denominator, and the product $\Delta u^T C \Delta v$ in the numerator. Thus, an improved approximation for $\Delta\lambda$ is given by

$$\Delta\lambda \approx \Delta_1 + \Delta_2 \approx \frac{u^T \epsilon C v}{u^T v} + \frac{1}{\lambda} \frac{u^T \epsilon^2 C^2 v}{u^T v}. \quad (12)$$

The above formulas take advantage of the fact that in many situations the eigenvector does not change much by the perturbation, and thus a few steps (one or two) of the power method already give a very accurate approximation, as we shall see in the next section. This, however, is not the case when the dominant eigenvalue λ is nearly degenerate, in which case the convergence rate of the power method is $|\frac{\lambda_2}{\lambda}| \approx 1$, where λ_2 is the second largest eigenvalue in absolute value. More precisely, the accuracy of our approximation is related to the spectral gaps through the perturbation formula, which, for undirected networks, takes the form [19]

$$\Delta\lambda = \epsilon v^T C v + \epsilon^2 \sum_{i=2}^n \frac{(v^{(i)} C v)^2}{\lambda - \lambda_i} + o(\epsilon^2),$$

where λ_i and $v^{(i)}$ ($i \geq 2$) are the nondominant eigenvalues and corresponding eigenvectors of A . When $\lambda_1 \approx \lambda_2$, the second term becomes large, indicating that λ is highly sensitive to perturbations.

B. Adding or removing an arbitrary subgraph

Most of the interesting problems in the context of networks involve discrete changes, where A becomes $A+B$, and the nonzero entries of B are finite and usually of the same order of magnitude as the nonzero entries of A . If, however, the modifications are limited to a small number of links in a large network, then $\|B\|_2 \ll \|A\|_2$, and the perturbation results (5) and (6) would be valid with ϵC replaced by B . The other approximations we have made are also likely to be valid, and this is supported by the fact that Eq. (12) improves significantly over Eq. (5) for the example networks discussed in Sec. III. Equations (5) and (12) then become

$$\Delta\lambda \approx \Delta_1 = \frac{u^T B v}{u^T v}, \quad (13)$$

$$\Delta\lambda \approx \Delta_1 + \Delta_2 \approx \frac{u^T B v}{u^T v} + \frac{1}{\lambda} \frac{u^T B^2 v}{u^T v}, \quad (14)$$

and our approximation schemes for the spectral impact I_B for small but finite modifications are

$$\hat{I}_B = \frac{1}{\lambda} \frac{u^T B v}{u^T v}, \quad (15)$$

$$\hat{\hat{I}}_B = \frac{1}{\lambda} \frac{u^T B(v + Bv/\lambda)}{u^T v}. \quad (16)$$

Equation (15) is linear in B , and therefore the change can be decomposed into the sum of elementary changes, as $B = \sum_i B_i$, where B_i can represent, for example, a modification of a single edge. The corresponding first-order approximation for the SI is obtained from the individual contributions, as $\hat{I}_B = \sum_i \hat{I}_{B_i}$. On the other hand, Eq. (16) has a linear and a quadratic dependence on B and linear superposition cannot in general be used. However, if all the products of elementary changes $B_i B_j$ are zero matrices (for example, if B_i 's represent the disconnected components of the subgraph), then $\hat{\hat{I}}_B = \sum_i \hat{\hat{I}}_{B_i}$.

When the changes are limited to a small number of links, we have a clear computational advantage of having highly sparse matrix B . Also, including changes in the eigenvector does not imply much computational burden. In fact, the extra computational cost of using Eq. (14) instead of Eq. (13) amounts to computing the extra vector $u^T B$ and projecting it onto Bv [20].

C. Modifying weight on single edge

Adding weight b_{ij} to directional edge (i, j) of a graph [21] corresponds to changing the adjacency matrix from A to A

+ B_{ij} , where B_{ij} is an $n \times n$ matrix containing all zeros except for the (i, j) entry b_{ij} . The spectral impact $I_{B_{ij}}$ can be approximated, using Eq. (15), as

$$\hat{I}_{B_{ij}} = \frac{b_{ij} u_i v_j}{\lambda u^T v}. \quad (17)$$

This is equivalent to Eq. (2), in the case $b_{ij} = -a_{ij}$.

The second-order approximation (16) introduces a correction only if a modification on a self-loop is introduced, as $(B^2)_{ij}$ is zero otherwise. Indeed, $I_{B_{ij}}$ can be approximated at the second order as

$$\hat{\hat{I}}_{B_{ij}} = \frac{b_{ij} u_i v_j}{\lambda u^T v} \left(1 + \delta_{ij} \frac{b_{ii}}{\lambda} \right), \quad (18)$$

where δ_{ij} is the Kronecker delta.

If the change is introduced *bidirectionally*, that is, b_{ij} is added to the weight of edge (i, j) , and b_{ji} to edge (j, i) , then the first- and second-order approximations to the SI become

$$\hat{I}_{B_{ij,ji}} = \frac{b_{ij} u_i v_j + b_{ji} u_j v_i}{\lambda u^T v} \quad (19)$$

and

$$\hat{\hat{I}}_{B_{ij,ji}} = \frac{b_{ij} u_i v_j + b_{ji} u_j v_i}{\lambda u^T v} + \frac{b_{ij} b_{ji} (u_i v_i + u_j v_j)}{\lambda^2 u^T v}. \quad (20)$$

In this case Eq. (20) can have nonzero correction term for modification of weights on an edge other than a self-loop. For undirected networks, modification must be bidirectional and symmetric ($b_{ij} = b_{ji}$), and the formulas become

$$\hat{I}_{B_{ij}} = \frac{2b_{ij} v_i v_j}{\lambda} \quad (21)$$

and

$$\hat{\hat{I}}_{B_{ij}} = \frac{2b_{ij} v_i v_j}{\lambda} + \frac{b_{ij}^2 (v_i^2 + v_j^2)}{\lambda^2}. \quad (22)$$

Note that Eq. (22) always introduces a non-negative correction to the first-order approximation for modifications to undirected networks.

D. Removing a node

Removing node k in a graph corresponds to removing all edges touching it. In this case, the entries of B are $b_{ij} = -a_{ij}(\delta_{ik} + \delta_{jk} - \delta_{ik}\delta_{jk})$ and the first-order approximation reads

$$\hat{I}_k = \left(-2 + \frac{a_{kk}}{\lambda} \right) \frac{u_k v_k}{u^T v}, \quad (23)$$

while the second-order one yields

$$\hat{\hat{I}}_k = \left(-1 + \frac{a_{kk}}{\lambda} - \frac{a_{kk}^2}{\lambda^2} + \frac{1}{\lambda^2} \sum_{i=1}^n a_{ik} a_{ki} \right) \frac{u_k v_k}{u^T v}. \quad (24)$$

In the case that there is no self-loop connecting node k with itself, the above formulas simplify to

$$\hat{I}_k = -2 \frac{u_k v_k}{u^T v} \quad (25)$$

and

$$\hat{\hat{I}}_k = \left(-1 + \frac{1}{\lambda^2} \sum_{i=1}^n a_{ik} a_{ki} \right) \frac{u_k v_k}{u^T v}. \quad (26)$$

The summation term in the last equation represents an improvement over the previous result, Eq. (3).

If, furthermore, the network is undirected and unweighted, first and second-order approximations become

$$\hat{I}_k = -2v_k^2 \quad (27)$$

and

$$\hat{\hat{I}}_k = \left(-1 + \frac{d_k^{\text{out}}}{\lambda^2} \right) v_k^2, \quad (28)$$

where $d_k^{\text{out}} \equiv \sum_{i=1}^n a_{ki}$ is the *out-degree* of vertex k .

The last two equations show well that, in the case of removing a node, first- and second-order approximations yield rather different results, with the estimate from Eq. (27) more than double that from Eq. (28). This difference comes from the fact that, in the case of removing node k , the k th component of the new dominant eigenvector becomes zero, regardless of its previous value, and thus the change in u and v is not negligible.

III. APPLICATIONS

In this section, the accuracy of various approximations is assessed using both synthetic and real-world graphs. The true dominant eigenvalue and corresponding eigenvector are estimated in double precision using the MATLAB function `eigs` [22] and compared to the estimates obtained by implementing our approximation formulas.

As an example of artificial networks, the Erdős-Rényi random graph [23] is used, with $n=1000$ nodes and the probability of connection $p=0.01$. The particular realization used is labeled \mathcal{G}_1 for convenience and has 5004 undirected links, without self-loops. The largest degree is $d_{\max}=20$, the minimum is $d_{\min}=2$, and the average is $d_{\text{mean}}=10.01$. The largest three eigenvalues (in magnitude) of the corresponding adjacency matrix are computed to be $\lambda=11.0741$, $\lambda_2=-6.53518$, and $\lambda_3=6.50196$. The components of the eigenvector v are shown in increasing order in Fig. 1(a). We also analyze three real-world networks, which we refer to as \mathcal{G}_2 , \mathcal{G}_3 , and \mathcal{G}_4 for convenience. The basic properties of these networks are reported in Table I, along with the pertinent references. \mathcal{G}_2 is a biological example, \mathcal{G}_3 is a social interaction network, while \mathcal{G}_4 can be regarded as an instance having both an engineering and social character.

The impact on λ of the removal of various subgraphs of \mathcal{G}_1 is analyzed next: the results from the first- and second-order approximations are plotted versus the actual SI. In Fig. 1(b), the removal of edges is considered; the maximum reduction in λ is about 0.1% and it is satisfactorily predicted by both the first- and second-order formulas: the former has

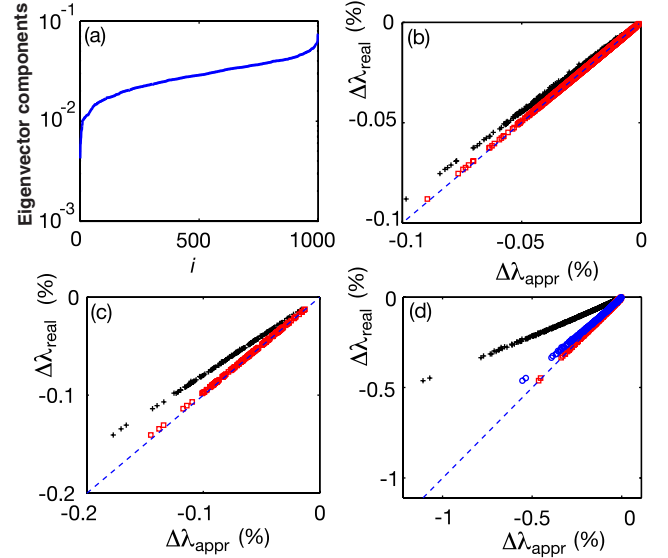


FIG. 1. (Color online) Results for the Erdős-Rényi network \mathcal{G}_1 . (a) Eigenvector components sorted in the increasing order. The true SI is plotted against its approximation (both in percentage) for the removal of (b) edges, (c) triangles, and (d) nodes. The black plus symbols correspond to the first-order approximation (15) and the red (light gray) squares to the improved approximation (16). In (d) the blue (dark gray) circles correspond to the approximation in Ref. [11].

a tendency of overestimating the change, while the latter is more accurate. As expected, there are edges that have a greater impact on λ than others. In particular, the impact of edge (i, j) is dictated by the components v_i and v_j of v , as given in Eqs. (21) and (22). The effect of removing a simple *motif* is shown in Fig. 1(c), where the 171 triangles occurring in \mathcal{G}_1 are individually removed. The observed $\Delta\lambda$ is higher in this case, though the approximations are still satisfactory. Figure 1(d) investigates the removal of nodes, comparing Eqs. (27), (28), and (3): while the first-order approximation is off by a factor of about 2, the second-order formula with the d_k/λ^2 term is the most accurate. Note that for both the first- and second-order estimates the plots are almost mono-

TABLE I. Examples of real-world networks.

| |
|--|
| \mathcal{G}_2 , yeast protein interaction network [24] |
| 2361 vertices, 13828 edges |
| $d_{\min}=1$, $d_{\text{mean}}=5.86$, $d_{\max}=65$ |
| $\lambda=19.4861$, $\lambda_2=16.1340$, $\lambda_3=14.3339$ |
| \mathcal{G}_3 , network of e-mail interchanges [25] |
| 1133 vertices, 5451 edges |
| $d_{\min}=1$, $d_{\text{mean}}=9.62$, $d_{\max}=71$ |
| $\lambda=41.4940$, $\lambda_2=33.9272$, $\lambda_3=30.0687$ |
| \mathcal{G}_4 , U.S. power grid ^a 4941 vertices, 6594 edges |
| $d_{\min}=1$, $d_{\text{mean}}=2.67$, $d_{\max}=19$ |
| $\lambda=7.4831$, $\lambda_2=6.6092$, $\lambda_3=5.5728$ |

^aThe data was retrieved from P. Tsaparas' webpage on data sets and codes for complex networks, at <http://www.cs.helsinki.fi/u/tsaparas/MACN2006/data-code.html>.

TABLE II. Predicting the relative ranking of edges and nodes based on SI. The numbers indicate the fraction of all possible pairs of edges or nodes for which the relative ranking is correctly predicted by each approximation scheme. The local methods are based on the eigenvector components estimated by the normalized degree (LM1), the normalized sum of the neighbors' degree (LM2), and the normalized sum of the degrees of the neighbors' neighbors (LM3).

| Networks | Ranking of edges | | | | | Ranking of nodes | | | | |
|-----------------|------------------|-----------------|---------------|------------|------------|------------------|-----------------|---------------|------------|------------|
| | Eq. (13) (%) | Eq. (14) (%) | Local Methods | | | Eq. (13) (%) | Eq. (14) (%) | Local Methods | | |
| | | | LM1 (%) | LM2 (%) | LM3 (%) | | | LM1 (%) | LM2 (%) | LM3 (%) |
| \mathcal{G}_1 | 99.34 | 99.95 | 90.20 | 96.63 | 98.04 | 99.59 | 99.94 | 93.79 | 96.81 | 97.86 |
| \mathcal{G}_2 | 98.46 | 99.95 | 82.31 | 88.01 | 90.13 | 99.85 | 99.96 | 84.30 | 88.81 | 91.52 |
| \mathcal{G}_3 | 98.96 | 99.97 | 85.08 | 91.40 | 93.85 | 99.86 | 99.95 | 87.65 | 91.88 | 94.64 |
| \mathcal{G}_4 | 96.86 | 96.51 | 83.68 | 91.07 | 93.60 | 96.59 | 95.92 | 85.91 | 82.40 | 81.99 |

tonic, indicating that the relative ranking of nodes, edges, and triangles defined by the SI are accurately estimated by these formulas. Indeed, for a randomly chosen pair of nodes or edges, the estimate is correct with probability close to one (Table II).

Figure 2 shows the eigenvector components, the real SI, and the approximated SI for removing an edge and removing a node from the real-world networks \mathcal{G}_2 , \mathcal{G}_3 , and \mathcal{G}_4 . The eigenvector components are shown in the order of increasing magnitude; in several cases, the smallest components appear

to be rather small and fall below the axis limit on the figure. The presence of components in the dominant eigenvector spanning several orders of magnitude amounts to large discrepancies in the importance of edges. When the removal of edges is analyzed, both the first- and second-order approximations for the SI are satisfactory. On the other hand, if removal of nodes is considered, the second-order formula of Eq. (28), containing a correction for the degree of the node, yield results more accurate than Eq. (3). In this case, the SI is as large as -7% . The relative ranking based on SI is also

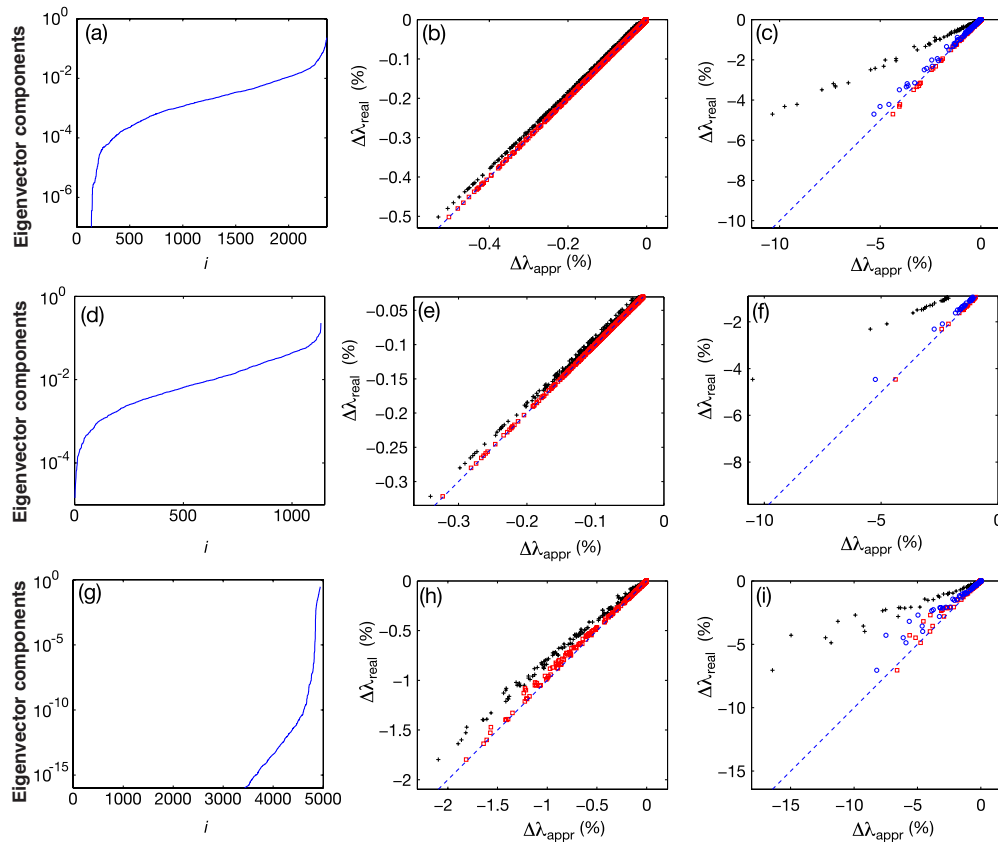


FIG. 2. (Color online) Results for the three examples of real-world networks \mathcal{G}_2 (top row), \mathcal{G}_3 (middle row) and \mathcal{G}_4 (bottom row) in Table I. The left column [(a), (d), and (g)] shows the eigenvector components sorted in the increasing order. The middle column [(b), (e), and (h)] shows the true SI vs approximated SI for the removal of edges, while the right column [(c), (f), and (i)] is for the removal of nodes. The meaning of the symbols is the same as in Fig. 1.

accurately predicted for these real-world networks, as shown in Table II. Note that increasing the order of approximation improves the accuracy, except for the case of \mathcal{G}_4 . For node removals in \mathcal{G}_4 , the rank prediction accuracy, although high in general, falls slightly with the increasing order of approximation. This is due to relatively large fluctuation of the approximation error among different nodes [Fig. 2(i)], which is a likely consequence of heterogeneity and hierarchical nature of the power-grid network topology.

Since the spectral gap $\lambda - \lambda_2$ mainly determines the sensitivity of λ to structural perturbations, it is directly related to the accuracy of our approximation formulas. Indeed, the larger the spectral gap, the better the approximation will generally be. Networks with large spectral gaps are known to be homogeneous and well-connected, avoiding structural bottlenecks [26], and can also be characterized by having large expansion constant [27,28].

IV. APPROXIMATING SI USING LOCAL INFORMATION

In Sec. II, formulas for approximating the SI were given, based on Eq. (15). However, the knowledge of both the largest eigenvalue λ and its corresponding left and right eigenvectors u and v is required. Such information is in some situations impractical or even impossible to obtain, since it is equivalent to solving an eigenvalue problem for the adjacency matrix A , which requires knowing all the entries of A .

For a given set of links to be modified, however, our approximation formulas only require—aside from a normalization constant—a few components of u and v corresponding to the nodes attached to these links. These u and v components can be approximated by iterative methods. When using the SI for ranking different subgraphs, such as single edges or nodes, or pairs of edges or nodes, this approach can be useful because the normalization constant does not affect the ranking.

Assuming that the spectrum of A satisfies $|\lambda| > |\lambda_2| \geq \dots \geq |\lambda_n|$, one can adopt the power method [15–17] to solve for the dominant eigenvalue λ and its corresponding left and right eigenvectors u and v . The starting point of this method is a normalized vector $v^{(0)}$, that in this case can be taken as

$$v^{(0)} = \frac{1}{\sqrt{n}} [1, 1, \dots, 1]^T. \quad (29)$$

Then, for $t=1, 2, \dots$, until convergence, the following is iterated:

$$y^{(t)} = Av^{(t-1)}, \quad \lambda^{(t)} = \|y^{(t)}\|_2, \quad v^{(t)} = y^{(t)} / \|y^{(t)}\|_2. \quad (30)$$

The convergence of both the eigenvalue and eigenvector is geometric, with rate $O(|\lambda_2/\lambda|^t)$ [29]. This algorithm can be straightforwardly adjusted for the computation of the left eigenvector u , and the same convergence rates apply. Although other iterative schemes for the computation of the dominant eigenvalue and the corresponding eigenvector are available, the power method is used here since its iterations directly highlight the local information of a graph.

Indeed, the first iteration of the power method gives the degree, the number of connections that each node has, up to

a normalization constant. The k th component of $u^{(1)}$ is proportional to the *in-degree* of node k , $d_k^{\text{in}} = \sum_{i=1}^n a_{ik}$, while the k th component of $v^{(1)}$ scales with the *out-degree* of the same node, $d_k^{\text{out}} = \sum_{i=1}^n a_{ki}$. If the graph is undirected and unweighted, the k th component of both $u^{(1)}$ and $v^{(1)}$ are proportional to the number of direct neighbors that node k has (counting itself, if a self-loop is in place).

The second iteration of the power method returns information about the number of connections that these neighbors have. For an undirected unweighted graph, the k th component of $u^{(2)}$ and $v^{(2)}$ is proportional to the sum of the degrees of the direct neighbors (where the same node can be counted several times, and node k itself is always included in the count). If the graph is directed or weighted, the proper weights need to be added and the directions of the connections considered. Subsequent iterations provide better approximations of the k th eigenvector component, involving larger neighborhood of node k .

Combing these with Eq. (15), we obtain successive approximations to the SI. For edge (i, j) in an undirected unweighted network, for example, the first two iterations give

$$\hat{I}_{B_{ij}}^{(1)} \sim d_i^{\text{in}} d_j^{\text{out}} \quad (31)$$

$$\hat{I}_{B_{ij}}^{(2)} \sim \left(\sum_{k=1}^n a_{ki} d_k^{\text{in}} \right) \left(\sum_{k=1}^n a_{jk} d_k^{\text{out}} \right). \quad (32)$$

Figure 3 shows the results of the local computation for the Erdős-Rényi graph \mathcal{G}_1 . Similar results can be obtained for the networks \mathcal{G}_2 , \mathcal{G}_3 , and \mathcal{G}_4 , which allow accurate prediction of edge- and node-ranking, as shown in Table II. Analogously to the effect of the order of approximation, increasing the number of iterations for the local method improves the accuracy for all cases, except for \mathcal{G}_4 (due to the same large fluctuation of the approximation error among different nodes). The accuracy of this local approximation also depends on the ratio $|\lambda_2/\lambda|$, since it dictates the convergence rate of the method.

V. PERTURBATION OF THE GRAPH LAPLACIAN

Perturbation results, based on either global or local information, can analogously be derived for eigenvalues and eigenvectors associated with the *graph Laplacian* L , defined as $L = D - A$, where D is the diagonal matrix of node in-degrees. For simplicity, only connected and undirected graphs are considered, so that the associated graph Laplacians are positive-semidefinite matrices. The spectrum of L is assumed to satisfy

$$0 = \mu_1 < \mu_2 < \mu_3 \leq \dots \leq \mu_{n-1} < \mu_n, \quad (33)$$

so that μ_2 and μ_n are nondegenerate. The second smallest eigenvalue, μ_2 , and the largest one, μ_n , are often used to characterize properties of the underlying graph. In particular, μ_2 is known as the *algebraic connectivity* of the graph [30] and quantifies the connectedness of the graph ($\mu_2=0$ if the graph is disconnected). The algebraic connectivity, as well as the eigenratio μ_2/μ_n , are closely related to the stability of synchronized states in coupled dynamical systems [31,32].

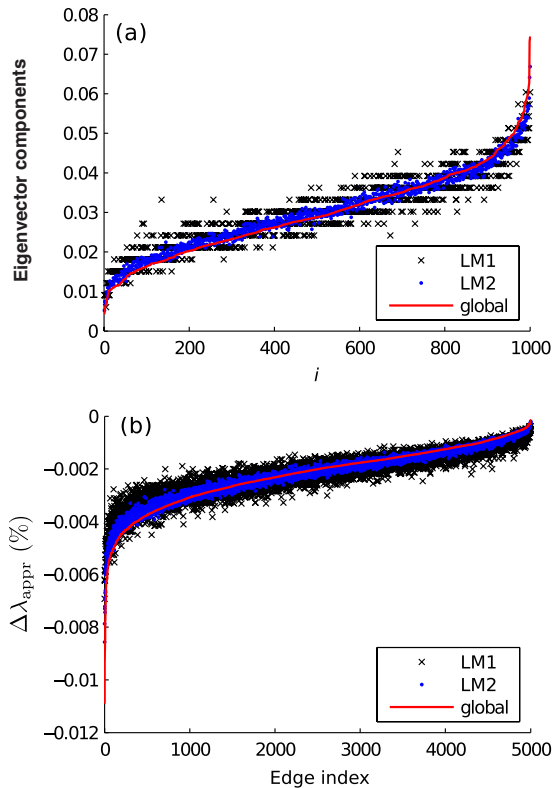


FIG. 3. (Color online) Convergence of local SI approximation for edge removal in the Erdős-Rényi network \mathcal{G}_1 . (a) The eigenvector components sorted in the increasing order of their corresponding values computed globally [red (light gray) line]. LM1 denotes the approximation by the normalized degree (black cross symbols), while LM2 denotes the approximation by the normalized sum of the degrees of the neighbors [blue (dark gray) dots]. (b) The SI for edge removal estimated using the approximations of the eigenvector in (a), showing quick convergence to the SI computed using the globally computed eigenvector. The first-order formula (21) was used to estimate the SI.

Denote by x and y the normalized eigenvectors related to μ_2 and μ_n , respectively,

$$Lx = \mu_2 x, \quad Ly = \mu_n y, \quad \|x\|_2 = \|y\|_2 = 1. \quad (34)$$

If an arbitrary subgraph (assumed to be symmetric and with no self-loop) is added to or removed from the original graph, then the adjacency matrix changes from A to $A+B$. The change in μ_2 and μ_n can then be approximated, at the first order, by

$$\Delta\mu_2 \approx \sum_{i<j} b_{ij}(x_i - x_j)^2 \quad (35)$$

and

$$\Delta\mu_n \approx \sum_{i<j} b_{ij}(y_i - y_j)^2. \quad (36)$$

Combining the two equations above and neglecting terms containing more than one Δ (either $\Delta\mu_2$ or $\Delta\mu_n$) the change in the eigenratio $r = \mu_2 / \mu_n$ is predicted by

$$\Delta r \approx \frac{1}{\mu_n^2} \sum_{i<j} b_{ij} [\mu_n(x_i - x_j)^2 - \mu_2(y_i - y_j)^2]. \quad (37)$$

Similarly to the formula for the dominant eigenvalue of A , the above equations can be adopted to develop strategies for targeting the network evolution toward some desired state, for example, to enhance (or weaken) the network synchronizability. Our formalism can also be extended to other spectral quantities.

VI. SUMMARY AND CONCLUSIONS

In this paper, we have introduced the concept of spectral impact of an arbitrary link modification in a network as the relative change in the largest eigenvalue of the adjacency matrix induced by the modification. Based on the standard perturbation method and an approximation for the second-order term, we obtained improved approximation formulas for the spectral impact that requires only the most dominant eigenvalue of the original network and its left and right eigenvectors. Using the Erdős-Rényi random graph, as well as real-world examples of large complex networks from biological, social, and technological applications, we confirmed the accuracy of the formulas for the addition and/or removal of nodes, links, and triangles. We have also shown that the first iteration of a local approximation scheme based on the power method is equivalent to using the node (or subgraph) degree for ranking, and that further iterations quickly improve the accuracy by incorporating the connectivity structure of larger neighborhood of the node (or subgraph). The analysis leading to the approximation formulas readily applies to the spectrum of other relevant matrices associated with the network, such as the Laplacian matrix treated briefly in this paper and the biased adjacency matrix studied in Ref. [33].

Some problems on the approximation schemes still remain open. How does the network topological structures, such as the small-world, scale-free, and modular structures, affect the accuracy of the approximations? More generally, how does the robustness of the network with respect to its spectral properties depend on the network structure, and can it be used to classify networks, similarly to the existing spectral classification [26]? How can we appropriately measure the “smallness” of perturbation B to predict the accuracy? It is also important to extend our method to the case of degenerate dominant eigenvalues, which may arise when the network evolved under constraints or under pressure to optimize its functions [34,35].

Our results have several potential applications for large networks whose performance depends on their spectral properties. The approximation schemes may be used in a damage control strategy for such networks, in which sudden structural damage that cannot be immediately fixed, such as the removal of multiple edges or nodes, is compensated by changes in other parts of the network (see Ref. [36] for an example of such a compensatory perturbation in a different context). They may also be used to develop gradient-descent-like algorithms to solve the problem of designing networks that satisfy specific spectral (and thus dynamical) properties

[37–39], or to develop efficient updating schemes for spectrum-based statistics of evolving networks, similar to those for local statistics reported in Ref. [12]. The ranking of subgraphs, or motifs, according to the spectral impact in a given network and for a selected eigenvalue (not necessarily the largest) gives rise to interesting questions: how does this ranking depend on the subgraph, the choice of the eigenvalue, and local and global properties of the network? Our improved formula reflects the fact that there is a nonlinear effect: the SI of the union of two subgraphs is not simply the sum of their individual SI. How does this nonlinear effect correlate with topological structures such as communities?

With the tools developed in this paper, we wish to tackle some of the above open problems and applications in our future work.

ACKNOWLEDGMENTS

A.M. gratefully acknowledges the partial support of the MAE Department through an instructor's assistantship. J.S. has been supported by the ARO Grant No. 51950-MA. The authors would like to thank D. ben-Avraham, A. Alhakim, E. M. Bollt, and J. D. Skufca for numerous discussions.

-
- [1] D. Watts and S. Strogatz, *Nature (London)* **393**, 440 (1998).
 [2] A.-L. Barabási and R. Albert, *Science* **286**, 509 (1999).
 [3] R. Albert and A.-L. Barabási, *Rev. Mod. Phys.* **74**, 47 (2002).
 [4] M. E. J. Newman, *SIAM Rev.* **45**, 167 (2003).
 [5] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, *Rev. Mod. Phys.* **80**, 1275 (2008).
 [6] E. Estrada and J. A. Rodríguez-Velázquez, *Phys. Rev. E* **71**, 056103 (2005).
 [7] J. M. Kleinberg, *J. ACM* **46**, 604 (1999).
 [8] A. N. Langville and C. D. Meyer, *SIAM Rev.* **47**, 135 (2005).
 [9] C. R. MacCluer, *SIAM Rev.* **42**, 487 (2000).
 [10] J. G. Restrepo, E. Ott, and B. R. Hunt, *Phys. Rev. Lett.* **96**, 254103 (2006).
 [11] J. G. Restrepo, E. Ott, and B. R. Hunt, *Phys. Rev. Lett.* **97**, 094102 (2006).
 [12] J. Sun, J. P. Bagrow, E. M. Bollt, and J. D. Skufca, *Phys. Rev. E* **79**, 036116 (2009).
 [13] P. Lancaster and M. Tismenetsky, *The Theory of Matrices with Applications*, 2nd ed. (Academic Press, New York, 1985).
 [14] This name is chosen to reflect the fact that the underlying measure is the spectral radius and that we consider not only removal but also addition of links.
 [15] J. H. Wilkinson, *The Algebraic Eigenvalue Problem* (Oxford University Press, Oxford, UK, 1965).
 [16] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed. (Johns Hopkins University Press, Baltimore, 1996).
 [17] J. Demmel, *Applied Numerical Linear Algebra* (SIAM, Philadelphia, 1997).
 [18] Details on the power method are reported in Sec. IV. Here, the assumption is that v and $v + \Delta v$ are close to being parallel, and hence a single iteration of the power method is satisfactory. The spectral gap of $A + \epsilon C$ is not critical for this approximation.
 [19] Here the formula is presented for undirected networks for convenience, and the argument holds similarly in general cases.
 [20] The vector Bv is considered to be available from the first-order approximation. Alternatively, one could proceed by computing $u^T B$ at the first order and then Bv : the increase in the computational cost is, in general, the same.
 [21] As particular case, taking $b_{ij} = -a_{ij}$ amounts to deleting the edge (i, j) . On the other hand, if $a_{ij} = 0$, then adding b_{ij} means introducing a directed connection with a certain weight.
 [22] *MATLAB User's Guide* (The MathWorks Inc., Natick, 2006).
 [23] B. Bollobas, *Modern Graph Theory* (Springer, New York, 1998).
 [24] D. Bu, Y. Zhao, L. Cai, H. Xue, X. Zhu, H. Lu, J. Zhang, S. Sun, L. Ling, N. Zhang *et al.*, *Nucleic Acids Res.* **31**, 2443 (2003). The data can be downloaded from the University of Florida sparse matrix collection, at <http://www.cise.ufl.edu/research/sparse/matrices>
 [25] R. Guimera, L. Danon, A. Diaz-Guilera, F. Giralt, and A. Arenas, *Phys. Rev. E* **68**, 065103(R) (2003), the data are available at the webpage of Dr. Arenas, <http://deim.urv.cat/~aarenas/data/xarxes/email.zip>
 [26] E. Estrada, *Phys. Rev. E* **75**, 016103 (2007).
 [27] S. Hoory, N. Linial, and A. Wigderson, *Bull. Am. Math. Soc.* **43**, 439 (2006).
 [28] N. Alon and V. D. Milman, *J. Comb. Theory, Ser. B* **38**, 73 (1985).
 [29] This convergence result is easy to obtain if A is diagonalizable. For nondiagonalizable cases, see Ref. [15] for details.
 [30] M. Fiedler, in *Combinatorics and Graph Theory*, Banach Center Publications Vol. 25 (PWN-Polish Scientific Publishers, Warsaw, 1989), pp. 57–70.
 [31] L. M. Pecora and T. L. Carroll, *Phys. Rev. Lett.* **80**, 2109 (1998).
 [32] J. Sun, E. M. Bollt, and T. Nishikawa, *Europhys. Lett.* **85**, 60011 (2009).
 [33] E. Ott and A. Pomerance, *Phys. Rev. E* **79**, 056111 (2009).
 [34] T. Nishikawa and A. E. Motter, *Physica D* **224**, 77 (2006).
 [35] T. Nishikawa and A. E. Motter, *Phys. Rev. E* **73**, 065106(R) (2006).
 [36] A. E. Motter, N. Gulbahce, E. Almaas, and A.-L. Barabasi, *Mol. Syst. Biol.* **4**, 168 (2008).
 [37] A. Hagberg, P. J. Swart, and D. A. Schult, *Phys. Rev. E* **74**, 056116 (2006).
 [38] J. Sun, T. Nishikawa, and D. ben-Avraham, *Phys. Rev. E* **78**, 026104 (2008).
 [39] A. Hagberg and D. A. Schult, *Chaos* **18**, 037105 (2008).