Journal of Complex Networks (2023) **4**, Advance Access Publication on 11 July 2023
https://doi.org/10.1093/comnet/cnad026

Spectral techniques for measuring bipartivity and producing partitions

Azhar Aleidan and Philip A. Knight†
Department of Mathematics and Statistics, University of Strathclyde, Richmond Street, Glasgow G1 1XH, UK
†Corresponding author. Email: p.a.knight@strath.ac.uk

[Received on 3 March 2023; editorial decision on 20 June 2023; accepted on 26 June 2023]

Complex networks can often exhibit a high degree of bipartivity. There are many well-known ways for testing this, and in this article, we give a systematic analysis of characterizations based on the spectra of the adjacency matrix and various graph Laplacians. We show that measures based on these characterizations can be drastically different results and leads us to distinguish between local and global loss of bipartivity. We test several methods for finding approximate bipartitions based on analysing eigenvectors and show that several alternatives seem to work well (and can work better than more complex methods) when augmented with local improvement.

Keywords: bipartite graph; graph spectra; modularity.

1. Introduction

Complex networks exist in many disparate fields of study including maths, social sciences and informatics. Describing these systems enables us to discover useful properties. One such property is exhibited by so-called 'two-mode' networks, where two sets of nodes are connected by links that represent the relationships between them are described as bipartite [1]. Even networks that are not truly bipartite may show more bipartivity than would be seen in a completely random network. The degree of bipartivity in a network can be measured [2] with a view to exploiting this structure where present, or correcting for it, to increase efficiency.

A number of authors have given examples of real-world networks, where identifying closeness to bipartivity gives useful insight. For example, Kunegis [3] observes that in networks formed through sexual selection, near bipartivity is a common property; and in [4], the authors draw a parallel between stability of fullerene isomers and the nearness to bipartivity of graph models of their molecular form. In [5], the authors observed that one can find a degree of bipartivity in a network depicting communication between airports; that is, most communication is between the partitions while there is little between members of the same group. The authors focused on European airline networks to see how varying degrees of bipartivity affect their efficiency. The authors showed that that traditional airlines appear to be much more bipartite than low cost airlines. If we measure transportation efficiency as the ratio of the number of passengers to the number of hours flown by a carrier, it can be seen that a low degree of bipartivity in a transportation network drives airline efficiency. Without a way to measure bipartivity, inefficiencies like the one just discussed would be more difficult to identify [1].

In order to measure bipartivity, one has to characterize the property. In this article, we look at how different characterizations have been used, and can be extended, to measure the degree of bipartivity within a network. We show that the different characterizations can lead to profoundly different conclusions about the level of bipartivity within a network. For example, we show that is possible to generate a network that is arbitrarily close to being bipartite by one measure while being arbitrarily far by another.

We compare the various *a priori* measures on synthetic and real-world data in order to determine which is the most useful in predicting whether a network is in any sense close to bipartite.

Finally, we test a number of existing algorithms that have been proposed for finding approximate bipartitions and compare them with simple methods inspired by the equivalent characterizations of exact bipartivity on a number of real-world networks. There is variation in performance, but this can be mitigated against, and we find that we can regularly improve results with by some simple post-processing via local improvement.

The article is organized as follows. First, we introduce various characterizations of bipartivity from which we derive a series of measures. We then give an analysis of these measures before presenting a number of algorithms for finding bipartite perturbations of networks. We examine the performance of these algorithms before drawing conclusions.

Throughout the main body of this article, we focus on simple networks. We use the notation $G(V, E)$ to denote a network (or graph) G with n nodes (or vertices) $V = \{v_1, v_2, \ldots, v_n\}$ connected by m bidirectional edges $E = \{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m\}$, unless otherwise stated. Each edge ε_i is distinct and connects two distinct nodes v_{p_i} and v_{q_i}. We write $\varepsilon_i = (v_{p_i}, v_{q_i})$. We denote the adjacency matrix of G by $\mathbf{A}$, the complete graph with n nodes by K_n, the path graph with n edges by P_n and the complete bipartite graph with partitions of size n_1 and n_2 by $K_{n_1 n_2}$. Unless otherwise stated, our networks will be assumed to be strongly connected.

Matrices and vectors are denoted by bold letters, and we use subscripts to denote their dimensions when appropriate. We use $\mathbf{O}$, $\mathbf{I}$ and $\mathbf{E}$ to denote the zero matrix, the identity matrix and a matrix of ones, respectively, $\mathbf{1}$ to denote a vector of ones (subscripted with its dimension when necessary), and $\mathbf{e}_i$ to denote the ith column of $\mathbf{I}$.

2. Characterizing bipartivity

The classic definition of network bipartivity is as follows.

DEFINITION 2.1 A network is bipartite if the nodes of the network $G(V, E)$ can be divided into disjoint sets $V = V_1 \cup V_2$ such that for all $(u, v) \in E$, either $u \in V_1$ and $v \in V_2$ or $u \in V_2$ and $v \in V_1$.

There are a number of other characterizations that have been proposed in the literature, and many of these are collected in Theorem 2.2. Before, we state the theorem we introduce some notation that will be useful to us. We write the spectrum of G as

$$\sigma(G) = \{[\lambda_1]^{p_1}, [\lambda_2]^{p_2}, \ldots, [\lambda_l]^{p_l}\},$$

where $\lambda_1 > \lambda_2 > \cdots > \lambda_l$ are the distinct eigenvalues of $\mathbf{A}$ (which are real since G is undirected) and p_i is the multiplicity of λ_i (since G is connected $p_1 = 1$ by the Perron–Frobenius theorem). We note that if $f(\mathbf{A})$ is a well-defined function of a matrix then

$$\operatorname{tr}(f(\mathbf{A})) = \sum_{i=1}^{l} p_i f(\lambda_i).$$

Suppose $\mathbf{D}$ is the diagonal matrix whose entries are the degrees of the nodes of G, then the matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$ is known as it graph Laplacian and the matrix $\mathbf{Q} = \mathbf{A} + \mathbf{D}$ is called its signless Laplacian. The

normalized Laplacian is the matrix $\mathbf{N} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$. And given a partition of the nodes $P = \{V_1, V_2\}$ the modularity is given by

$$Q = Q(P) = \frac{1}{4m}\mathbf{s}^\top \left(\mathbf{A} - \frac{\mathbf{k}\mathbf{k}^\top}{2m}\right)\mathbf{s} = \frac{1}{4m}\mathbf{s}^\top \mathbf{B}\mathbf{s}, \tag{2.1}$$

where $\mathbf{k}$ is the vector of node degrees, and $\mathbf{s}$ is an indicator vector whose ith element is $+1$ if $v_i \in V_1$ and -1 if $v_i \in V_2$. We recall that modularity is used in community detection to measure the quality of a proposed partition P and that $-1/2 \leq Q \leq 1$ with a high value of Q indicating that P gives good community structure [6]. In the case of a partition into more than two sets the definition, we have given must be modified slightly. Also we will use subgraph centralization, which is defined to be the mean of all subgraph centralities of the nodes of a network [1], that is

$$\langle SC \rangle = \frac{1}{n}\sum_{i=1}^{n} \left(e^{\mathbf{A}}\right)_{ii} = \frac{1}{n}\mathrm{tr}(e^{\mathbf{A}}).$$

THEOREM 2.2 If $G(V, E)$ is a simple connected network, then the following conditions are equivalent.

1. G is bipartite.
2. The adjacency matrix of G can be permuted to $\begin{bmatrix} \mathbf{O} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{O} \end{bmatrix}$.
3. G has no cycles of odd length.
4. G has no closed walks of odd length.
5. $\lambda_1 = -\lambda_l$.
6. The spectrum of G is symmetric about 0.
7. $\mathrm{tr}(\sinh \mathbf{A}) = 0$.
8. $\langle SC \rangle = \frac{1}{n}\mathrm{tr}(\cosh \mathbf{A})$.
9. $\mathbf{Q}$ is singular.
10. $\mathbf{N}$ has maximum eigenvalue 2.
11. $Q(P) = -1/2$, where P is the bipartition of nodes.

None of these results is new; 2–6 are standard text book results and 7–11 can be found in [5–8]. We note that while the normalized and signless Laplacians can be used to characterize bipartivity, there appears to be no way to use the standard graph Laplacian. However, in Section 5, we show all three can be used to derive spectral algorithms for finding approximate bipartitions.

3. Bipartivity measures

Using Theorem 2.2, we can characterize closeness to bipartivity in myriad ways. In this section, we introduce a number of measures. We want to identify computationally tractable measures. These are generally inspired by algebraic properties of the adjacency matrix and Laplacians of a bipartite graph.

We note that almost all of the measures listed here, as well as simple variants, are well known in the literature.

Calculating the proportion of edges that destroy the bipartivity in a network gives an easy way to define the degree of network bipartivity. That is, we can measure the degree of bipartivity by counting the minimum number of edges we need to remove to create a perfectly bipartite network [9]. The edges we remove are commonly referred to as frustrated edges.

Let m_d be the number of such edges in G. Then, the bipartivity of G can be measured by $b_c = 1 - m_d/m$, where the subindex c is introduced to indicate that this index is combinatorial in nature [9]. Clearly, b_c equals 1 if and only if G is bipartite. And since, we never need to remove more than half the edges of a network to make it bipartite (the limiting case being K_n as $n \to \infty$) $b_c \geq 1/2$ for all networks. To make the comparison between measures more transparent, we will normalize the ones we study in this article so that they give values across the range $[0, 1]$ with the value of 1 occurring if and only if a graph is exactly bipartite.[1] For this reason, we will use the measure b_C in place of b_c, where

$$b_C = 1 - 2\frac{m_d}{m}.$$

This measure is impractical to calculate in practice because determining m_d is an NP complete problem [9] which is one of the reasons for introducing alternative measures for bipartivity.

Instead of using frustrated edges, one can use the fact that since a simple bipartite graph contains no odd cycles we can measure closeness to bipartivity by quantifying the frequency of odd cycles or closed walks and the proportion of even closed walks to the total number of closed walks is a measure of network bipartivity. This can be quantified using subgraph centralization with the measure

$$b_s = \frac{\operatorname{tr}(\cosh \mathbf{A})}{\operatorname{tr}(e^{\mathbf{A}})}, \tag{3.1}$$

since the entries of $\cosh \mathbf{A}$ give weighted sums of just the even length walks in a graph. Hence, $b_s \leq 1$ and $b_s = 1$ if and only if G is bipartite. Furthermore, as $\sinh(\lambda_j) \leq \cosh(\lambda_j)$, $\forall \lambda_j$, then $b_s \geq \frac{1}{2}$ [1]. The lower bound is again the limiting value for K_n since

$$b_s = \frac{\cosh(n-1) + (n-1)\cosh(-1)}{e^{n-1} + (n-1)e^{-1}}$$

and

$$\lim_{n\to\infty} b_s(K_n) = \lim_{n\to\infty} \frac{\cosh(n-1) + (n-1)\cosh(-1)}{e^{n-1} + (n-1)e^{-1}} = \frac{1}{2}.$$

A slight adaptation of (3.1) gives a measure of bipartivity in the range $[0, 1]$ as observed in [5]. This is carried out by using the traces of the hyperbolic cosine and sine matrix functions, namely

$$b_e = \frac{\operatorname{tr}(\cosh \mathbf{A}) - \operatorname{tr}(\sinh \mathbf{A})}{\operatorname{tr}(\cosh \mathbf{A}) - \operatorname{tr}(\sinh \mathbf{A})} = \frac{\operatorname{tr}(e^{-\mathbf{A}})}{\operatorname{tr}(e^{\mathbf{A}})}.$$

This can be written instead in terms of the eigenvalues of $e^{\mathbf{A}}$ and $e^{-\mathbf{A}}$. Again, $\lim_{n\to\infty} b_e(K_n) = 0$.

[1] We note that in [3], a series of measures are given which are zero if and only if the graph is bipartite.

One attractive feature of measures based on walks is the tangible relationship between the connectivity of the edges in the graph and bipartivity. Measures based purely on spectral properties, as described below, lack this direct connection. But, they are still popular in the literature and as we will see in our experiments, they can perform very well as a surrogate for frustrated edges.

The simplest spectral measure is probably the ratio

$$b_\lambda = \frac{|\,\lambda_l\,|}{\lambda_1},$$

where λ_1 and λ_l are the most positive and negative values of the adjacency matrix respectively. Since $\mathbf{A}$ is non-negative for any graph, $b_\lambda \leq 1$ by the Perron–Frobenius theorem and by Theorem 2.2 we have equality if and only if G is bipartite. On the other hand, since $\sigma(K_n) = \{[n-1]^1, [-1]^{n-1}\}$, $b_\lambda(K_n) = 1/(n-1) \to 0$ as $n \to \infty$. The measure b_λ is well known, for example, see [3].

Using the symmetry of the whole spectrum, we can measure closeness to bipartivity by measuring how much this symmetry is disturbed. For example,

$$\sigma_{\text{sym}}(G) = \sum_{i=1}^{l} = |\lambda_i + \lambda_{l+1-i}|$$

equals zero if and only if G is bipartite whereas it equals $n-2$ for K_n and so we can use

$$b_\sigma = \frac{1}{1+\sigma_{\text{sym}}(G)}$$

as another measure whose range is $[0,1]$. We can link $\text{sym}(G)$ to the number of odd closed walks as follows. For simplicity, suppose G has an even number of nodes and no repeated eigenvalues. If $f(x)$ is a function whose Maclaurin series has zero coefficients for even powers and odd coefficients $c_{2k+1} \geq 0$ (such as $\sinh x$) then

$$\text{tr}(f(\mathbf{A})) = \sum_{i=1}^{n} f(\lambda_i) = \sum_{i=1}^{n}\sum_{k=1}^{\infty} c_{2k+1}\lambda_i^{2k+1} = \sum_{k=1}^{\infty}\sum_{i=1}^{n/2} c_{2k+1}(\lambda_i^{2k+1} + \lambda_{n-i+1}^{2k+1})$$

and unless $\lambda_i + \lambda n - i + 1 = 0$, we get a positive contribution to $\text{tr}(f(\mathbf{A}))$. In practice, this measure seems to give no more insight than simpler ones, and we do not consider it further. However, we note that the symmetry is exploited in a similar way to derive an algorithm for finding bipartitions in [10].

We can use the properties of the signless and normalized Laplacians to come up with measures, too. For example, $b_Q = 1/(1+\lambda_{\min}(\mathbf{Q}))$ and $b_N = |1-\lambda_{\max}(\mathbf{N})|$ (noting that $\sigma(\mathbf{N}) \subset [0,2]$ for all simple graphs). Again, for K_n both of these measures tend to zero as $n \to \infty$. Kunegis introduces similar measures in [3] and notes that $\lambda_{\min}(\mathbf{Q})$ can be linked to the number of frustrated edges via a relaxation of a minimization of a quadratic form.

Finally, we observe that we can also use modularity, for example, with the measure $b_{\text{mod}} = 2(1-Q)/3$. This measure, though, is dependent on the bipartition used to find Q. From (2.1), we see every intra-partition edge gives a positive contribution to modularity, and so if we pick the optimal partition in terms of frustrated edges we should be able to minimize Q. But since this partition is *NP* hard to find, this is not something we can do in practice. In Section 5, we will consider a number of methods of finding

bipartitions, and the results of these algorithm can then be measured by b_{mod} (or more directly by Q). We note that if we link two copies of K_n with a single edge then the bipartition induced by splitting the nodes into the respective complete graphs gives a modularity that converges to 1 as $n \to \infty$.

Given the choice of measures, which should be used in practice if we want an effective tool for measuring near bipartivity? To answer that question, we will analyse some of the measures both theoretically and empirically. In Section 4, we show that there can be a great disparity between the measures and we look to see whether this is more than a theoretical effect in Section 6. It is also worth observing that there is some artifice in mapping quantities onto the range [0, 1], and there are many alternative ways of doing so. Another aspect that needs to be considered is the cost. Clearly, measures that are based on the whole spectrum (such as b_σ and b_e) carry a computational penalty. If cost is an issue, then we can use estimates: λ_1 and λ_l can be approximated cheaply, often to high accuracy, which is clearly an advantage for b_λ, but such estimates can also help with other measures. For example b_e can be approximated by $e^{\lambda_1}/e^{-\lambda_l}$.

4. Analysis of bipartivity measures

In this section, we give contrasting examples showing the disparity between bipartivity measures on a two families of networks.

Our first family extends a 'nearly' bipartite class of graphs, known as agave graphs, illustrated in Fig. 1. The network is generated by adding a single edge to K_{n2} (the edge is added to the partition with two nodes) and was used as an example by Holmes *et al.* in [2].

It is straightforward to compute the spectrum of A since it has rank 3 and the non-zero eigenvalues are easy to track down which makes it easy to compare our spectral bipartivity measures. Furthermore, b_c is easy to compute in this case since there is only one frustrated edge. On the other hand, as we increase the number of modes, the number of odd cycles increases rapidly and this leads to a significant difference between certain bipartivity measures. A simple generalization of the agave graph allows us to take this difference to the extreme.

THEOREM 4.1 There is a family of networks parameterized by n such that as $n \to \infty$, $b_\lambda \to 1$, $b_C \to 1$ and $b_e \to 0$.

Proof. We construct a graph G_n with $n^2 + n$ nodes, where n nodes are connected to every node in the graph (except themselves) but there are no intra-connections between the other n^2 nodes. If $n = 2$, this is an agave graph. The adjacency matrix of the graph G_n is

$$\mathbf{A} = \begin{bmatrix} \mathbf{K}_n & \mathbf{E}_{nn^2} \\ \mathbf{E}_{nn^2}^T & \mathbf{O} \end{bmatrix},$$

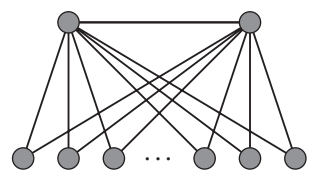


FIG. 1. An agave graph.

where $\mathbf{K}_n$ is the adjacency matrix of a complete graph. We can calculate the spectrum of $\mathbf{A}$ explicitly given its relatively low rank. The result can be found in [11, pp. 138–141], where this graph is an instance to the class of networks with an n-fully meshed star topology. We give a simple derivation of the result. To do this, we first apply the similarity transformation induced by $\mathbf{V} = \begin{bmatrix} \mathbf{I}_n & \mathbf{O} \\ \mathbf{O} & \mathbf{Z} \end{bmatrix}$, where $\mathbf{Z} = \frac{1}{n}\begin{bmatrix} 1 & \mathbf{1}^T \\ \mathbf{1} & -\mathbf{I}_{n^2-1} \end{bmatrix}$. Then,

$$\mathbf{V}^{-1}\mathbf{A}\mathbf{V} = \begin{bmatrix} \mathbf{K}_n & n\mathbf{1} & \mathbf{O} \\ n\mathbf{1}^T & 0 & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} \end{bmatrix}.$$

To find the eigenvalues of $\mathbf{X} = \begin{bmatrix} \mathbf{K}_n & n\mathbf{1} \\ n\mathbf{1}^T & 0 \end{bmatrix}$, we use the spectral decomposition of the complete graph $\mathbf{K}_n = \mathbf{Q}\mathbf{D}\mathbf{Q}^\top$ (where the eigenvalues are ordered so that the bottom right element of $\mathbf{D}$ is $n-1$, and hence the final column of $\mathbf{Q}$ is $\mathbf{1}/\sqrt{n}$). Thus, using the orthogonality of $\mathbf{Q}$, we find

$$\begin{bmatrix} \mathbf{Q}^\top & \mathbf{O} \\ \mathbf{O} & 1 \end{bmatrix}\mathbf{X}\begin{bmatrix} \mathbf{Q} & \mathbf{O} \\ \mathbf{O} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{D} & n\mathbf{Q}^\top\mathbf{1} \\ n\mathbf{1}^\top\mathbf{Q} & 0 \end{bmatrix} = \begin{bmatrix} -\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0}^\top & n-1 & n^{3/2} \\ \mathbf{0}^\top & n^{3/2} & 0 \end{bmatrix}.$$

Hence

$$\sigma(G_n) = \left\{[0]^{n^2-1}, [-1]^{n-1}, [f(n)+g(n)]^1, [f(n)-g(n)]^1\right\},$$

where

$$f(n) = \frac{n-1}{2}, \quad g(n) = \sqrt{\frac{(n-1)^2}{4} + n^3}.$$

Now, we compute the bipartivity measures.

Firstly, note that we can make the network bipartite by removing the $n(n-1)/2$ edges from the K_n block and hence

$$b_C \geq 1 - \frac{n(n-1)}{n^3 + n(n-1)/2} \geq 1 - \frac{1}{n}.$$

Next,

$$b_\lambda = \frac{g(n) - f(n)}{g(n) + f(n)} = 1 - o(n).$$

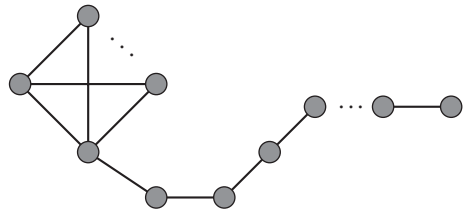


FIG. 2. A graph with a long tail.

Finally,

$$b_e = \frac{(n^2-1)+(n-1)e+e^{-f(n)-g(n)}+e^{-f(n)+g(n)}}{(n^2-1)+(n-1)e^{-1}+e^{f(n)+g(n)}+e^{f(n)-g(n)}} = \frac{2e^{f(n)}\cosh g(n)+n^2-1+(n-1)e}{2e^{-f(n)}\cosh g(n)+n^2-1+(n-1)e^{-1}}$$

which for large n behaves increasingly like e^{1-n} and hence $b_e \to 0$ as $n \to \infty$. □

For our second family, we consider networks where the path graph P_N is connected by a single edge to K_n, as illustrated in Fig. 2. If we let N grow much faster than n, then we can show a very different behaviour from the previous one. It has adjacency matrix

$$\mathbf{A}_T = \begin{bmatrix} \mathbf{K}_n & \mathbf{C} \\ \mathbf{C}^\top & \mathbf{P}_N \end{bmatrix},$$

where $\mathbf{C}$ is a matrix which has a single non-zero entry in its bottom lefthand corner, and $\mathbf{P}_N$ is the adjacency matrix of P_N. The proof relies on the following conjecture for which we only have numerical evidence.

CONJECTURE 4.2 For sufficiently large n, the eigenvalues of $\mathbf{A}_T$ ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$) and those of

$$\begin{bmatrix} \mathbf{K}_n & \mathbf{O} \\ \mathbf{O}^\top & \mathbf{P}_N \end{bmatrix}$$

($\mu_1 \geq \mu_2 \geq \cdots \geq \mu_n$) are such that for all i

$$|\lambda_i - \mu_i| = O(n^{-2})$$

for $N \geq n$.

THEOREM 4.3 For the graphs G_n with adjacency matrices $\mathbf{A}_T$, we can choose $N = N(n)$ such that as $n \to \infty$, $b_\lambda \to 0$, $b_C \to 1$ and $b_e \to 1$.

Proof. Let G_n' be the graph with disjoint components K_n and P_N. By our conjecture, this has similar eigenvalues to G_n and standard results give

$$\sigma(G_n') = \sigma(K_n) \cup \sigma(P_N) = \{[n-1]^1, [-1]^{n-1}\} \cup \{2\cos\frac{\pi j}{N+2}, j = 1, \ldots, N+1\}.$$

Turning to the measures, we observe that by removing half the edges from the complete graph we have a bipartite graph and hence $m_d \leq n(n-1)/4$ and so long as $n^2/N = o(n)$ we will have $b_C \to 1$ as $n \to \infty$.

Next note that $\lambda_1(G'(n)) = n-1$ while $\lambda_{n+N}(G'_n) \geq -2$ and hence $b_\lambda \to 0$ as $n \to \infty$.

Finally,

$$b_e = \frac{(n-1)e + e^{1-n} + S(N)}{(n-1)e + e^{n-1} + S(N)},$$

where

$$S(N) = \sum_{j=1}^{N+1} \exp(2\cos\frac{\pi j}{N+2}) \approx \int_1^{N+1} \exp(2\cos\frac{\pi x}{N+2})dx = \frac{N+2}{\pi}\int_{1/(N+2)}^{\pi-\pi/(N+2)} e^{2\cos x}dx$$

and hence for large N, we can use to high accuracy the approximation

$$S(N) = (N+2)J_0(2i),$$

where $J_0(x)$ is a Bessel function of the first kind. Note that $J_0(2i) \approx 2.28$. We end up with the approximation for large n and N

$$b_e \approx \frac{J_0(2i)N}{e^{n-1} + J_0(2i)N},$$

and if N is sufficiently bigger than n (e.g. $N > e^{2n}$), we see that the right handside of this expression approaches 1 as $n \to \infty$. □

We observe that the difference in size of the path graph and the complete graph in this example is rather extreme. To illustrate the result, we have computed b_λ and b_e explicitly for $n = 2, \ldots, 8$ with $N = [e^n]$ (with e^{2n} we have a much smaller range of n with which to work with computationally). In Fig. 3, we record the values of b_e and b_λ and show the effectiveness of our bounds.

Thus, the choice of measure we make to assess near bipartivity can in theory have a sizeable impact. Observe that in the class of graphs considered in Theorem 4.1, a single edge is enough to introduce a large number of walks of odd length, hence the rapid reduction in b_e, but for the class in Theorem 4.3, the lack of bipartivity is effectively 'quarantined' by the lack of connectivity between the two parts of the graph but the large eigenvalue of K_n distorts the measure b_λ. We could describe the loss of bipartivity in Theorem 4.1 as global while that in Theorem 4.3 is local. A similar observation has been made about the potential for the Watts–Strogatz and Newman clustering coefficient measures to give very different perspectives on global properties of a network [12] while in practice they seem to be strongly correlated for real world networks. In experiments in Section 6, we look at how bipartivity measures behave in practice.

5. Algorithms

An obvious goal when looking at nearly bipartite networks is to find a way of partitioning the graph so that there are as few frustrated edges as possible. As we have previously noted, finding the optimal solution is an NP-hard problem. But finding such partitions either exactly or approximately is

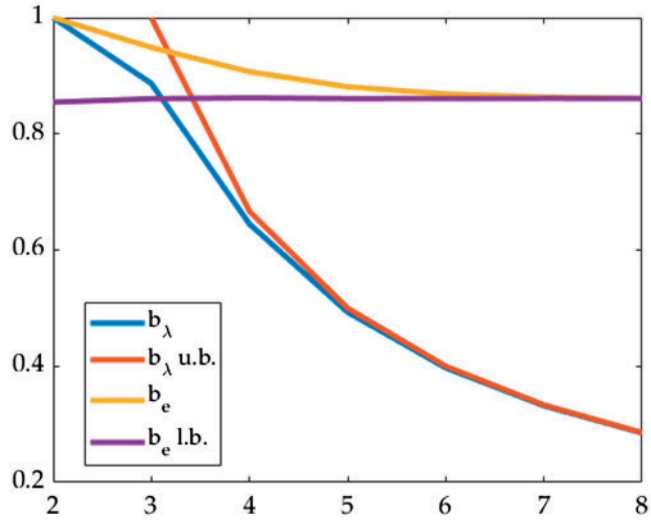


FIG. 3. b_λ and b_e.

clearly related to finding community structures and is often described as finding anti-communities. A number of methods have been proposed to find good bipartitions, and we summarize some of these below.

As observed in [13], community structures in the complement graph correspond to anti-communities in the original graph and so any community detection algorithm can be employed. This approach can be costly, since it may involve working with a dense graph but provides us with a starting point.

As a number of community detection algorithms look to maximize modularity, one could also look to find good anti-communities by minimizing modularity. In [13], Newman proposes using the eigenvector associated with the smallest eigenvalue of the modularity matrix $\mathbf{B}$ defined in (2.1). Newman goes further and suggests a method for finding multiple anti-communities but as our focus solely is on bipartitions we will not consider this further.

Estrada [12] proposes using the so-called anti-communicability matrix $e^{-\mathbf{A}}$ to find an approximate bipartition. Signs of the entries of this matrix indicate whether there are more even or odd walks between pair of nodes and by defining a graph which has an edge between nodes i and j if and only if the (i,j) entry of $e^{-\mathbf{A}}$ is positive we should cluster together nodes that belong in the same partition. Ideally these clusters would form cliques, but in any case the bipartition can be found using a standard community detection algorithm on the new graph.

One can use the connection between bipartivity and the spectral properties of adjacency matrices and Laplacians described in Theorem 2.2 to come up with algorithms for finding approximate bipartitions in a way akin to the Fiedler vector for community detection. For example, we can use the signs of the elements of the eigenvector associated with the most negative eigenvalue of the adjacency matrix to bipartition, an idea that has proved popular in areas such as bioinformatics [14]. Similar methods can be derived from the following theorem.

THEOREM 5.1 Suppose G is a connected bipartite graph with partition $P = \{V_1, V_2\}$ and adjacency matrix $\mathbf{A}$. Labels V_1 and V_2 so that $V_1 \cup V_2 = \{1, \ldots, n\}$. Let $\mathbf{x}$ be one of the following eigenvectors.

1. The eigenvector associated with the most negative eigenvalue of $\mathbf{A}$.
2. The eigenvector associated with the smallest eigenvalue of $\mathbf{Q}$.
3. The eigenvector associated with the largest eigenvalue of $\mathbf{N}$.
4. The eigenvector associated with the Fiedler vector of the complement graph.
5. The eigenvector associated with the largest eigenvalue of $\mathbf{L}$.

Then, nodes i and j are in the same partition if and only if $x_i x_j > 0$.

The results 1–3 can be inferred from [15] and 4 is a simple corollary of Fiedler's results for the Laplacian of graphs [16]. Note that we did not give a measure of bipartivity based on $\mathbf{L}$. Nevertheless it can still be used to find a bipartition as we now show.

Proof of 5. Suppose $V_1 = \{1, \ldots, r\}$, $V_2 = \{r+1, \ldots, n\}$. We can write a signed incidence matrix for G as $\mathbf{C} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \end{bmatrix}$ where $\mathbf{C}_1 \in \mathbb{R}^{m \times r}$, $\mathbf{C}_2 \in \mathbb{R}^{m \times (n-r)}$ with $\mathbf{C}_1 \geq 0$ and $\mathbf{C}_2 \leq 0$ such that $\mathbf{L} = \mathbf{C}^\top \mathbf{C}$.

Consider $\mathbf{C}\mathbf{C}^\top = \mathbf{C}_1\mathbf{C}_1^\top + \mathbf{C}_2\mathbf{C}_2^\top \geq 0$ with dominant eigenvalue μ and corresponding eigenvector $\mathbf{y}$. By the Perron–Frobenius theorem $\mathbf{y} > 0$. But $\mathbf{x} = \mathbf{C}^\top \mathbf{y}$ is the eigenvector of the dominant eigenvalue of $\mathbf{L}$ because $(\mathbf{C}^\top \mathbf{C})(\mathbf{C}^\top \mathbf{y}) = \mu \mathbf{C}^\top \mathbf{y}$, and $\mathbf{C}\mathbf{C}^\top$ and $\mathbf{C}^\top \mathbf{C}$ share the same non-zero eigenvalues. So

$$\mathbf{x} = \begin{bmatrix} \mathbf{C}_1^\top \mathbf{y} \\ \mathbf{C}_2^\top \mathbf{y} \end{bmatrix},$$

and the first r entries of $\mathbf{x}$ are positive and the next $n - r$ are negative, as required. □

So, we have a variety of eigenvectors to choose from to form an approximate bipartition for nearly bipartite matrices. Computationally, there is very little to choose between finding individual eigenvectors of $\mathbf{A}$, $\mathbf{L}$, $\mathbf{N}$ and $\mathbf{Q}$. We test their performance in the next section.

Other methods related to spectral properties of bipartite graphs have been described in the literature. For example, the authors of [10] use the symmetry of the spectrum to find a bipartition, but it requires computation of the complete spectral decomposition of the adjacency matrix. And in [17], the authors extend the idea of anti-communicability to define a so-called anti-communicability angle between nodes. This works with the matrix $e^{-\mathbf{A}}$ after a normalization of the entries to embed the graph into a Euclidean space. The authors propose using k-means to bipartition the nodes (and the idea can be extended to multi-way partitions). In the next section, we compare the methods in [10, 17] to methods based on a single eigenvector.

Whatever method is used, once we have computed an approximate bipartition we can try and do better with local improvement. This is a well-established method in community detection [18], where nodes can be swapped between partitions in order to reduce the number of inter-community edges. The idea is simplified when attempting to find a better bipartition since we do not need to swap nodes. We simply need to move individual nodes across partitions if they reduce the number of frustrated edges, or some other measure of bipartivity. We have implemented this by using local improvement to reduce modularity. We do this as modularity is not directly tied to the spectral properties we try and exploit in our algorithms, and as we now show, it can easily be updated as we swap nodes and so is cheap to implement.

Suppose $P = \{V_1, V_2\}$, where $V_1 = \{1, \ldots, r\}$ and $V_2 = \{r+1, \ldots, n\}$, and we want to swap node i from V_1 to V_2. We have

$$Q_{\text{old}} = \frac{\mathbf{s}^\top \mathbf{B}\mathbf{s}}{4m} \text{ and } Q_{\text{new}} = \frac{\bar{\mathbf{s}}^\top \mathbf{B}\bar{\mathbf{s}}}{4m},$$

where $\mathbf{s}$ and $\bar{\mathbf{s}}$ are the indicator vectors for the old and new partitions[2], so $\bar{\mathbf{s}} = \mathbf{s} - 2\mathbf{e}_i$.

A simple calculation gives

$$Q_{\text{new}} = \frac{1}{4m}(\mathbf{s} - 2\mathbf{e}_i)^\top \mathbf{B}(\mathbf{s} - 2\mathbf{e}_i) = Q_{\text{old}} + \frac{1}{m}(\mathbf{e}_i^\top \mathbf{B}\mathbf{e}_i - \mathbf{s}^\top \mathbf{B}\mathbf{e}_i).$$

Now, $\mathbf{e}_i^\top B\mathbf{e}_i = a_{ii} - k_i^2/(2m) = -k_i^2/(2m)$ and since

$$\mathbf{B}\mathbf{1}_n = \mathbf{A}\mathbf{1}_n - \frac{\mathbf{k}\mathbf{k}^\top}{2m}\mathbf{1}^n = \mathbf{k} - \frac{2m}{2m}\mathbf{k} = 0,$$

we have

$$-\frac{1}{m}\mathbf{s}^\top \mathbf{B}\mathbf{e}_1 = \frac{2}{m}\begin{bmatrix}\mathbf{0}\\ \mathbf{1}_{n-r}\end{bmatrix}^\top \mathbf{B}\mathbf{e}_1 = \frac{2}{m}\mathbf{e}_i^\top \left[\mathbf{A} - \frac{\mathbf{k}\mathbf{k}^\top}{2m}\right]\begin{bmatrix}\mathbf{0}\\ \mathbf{1}_{n-r}\end{bmatrix} = \frac{2}{m}k_1^{V_2} - \frac{k_1\mathbf{k}^\top}{m^2}\begin{bmatrix}\mathbf{0}\\ \mathbf{1}_{n-r}\end{bmatrix} = \frac{2}{m}k_i^{V_2} - \frac{k_i k_{V_2}}{m^2}.$$

And so the change in modularity is given by

$$\Delta Q = \frac{-k_i^2}{2m^2} - \frac{k_i k_{V_2}}{m^2} + \frac{2k_i^{V_2}}{m},$$

where $k_i^{V_2}$ is the number of edges from node i that end in V_2 and k_{V_2} is the sum of the degrees of the nodes in V_2.

The equivalent change can be calculated for every node very quickly and we swap the node which gives the minimum value of ΔQ (so long as it is negative). We can repeat this process until we reduce Q no further.

6. Experiments

In this section, we detail a number of experiments we have conducted on both randomly generated and real-world networks to investigate the links between measures and the ability of the spectral algorithms to find approximate bipartitions effectively.

6.1 *Perturbing random graphs*

In this experiment, we add edges randomly and sequentially to randomly generated bipartite graphs. The number of edges added is an (increasingly crude) upper bound for the number of frustrated edges. Each time we add an edge, we measure b_λ, b_e, b_Q and b_N. The results can be seen in Fig. 4. On the left are

[2] $\mathbf{s}$ is defined after (2.1).

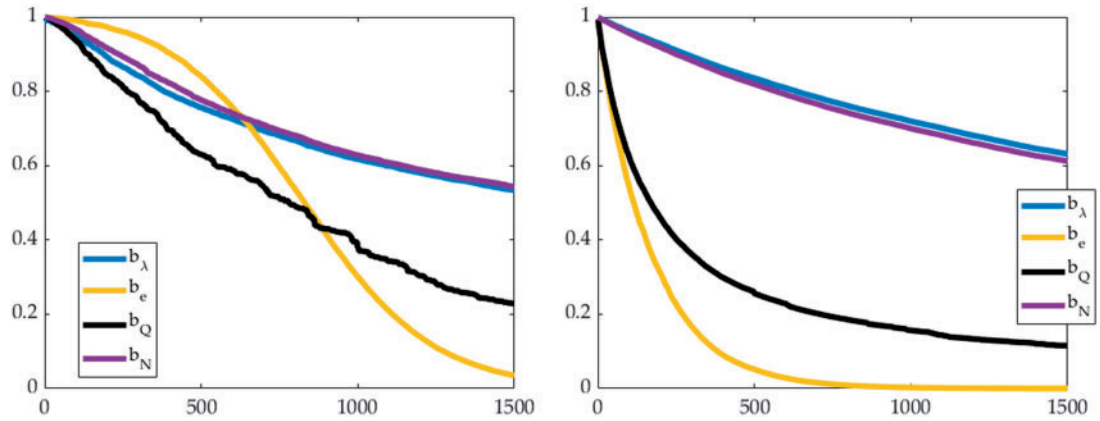


FIG. 4. Variation of bipartivity measures for random graphs.

the result when edges were added to trees randomly generated using Prüfer sequences [19] and so have a degree distribution that follows a Poisson model. The graph shows the results averaged over 20 trees each with 300 nodes but the trends were repeated for trees of other size and generated in other ways (e.g. exponential distribution). On the right, the initial networks were generated using an adjacency matrix of the form

$$\begin{bmatrix} \mathbf{O} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{O} \end{bmatrix},$$

where the (binary) entries of $\mathbf{B}$ are chosen at random according to some parameter p so that each has probability p of equalling 1. Thus, these matrices are generated in a very similar way to the Erdős–Rényi (ER) model. Edges were assigned randomly to the diagonal blocks of the matrix. Again the results were qualitatively similar for a wide range of values of p and dimension. We show the results averaged over 20 trials for $\mathbf{B} \in \mathbb{R}^{200\times 100}$ with $p = 0.1$.

Similar trends can be seen in other random models: the measures b_λ and b_N behave in a similar way, decaying relatively slowly compared to b_e and b_Q.

6.2 *Real-world graphs*

We now look at how the measures behave when we work with real-world networks. Given a non-bipartite graph we can remove edges to make it bipartite, then reintroduce these edges and track the measures. In this instance, to form a bipartition we simply split the nodes according to the signs of the elements of the eigenvector associated with the most negative eigenvalue. The trends here vary slightly more than with the random models. In Fig. 5, we present typical results. In the left hand picture, the graph is that of a food web in a waterway called Stony Stream with 112 nodes and 830 edges described in [20]. In the middle, the graph is an electronic circuit with 512 nodes and 819 edges described in [21]. And on the right, the graph of 994 nodes and 3640 edges is constructed from synonyms in Roget's thesaurus as described in [22]. More extensive results for over 30 real-world networks can be found in [23]. We note that typically b_e decays faster than the other measures, whose behaviour is more variable.

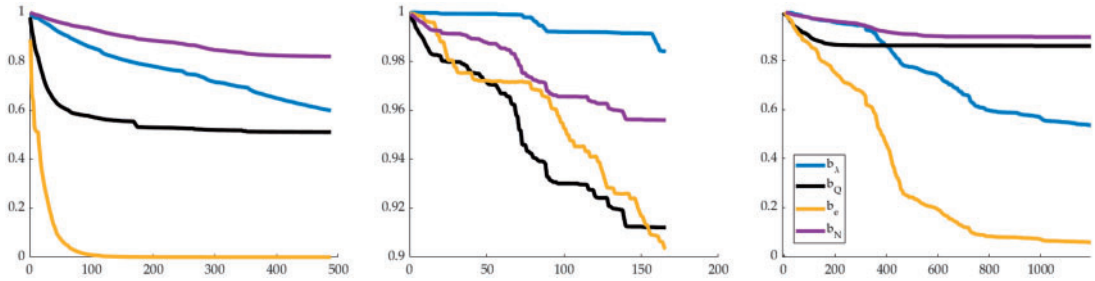
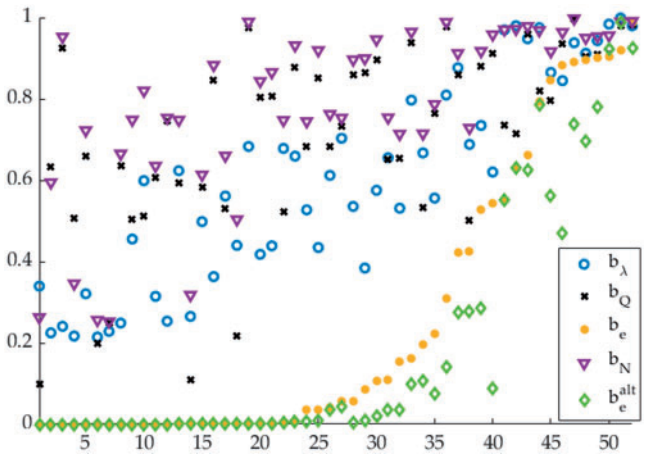


FIG. 5. Variation of bipartivity measures for real-world graphs.



	b_λ	b_N	b_e	b_Q	b_e^{alt}
b_λ	1	0.57	0.83	0.70	0.81
b_N		1	0.52	0.92	0.46
b_e			1	0.55	0.95
b_Q				1	0.51
b_e^{alt}					1

FIG. 6. Comparison of bipartivity measures for real-world graphs.

Next, we look at a variety of measures for 52 simple connected real-world networks curated from various on-line data repositories. They vary in size from 28 to 1586 nodes and from 67 to 18812 edges. The networks were not selected for any particular properties related to bipartivity.[3]. We have simply computed the four measures considered in our other experiments for each of these networks. The results can be seen in Fig. 6, where the networks are ordered by the size of b_e. We have used the additional measure $b_e^{\text{alt}} = e^{\lambda_1}/e^{-\lambda_l}$ which, as mentioned in Section 3, is a computationally cheaper approximation of b_e. To the right of the graph, we have given the correlations between pairs of measures. Clearly, the two measures that use the exponential are the most closely correlated, which suggests that if cost is a factor then we can use b_e^{alt} instead of b_e. Kunegis carries out a similar experiment on some comparable measures in [3] and recommends b_λ as it has a more uniform distribution over the range of networks. Our other measures are not exactly the same as his and seem to be more strongly correlated but the trends we

[3] Details about the networks, as well as sources to some of the original references and MATLAB code to reproduce the figures can be found at personal.strath.ac.uk/p.a.knight/Bipartivity.zip

see are similar. To gain more insight, it is worthwhile comparing the measures of bipartivity against the results of using the algorithms described in Section 5.

6.3 *Finding partitions*

We report the results of experiments to test various algorithms to find approximate bipartitions on the same 52 real-world graphs described above. We have tested eight algorithms, five derived from Theorem 5.1, and three alternatives described in Section 5, as follows.

1. Partition using the signs of the eigenvector associated with the most negative eigenvalue of **A**.
2. Partition using the signs of the eigenvector associated with the smallest eigenvalue of **Q**.
3. Partition using the signs of the eigenvector associated with the largest eigenvalue of **N**.
4. Partition using the signs of the eigenvector associated with the Fiedler vector of the complement graph.
5. Partition using the signs of the eigenvector associated with the largest eigenvalue of **L**.
6. Partition using the signs of the eigenvector associated with the most negative eigenvalue of the modularity matrix[4] as proposed by Newman [13].
7. Partition using the anti-communicability angle as described in [17].
8. Partition using Estrada's approach [12] with the communicability matrix.
9. Partition using the approach of Concas *et al.* [10].

For each matrix and each algorithm, we subsequently applied local improvement. We computed modularity in all cases and the results are illustrated in Fig. 7. Each circle indicates the size of modularity for a particular method on a particular matrix. Each row contains the results for a particular matrix, each column a particular method. The right and left blocks record the values with and without local improvement, respectively. Gaps in rows correspond to partitions with positive modularity, which is a crude indication of failure.

As is to be expected, different matrices show different levels of bipartivity. The results have been ordered by the modularity of the partition using, Method 3. We can see that there is a fair amount of variance in some of the rows in the left block but this is ameliorated (and modularity usually reduced somewhat) when we use local improvement. To distinguish between the methods, we look at the average modularity of the partitions they produce without and with local improvement. The results are as follows.

	1	2	3	4	5	6	7	8	9
Without	−0.222	−0.215	−0.226	−0.151	−0.151	−0.218	−0.144	−0.189	−0.082
With	−0.265	−0.263	−0.265	−0.256	−0.256	−0.264	−0.263	−0.264	−0.251

We see that in both cases, Method 3 does slightly better than any other and therefore for best performance we recommend using the normalized Laplacian along with local improvement to find a good approximate bipartition efficiently.

[4] **B** in (2.1).

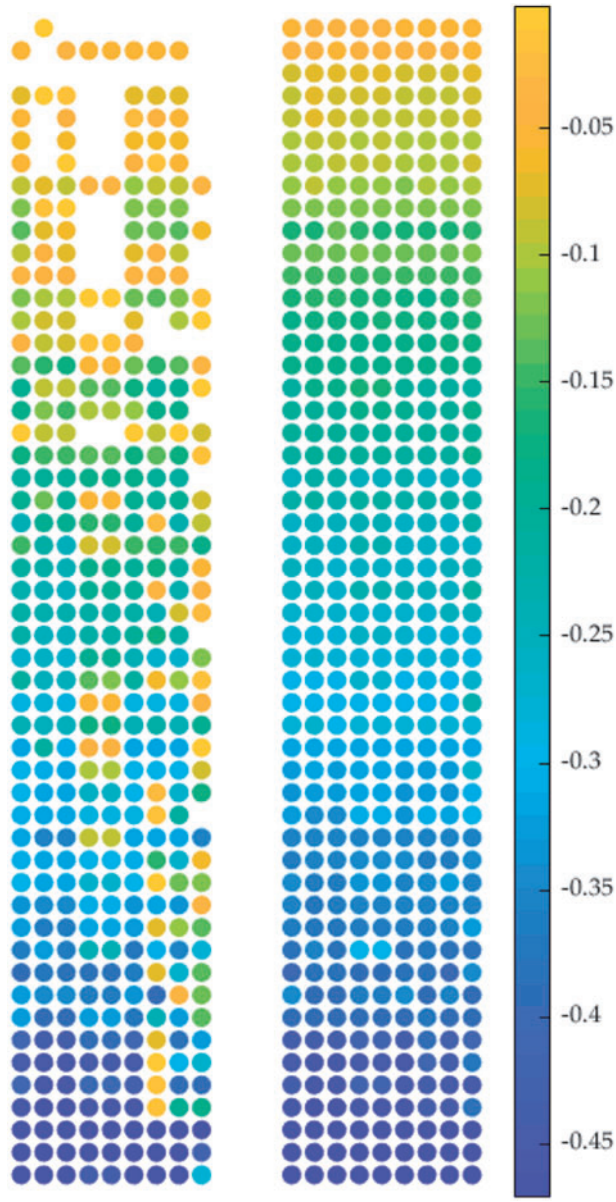


FIG. 7. Modularity of approximate bipartitions for real-world graphs.

We have used modularity as a proxy for the number of frustrated edges. We cannot use it as an *a priori* measure of bipartivity as we need the partition first. As a final experiment, we have compared the measures of bipartivity against the best partitions in terms of modularity score for each of the real-world matrices. Reordering the graph in the left of Fig. 6 according to modularity score gives Fig. 8 (we have omitted b_e^{alt} for clarity). It appears from the picture that b_λ follows the trend of modularity, and this is

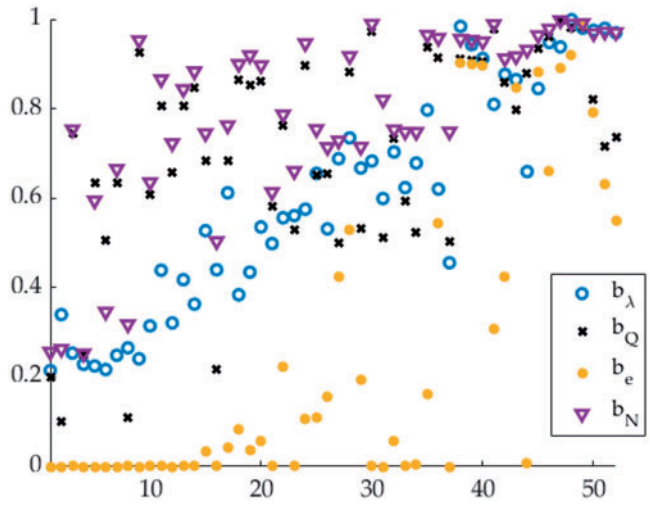


FIG. 8. Modularity versus bipartivity measures for real-world graphs.

confirmed by measuring the correlation between the measures and Q. For b_λ, it is -0.94 while for b_Q it is -0.58, for b_N it is -0.73, and for b_e it is -0.76 (negative correlation being what we would expect as modularity is minimized by bipartivity while our measures are maximized).

Finally, we investigate a near bipartite graph with a 'ground truth'. This is the graph of the connections between 60 commonly occurring adjectives and 60 commonly occurring nouns in the novel *David Copperfield* by Charles Dickens, introduced in [24]. The words are nodes in the graph and edges link words that appear adjacent at any point, and a degree of bipartivity is assumed as nouns should generally appear next to adjectives. We tested the nine algorithms listed above on the largest connected component of the resulting graph (112 nodes). The results are given in Fig. 9. The noughts and crosses represent the number of words that are misclassified by attempting the best bipartition with the algorithms (with and without local improvement), and the diamonds and triangles count the number of intra-partition edges. the performance of the algorithms is similar as in our previous test. Note that although local improvement always successfully reduces the number of frustrated edges it sometimes slightly increases the number of misclassifications, but given that the ground truth is not perfectly aligned with bipartivity (since many words are both adjectives and nouns) this is not unexpected. In terms of misclassifications, the minimum achieved (13 when using an eigenvector of $\mathbf{Q}$) matches the performance of the algorithm in [24] at much reduced cost.

7. Conclusions

Since bipartivity has so many equivalent characterizations there are many ways one can try and measure nearness to the property. We have restricted our analysis to spectral measures and algorithms, but this still gives us plenty to compare. Measures can be divided into those which use a purely algebraic characterization of bipartivity (in terms of spectral properties) and those which use the existence of odd cycles as a more direct manifestation of a lack of bipartivity. In experiments, it seems that both approaches give insightful results. In terms of performance versus cost, we recommend $b_{\mathbf{N}}$ since its cost is essentially just

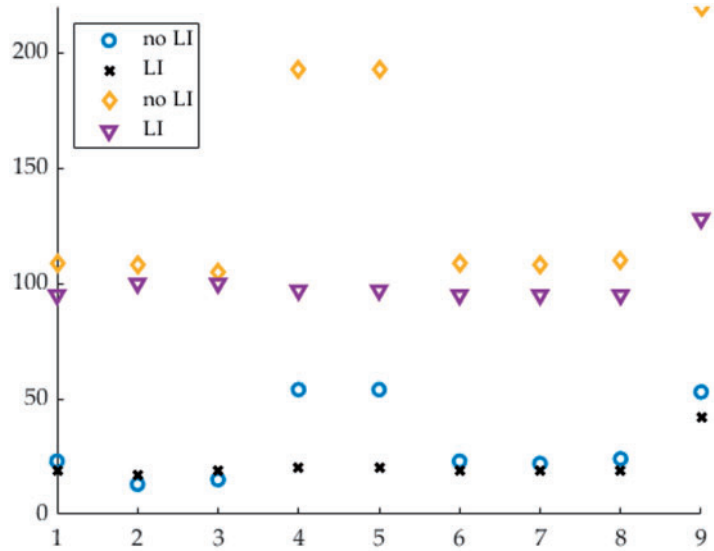


FIG. 9. Frustrated edges and misclassifcations in the Dickens graph.

that of finding the largest eigenvalue of a symmetric matrix for which an upperbound is already available and thus is generally very cheap to compute for even very large networks.

Overall it seems that measures and algorithms based on the adjacency matrix and the normalized Laplacian seem to be closely related to each other, and our results on minimizing modularity suggest that these are well correlated to the presence of frustrated edges. Our experiments with algorithms for finding a good partition into anti-communities show that simple spectral algorithms, based on the analysis of a single eigenvector, can perform as well (or better) than some of the more complex techniques that have been proposed in the literature, and that a little bit of post-processing with local improvement can really pay dividends. These algorithms can also can also give a relative cheap approximation of m_d, the number of frustrated edges, since we can get an upper bound by counting the intra-community edges in the bipartition we calculate for the cost of one eigenvector computation.

Our theoretical analysis shows that there is the possibility that measures based on very similar spectral ideas can diverge markedly. In particular, measures that look at the discrepancy between the number of even and odd walks between nodes seem to decay very sharply. It may be that we can exploit this to get a more nuanced view of the approximate bipartivity by using more than one measure on the same matrix. In Fig. 10, we show the adjacency matrices of two of our real-world graphs reordered to minimize modularity (non-zeros are shaded). In the left hand graph, $b_N = 0.97$ while $b_e = 0.55$ whereas in the right, $b_N = 0.75$ and $b_e = 5 \times 10^{-8}$. So, while b_N is large in both cases, the same is not true of b_e. We note that the reordering in the right-hand side shows that the graph has a classic core-periphery structure. Whether this characterization holds true more generally is a topic for future research.

Finally, a comment on the fact that we have stuck to spectral measures in our analysis. A well known limitation of bipartivity measures that rely on the spectrum of the graph is that b_c can differ for cospectral graphs, as illustrated in Fig. 11. Although we do not anticipate this being much of a problem in practice, one could always combine b_λ with b_Q or b_N. We are unaware of any graphs which are cospectral for adjacency matrices and Laplacians which have a different number of frustrated edges. We note that for

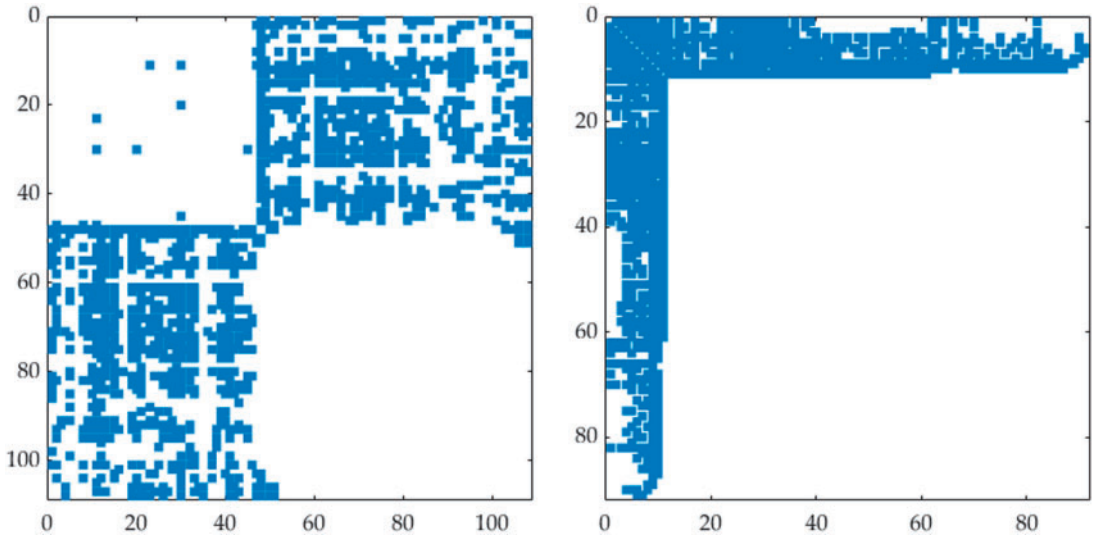


FIG. 10. Approximate bipartivity in two real-world graphs.

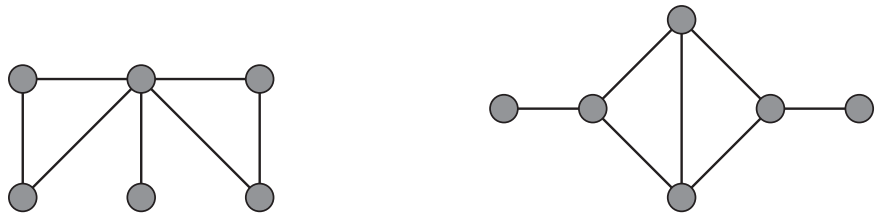


FIG. 11. Cospectral graphs, one with two frustrated edges (left) the other with one.

the graphs in Fig. 11 both b_Q and b_N judge the right-hand graph to be closer to being bipartite while our spectral algorithms all succeed in finding approximate bipartitions which require the removal of two edges for the left-hand graph and one for the right.

REFERENCES

1. ESTRADA, E. & RODRÍGUEZ-VELÁZQUEZ, J. A. (2005) Spectral measures of bipartivity in complex networks. *Phys. Rev. E*, **72**, 046105.
2. HOLME, P., LILJEROS, F., EDLING, C. R. & KIM, B. J. (2003) Network bipartivity. *Phys. Rev. E*, **68**, 056107.
3. KUNEGIS, J. (2015) Exploiting the structure of bipartite graphs for algebraic and spectral graph theory applications. *Internet Math.*, **11**, 201–321.
4. DOŠLIĆ, T. & VUKIČEVIĆ, D. (2007) Computing the bipartite edge frustration of fullerene graphs. *Discrete Appl. Math.*, **155**, 1294–1301.
5. ESTRADA, E. & GÓMEZ-GARDEÑES, J. (2016) Network bipartivity and the transportation efficiency of European passenger airlines. *Physica D*, **323**, 57–63.
6. BRANDES, U., DELLING, D., GAERTLER, M., GORKE, R., HOEFER, M., NIKOLOSKI, Z. & WAGNER, D. (2007) On modularity clustering. *IEEE Trans. Knowledge Data En.*, **20**, 172–188.

7. Das, K. C. (2010) On conjectures involving second largest signless Laplacian eigenvalue of graphs. *Linear Algebra Appl.*, **432**, 3018–3029.
8. Kirkland, S. & Paul, D. (2011) Bipartite subgraphs and the signless Laplacian matrix. *Appl. Anal. Discrete Math.*, **5**, 1–13.
9. Estrada, E. & Knight, P. A. (2015) *A First Course in Network Theory*. Oxford: Oxford University Press.
10. Concas, A., Noschese, S., Reichel, L. & Rodriguez, G. (2020) A spectral method for bipartizing a network and detecting a large anti-community. *J. Comput. Appl. Math.*, **373**, 112306.
11. Van Mieghem, P. (2010) *Graph Spectra for Complex Networks*. Cambridge: Cambridge University Press.
12. Estrada, E. (2012) *The Structure of Complex Networks: Theory and Applications*. Oxford: Oxford University Press.
13. Newman, M. E. J. (2006) Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, **74**, 036104.
14. Morrison, J. L., Breitling, R., Higham, D. J. & Gilbert, D. R. (2006) A lock-and-key model for protein–protein interactions. *Bioinformatics*, **22**, 2012–2019.
15. Paul, D. & Stevanović, D. (2019) Eigenvector-based identification of bipartite subgraphs. *Discrete Appl. Math.*, **269**, 146–158.
16. Fiedler, M. (1975) A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. *Czech. Math. J.*, **25**, 619–633.
17. Gómez-Gardeñes, J. & Estrada, E. (2021) Network bipartitioning in the anti-communicability Euclidean space. *AIMS Math.*, **6**, 1153–1174.
18. Kernighan, B. W. & Lin, S. (1970) An efficient heuristic procedure for partitioning graphs. *Bell Sys. Techn. J.*, **49**, 291–307.
19. Prüfer, H. (1918) Neuer Beweis eines Satzes über Permutationen. *Arch. Math. Phys.*, **27**, 742–744.
20. Thompson, R. M. & Mcintosh, A. R. (1998) Disturbance, resource supply, and food-web architecture in streams. *Ecol. Lett.*, **1**, 200–209.
21. Brglez, F., Bryan, D. & Kozminski, K. (1989) Combinational profiles of sequential benchmark circuits. *IEEE International Symposium on Circuits and Systems* (ISCAS), Portland, OR, USA: IEEE, pp. 1929–1934.
22. Knuth, D. E. (1993) *The Stanford GraphBase: A Platform for Combinatorial Computing*. New York: ACM Press.
23. Aleidan, A. (2023) Numerical analysis and network theory. PhD Thesis, University of Strathclyde.
24. Newman, M. E. J. & Leicht, E. A. (2007) Mixture models and exploratory analysis in networks. *Proc. Natl. Acad. Sci.*, USA, **104**, 9564–9569.