NETWORK SAMPLING ALGORITHMS AND APPLICATIONS

M. D. LAMAR AND R. K. KINCAID

1. Introduction

Networks appear throughout the sciences, forming a common thread linking research activities in many fields, such as sociology, biology, chemistry, engineering, marketing, and mathematics. For example, they are used in ecology to represent food webs and in engineering and computer science to design high quality internet router connections. Depending on the application, one network structural property may be more important than another. The structural properties of networks (e.g. degree distribution, clustering coefficient, assortativity) are usually characterized in terms of invariants [8], which are functions on networks that do not depend on the labeling of the nodes. In this chapter we focus on network invariants that are quantitative, that is, they can be characterized as network measures. An increasingly important application area is how network invariants affect the dynamics of a process on the network (e.g. respiration, current, traffic) [36]. In order to study the potential effect of incremental changes in network invariants on network dynamics, one or more network invariants are held constant, thereby creating a family of networks. In this chapter the degree distribution of a network is held constant whilst other network invariants are examined. In particular, we examine the effects of assortativity, the Randić index and eigenvalues of the Laplacian on network dynamics.

2. Notations and Definitions

We use the terms network and graph interchangeably. We assume the reader to have a knowledge of graph theory (see, e.g., [54]). Let a graph $G = (V, E)$ be given where V is the set of nodes and E the set of edges. For a directed graph, we use the similar notation $G = (V, A)$, where A is the set of arcs. We specify, when necessary, whether G is directed or not. For undirected graphs, we use the notation d_i to denote the *degree* of the node i, i.e., the number of edges incident to i. For directed graphs, we use the notation d_i^- and d_i^+ to denote the in- and out-degree of the node i, i.e., the number of edges with i as their head and tail, respectively. We use subscripts when there may be confusion on the graph in question, e.g., $d_i^-(G)$ or $d_i^+(G)$. When we discuss subsets of graphs, we use calligraphic font, e.g., $\mathcal{G}$. For undirected and directed graphs, we define $A(G)$ to be the node-node adjacency matrix and $D(G)$ to be the diagonal matrix with the degree sequence d (or d^+ in the directed case) along the main diagonal. We omit reference to G when the graph is clear from context.

For what follows, we assume that any set of graphs we use has a fixed number of nodes $n = |V|$ for all graphs in the set. We also assume that any set of graphs consists of either all directed graphs or all undirected graphs. We denote the set of undirected graphs by $\mathcal{U}$ and the set of directed graphs by $\mathcal{W}$.

For an undirected graph G, we define the *Laplacian* of G as the $n \times n$ matrix $L(G) = D(G) - A(G)$. The spectrum of $L(G)$ is well-studied. We simply note that $L(G)$ is symmetric and therefore has all real eigenvalues. As before, we omit reference to G when the graph meant is clear. There are multiple definitions for the Laplacian of a *directed* graph. The Laplacian of a strongly-connected directed graph G is defined as

$$L'(G) = D_\phi - \frac{D_\phi P + P^T D_\phi}{2} \tag{1}$$

where $P = D(G)^{-1}A(G)$ and D_ϕ is a diagonal matrix, with ϕ solving $\phi P = \phi$ subject to $||\phi||_1 = 1$ [10, 56]. Since L' is symmetric and has zero row sum, the eigenvalues of L' are real and nonnegative.

For an undirected graph G, the *degree sequence* of G is the non-increasing sequence of degrees of nodes of G, such as

$$d_1 \geq d_2 \geq \ldots \geq d_n,$$

where $d_i = \deg(i)$ with $i \in V$. For example, the degree sequence for the undirected graph in Figure 1 is

$$\{5, 3, 3, 2, 1, 1, 1, 1, 1\}.$$

We denote the eigenvalues of a graph G by $\lambda_i(G)$ where

$$0 = \lambda_1(G) \leq \lambda_2(G) \leq \lambda_3(G) \leq \ldots \leq \lambda_n(G)$$

are the n eigenvalues of L or L'. We define the *algebraic connectivity* [16] of G as λ_2. Our choice of L' above is motivated by the fact that λ_2 for L' has similar properties to λ_2 for L in the undirected case [56].

For an undirected graph G, we define the *generalized Randić index* [46, 34] $s(G)$ by

$$s(G) = \sum_{(i,j)\in E} d_i d_j. \tag{2}$$

Section 4.3 discusses the extension of $s(G)$ to directed graphs. Its relation to assortativity [43, 57] is discussed below.

Other common network invariants include assortativity, clustering coefficient, and average shortest path distance. Network assortativity is typically scaled between $[-1, 1]$ with values less than zero indicating that high degree nodes are more likely to be adjacent to low degree nodes (disassortativity). Assortativity, $r(G)$, can be shown to be equivalent to $s(G)$ using the equation

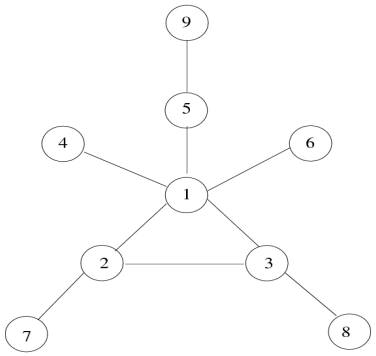


FIGURE 1. Undirected graph $G(V, E)$ with $|V| = 9$ and $|E| = 9$.

$$r(G) = \frac{[\sum_{(i,j)\in E} d_i d_j] - [\sum_{(i)\in V} \frac{1}{2} d_i^2]^2/|E|}{[\sum_{(i)\in V} \frac{1}{2} d_i^3] - [\sum_{(i)\in V} \frac{1}{2} d_i^2]^2/|E|} \tag{3}$$

where $|E|$ denotes the number of edges in the graph [32]. As you can see, $r(G)$ is linearly related to $s(G)$ since $s(G)$ is in the numerator and only the links and degrees scale it.

The clustering coefficient of a network computes the frequency of complete subgraphs on 3 nodes, a triangle (in social networks a triangle denotes that *a friend of your friend is my friend*). For each node i in the network compute

$$C_i = \frac{[\text{number of triangles in which node } i \text{ is incident}]}{[\text{number of three tuples of connected nodes centered on node } i\ ]}.$$

Then, the clustering coefficient $C = 1/n \cdot \Sigma_i C_i$. For the graph in Figure 1, $C = 1/9 \cdot (1/3 + 1/3 + 1/6) \approx 0.09$.

Given the shortest path distance matrix $\mathbf{D}$ of a graph, the average shortest path distance can be calculated by averaging the non-zero entries in $\mathbf{D}$. The longest shortest path for each node i is called the *eccentricity* of i. In Figure 1, the eccentricity of node 1 is 2, of node 2 is 3, and of node 7 is 4. The *diameter* of a network is the length of the longest shortest path (the maximum node eccentricity).

We make use of three types of undirected graphs, Erdős-Rényi, geometric and scale-free, whose structure depends on the parameters chosen.

Erdős-Rényi $G(n,p)$ Graphs. [20, 14] A number of nodes n and a probability of connection p are chosen. A random probability is generated for each possible edge. If the probability generated is less than p then an edge is added.

Geometric Graphs. A number of nodes n is chosen and placed on a unit square (or unit circle) at random. This gives each node i coordinates x, y. A radius r is chosen. An edge is placed betweeen nodes i and j if $(x_i - x_j)^2 + (y_i - y_j)^2 \leq r^2$ [52].

Power Law Graphs. A preferential attachment algorithm is used to create graphs whose degree sequences follow a power-law distribution. Following the convention in the literature we will refer to these graphs as "scale-free". A number of nodes n is chosen. New nodes are added and connected to existing nodes, based on a probability proportional to the current degree of the nodes, until n nodes are generated, making it more likely that a new node will be connected to a higher degree node [52]. The algorithm allows a minimum node degree to be specified.

3. Spectral properties and network dynamics

There are a number of well studied network invariants associated with the spectrum of a graph [8]. In particular, λ_2, λ_n, and $\frac{\lambda_n}{\lambda_2}$ have been shown to have a direct impact on a network's ability to synchronize flow activities at the nodes. In this section we investigate how the node (e.g., airport, neuron, oscillator) connectivity influences the flow (e.g. traffic, information, current) on a network. Of importance here is the *tunability* of a given network invariant. We realize tunability via optimization. Atay *et al.* [3, 2] provide definitions of node synchrony for networks. Instead of making direct use of these definitions we focus on algebraic connectivity, λ_2, a graph invariant that has been shown to correlate well (see [3, 4, 55]) with a network's capacity to synchronize. Intuitively, networks with small λ_2 are easier to *pull apart*. In particular, if $\lambda_2 = 0$, then the network is disconnected [16] and synchronization is impossible. Without more details regarding the flow on a network it is difficult to make definitive statements regarding synchronization. However, in general, the flow on a network is less likely to synchronize if λ_2 is small. The two leftmost networks in Figure 2 have identical degree distributions, but the network on the left is more weakly connected (e.g., the removal of a single edge can disconnect the network). Note that the two rightmost networks are identical to the two leftmost networks except for the addition of a leaf. Although both networks now have an edge connectivity of one (cutting one edge breaks the graph apart), λ_2 still reflects the higher *global* connectivity of the graphs.

For certain processes on networks (often described as a complex system) synchronization is an essential feature. For example, in mammals a small group of neurons (roughly 200) is responsible for generating a regular rhythmic output to motor cells that initiate a breath (see Section 5.1.1). Without synchronization of the neuronal output, breathing would be

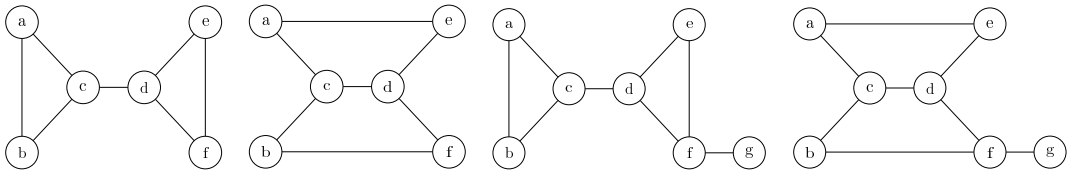


FIGURE 2. Left-to-right, $\lambda_2 = 0.44$, $\lambda_2 = 1.00$, $\lambda_2 = 0.34$, $\lambda_2 = 0.60$

ragged or not occur at all. Synchronization, as described here, leads to nodes (neurons) behaving in concert. In an air transportation setting such synchronization would be undesirable. Think of the airports as the neurons in our mammalian respiratory example. Inhaling means all planes land at all airports simultaneously. Exhaling means they depart together. The result is severe congestion. Thus, for this definition of synchronization, one would like an air transport network design to minimize synchronization.

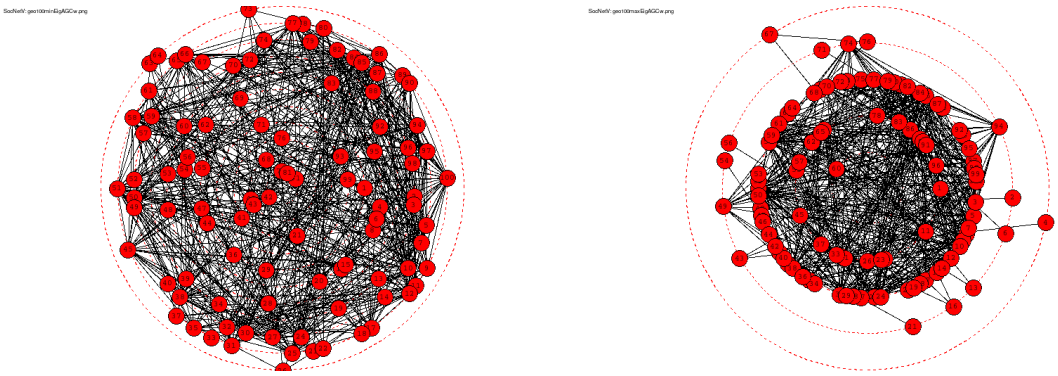
Figure 3 displays the result of optimizing λ_2 while holding the degree distribution fixed and $s(G)$ fixed. The graphs are generated by `socnetv`[1] and optimized with tabu search [27]. The position of each node in the plots is given with respect to the reciprocal of the eccentricity of each node i. The goal of the plots is to uncover qualitative differences between the graphs with small and large values of the second eigenvalue of the Laplacian.[2] Nodes with equal eccentricity values are plotted on the same (dashed line) circles. The circles with larger radii have larger eccentricity. Consequently, nodes near the center have shorter longest paths. The paired plots exhibit large qualitative differences in the eccentricity pattern.

Qualitatively, when λ_2 is small, the patterns are less organized, the eccentricity plot on the left in Figure 3 is more dispersed and consists of many rings of constant eccentricity. The eccentricity plot for the larger λ_2 (right half of Figure 3) is more organized, with fewer rings of constant eccentricity. Specifically, Figure 3(a) has 11 rings and Figure 3(b) has only 5 rings. The range of eccentricity values for the large λ_2 geometric graph, $[4, 8]$, dominates the range for the large λ_2 plot, $[26, 42]$. That is, the eccentricity pattern in Figure 3(b) is non-overlapping and interior to the one for Figure 3(a).

The diameter of the graph in Figure 3(a) is 42 while the graph diameter in Figure 3(b) is 8. For graphs with a fixed degree distribution and a fixed value of $s(G)$, this result – that λ_2 is inversely proportional to the eccentricity – appears to hold in general. We know of no theorem that proves this result but numerous computational tests support this claim. Moreover, the inverse relationship between λ_2 and the eccentricity does not hold if $s(G)$ is allowed to vary. (The interested reader is referred to [27] for further examples.)

[1]The source code and documentation can be found at `http://socnetv.sourceforge .net/`.

[2]We leave it to the reader to become acquainted the variety of measures and display features in `socnetv`. For the purposes of this exposition, we are interested only in the qualitative differences between the plots.



(a) $\lambda_2 = 0.009$, $e(V) = (26, 42)$, $c(G) = 0.426$ (b) $\lambda_2 = 0.314$, $e(V) = (4, 8)$, $c(G) = 0.297$

FIGURE 3. Geometric graphs: 100 nodes, $s(G) = 0.971$, fixed degree distribution

Several authors [3, 2, 13, 39, 44] have examined network dynamics as a function of network topology and have shown that different constrained topology optimization problems, such as maximizing synchronization or node proximity, lead to optimal topologies that, although not identical, share common features. Donetti et al. [13] call these optimal networks entangled. The result appears relevant to optimization of transport networks. Donetti et al. provide an illustrative example foreshadowing the topological significance of the spectral gap for networks dynamics. If a network consists of a number of disconnected subgraphs, its Laplacian is block-diagonal and the multiplicity of the trivial zero eigenvalue equals the number of disconnected subgraphs. Connecting the subgraphs weakly introduces small eigenvalues with nearly constant corresponding eigenvectors. This feature (small spectral gaps) provides a criterion for graph partitioning in well-known algorithms [42]. Intuitively, small λ_2 values imply the existence of well-defined modules that can be disconnected by cutting a small number of links, while large λ_2 values point to unstructured (entangled) graphs. Several authors, [13, 39, 4, 44], study synchronizability of diffusive processes on networks with identical nodes, considering a general dynamical process

$$\dot{x_i} = F(x_i) + \sigma \sum_{i=1}^{N} L_{ij} H(x_j), \quad i = 1, \ldots, N, \tag{4}$$

where x_i are dynamical variables, F is an evolution function, H is a coupling function, and σ is a coupling constant. Although diffusive processes are known to have synchronous states, the question is, under what conditions these states are stable. A linear stability analysis, [4], reveals that synchronized states are more stable for smaller λ_N/λ_2. Since the variability of the maximum eigenvalue is bounded [38], increasing stability of synchronized states amounts to maximizing the spectral gap λ_2. Other authors [3, 55] have used the spectral gap as an indicator of synchronization for discrete systems.

The normalized Laplacian, $\mathrm{L}' = D^{1/2} L D^{-1/2}$, and its eigenvalues $\{\lambda_i\}$ also play an important role, especially in the study of random walks, a subject relevant to propagation

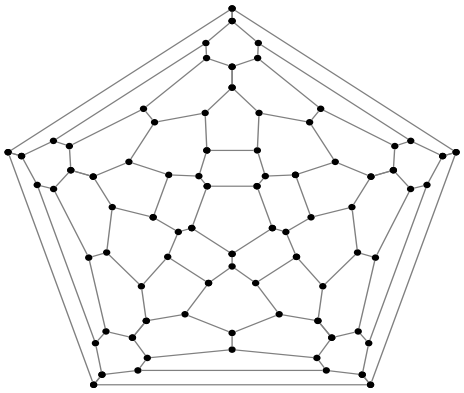


FIGURE 4. A Ramanujan or expander graph.

of traffic through networks. Large spectral gaps increase the rate at which random walks move and disseminate. A class of graphs with large spectral gaps, known as Ramanujan graphs (see Figure 4), is described by Donetti et al. [13]. These graphs are regular, have a vanishing clustering coefficient, a small average shortest path distance and a large girth (number of edges in the smallest cycle denotes the girth). Engineered systems do not typically fall into the class of Ramanujan graphs. For example, the nodes and node degree are clearly not identical for internet router networks or air transportation networks. However, changing the coupling constant in Equation (4) to σ/d_i, and thus normalizing the effect of the neighboring nodes (in turn, increasing the relevance to traffic networks), results in an optimal topology when the normalized spectral gap is maximized. These graphs are not characterized as nicely as the Ramanujan graphs. In particular, the networks are no longer regular and the degree distribution is not Possion. In [13] a plot of one instance of a graph with this degree distribution can be found. The plot is strikingly similar to one in [12] in which an optimization model was employed to construct the network. The optimization model employed in [12] minimizes a weighted graph distance that attempts to capture two conflicting objectives: avoidance of long paths (minimize diameter) and avoidance of heavy traffic (minimize node degree). Both [13] and [12] note that the degree distribution for the network apppears to decay faster than an exponential distribution and that the graph avoids construction of long paths. However, neither reference verifies the decay rate of the degree distribution.

4. Network sampling

Many applications require the ability to uniformly sample networks with constrained graph invariants [11, 35]. One of the most well-studied examples is the uniform sampling of networks with a fixed degree sequence. Other constraints can be added to this, including fixed assortativity or edge-degree correlation [27]. The well-known algorithm of Havel-Hakimi [22, 21] constructs networks from a degree sequence, albeit in a non-uniform manner. For

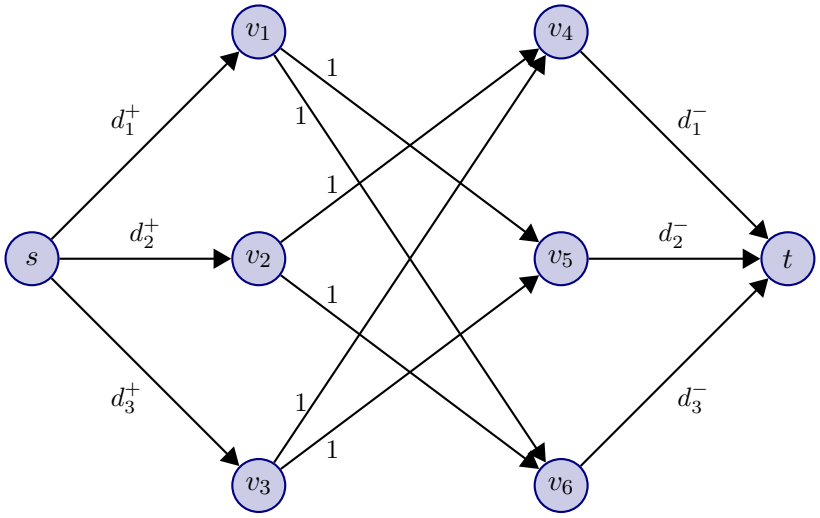


FIGURE 5. Example flow network used in the Ford-Fulkerson maximal flow algorithm to construct a network with degree sequence $\{(d_1^+, d_1^-), (d_2^+, d_2^-), (d_3^+, d_3^-)\}$.

the directed case, an analogous algorithm to Havel-Hakimi was developed by Kleitman and Wang [29].

Another algorithm to construct directed networks from fixed degree sequences uses the Ford-Fulkerson maximum flow algorithm [17]. To see this, suppose we have a directed degree sequence $\{(d_1^+, d_1^-), \ldots, (d_n^+, d_n^-)\}$. Create a directed flow network F with $2n + 2$ nodes $V = \{v_1, \ldots, v_{2n}, s, t\}$, where s is the source node and t is the target node. The arc set is given by

$$A = \{(s, v_i), (v_{n+i}, t)\}_{i=1}^{n} \cup \{(v_i, v_{n+j}) \mid i \neq j\}_{i,j=1}^{n}$$

with capacities

$$\begin{aligned} c(s, v_i) &= d_i^+, \quad i = 1, \ldots, n, \\ c(v_i, v_{n+j}) &= 1, \quad i, j = 1, \ldots, n \text{ with } i \neq j, \\ c(v_{n+i}, t) &= d_i^-, \quad i = 1, \ldots, n. \end{aligned}$$

Solving for a maximal flow using the Ford-Fulkerson algorithm will give a network realization of the degree sequence. An example flow network F for a degree sequence with three nodes is given in Figure 5.

The algorithms of Havel-Hakimi and Kleitman-Wang unfortunately do not construct networks from the set of all realizations with equal probability. In order to achieve a uniform sample from the space of realizations, more sophisticated algorithms must be used. There are two main classes of algorithms that are used in these circumstances. The first consists of importance sampling algorithms, where each network G in the sample space $\mathcal{G}$ or realizations has a positive probability $p_G > 0$ of being sampled, with the exact probability

p_G known. One can then get an unbiased estimator to a population measure $f(G)$ by performing a weighted sum over a set of samples, with each sample weight $w_G = 1/p_G$. There are modifications of the Havel-Hakimi and Kleitman-Wang algorithms that have been used with importance sampling [7, 19, 26].

The second method to achieve a uniform sample consists of using a simple algorithm like Havel-Hakimi or Kleitman-Wang to first construct a network realization, and then use edge or arc-switching techniques [11, 50]. For the directed case, an example of an arc-switch is shown here, where we move from two arcs $\{(i,j),(m,n)\} \subset A$ to $\{(i,n),(m,j)\} \subset A$.

Note that arc switches preserve the degree sequence. In these algorithms, since any two sample networks $G_1, G_2 \in \mathcal{G}$ are connected by a series of edge or arc-switches [50], we can then perform a Monte Carlo random walk on $\mathcal{G}$. Using different algorithmic modifications, such as Metropolis-Hastings, we can make the random walk's stationary distribution uniform [11]. The main drawback of these techniques is the lack of analytical measures of the mixing time, although some have been shown in certain cases [6].

4.1. **Degree distributions.** The most studied graph invariant is the degree distribution. Some of the most important types include binomial (Erdős-Rényi networks), power-law (scale-free networks), and poisson (geometric random networks).

There are many techniques to construct random networks with a prescribed degree distribution [20, 14, 45]. The technique we discuss in this chapter is the use of inversion methods to randomly sample a degree sequence from a specified distribution and then sample a network uniformly from this degree sequence. To randomly sample a degree sequence, we use the *probability integral transform* which states that if X is a continuous random variable with cumulative distribution given by F, then $U = F(X)$ is a random variable on $[0,1]$ with a uniform distribution. In theory, to generate a random number X from the distribution F one only needs to generate a uniform random variable U and then compute $X = F^{-1}(U)$. This is of limited use in general as it is computationally intractable to compute the inverse cumulative distribution (also known as the *quantile function*), except, for example, in the case of discrete distributions. As degree distributions are discrete, the inversion method can be translated into an algorithm as follows. Given a discrete random variable k denoting node-degree with the degree distribution $p_i \equiv P(k = i)$, we can compute the cumulative distribution $F_k = \sum_{i=1}^{k} p_i$. Now draw a uniform random variable U on the interval $[0,1]$ and choose k such that $F_{k-1} < U < F_k$.

One way to generalize the inversion method to directed networks is via the use of *copulas* [40], that is, a bivariate probability distribution with uniform marginals and specified correlation ρ between the random variables. If we denote the cumulative distribution function of the copula by $C(U_1, U_2)$ and the desired marginals' CDFs by $F_1(X_1) \equiv \text{Prob}[X \leq X_1]$ and $F_2(X_2) \equiv \text{Prob}[X \leq X_2]$, then we can generate a pseudo-random pair (X_1, X_2) from our desired bivariate probability distribution with marginals F_1 and F_2 by first drawing a random sample (U_1, U_2) from C and then constructing $\left(X_1 = F_1^{-1}(U_1), X_2 = F_2^{-1}(U_2)\right)$. In this way, X_1 and X_2 will have the approximate correlation between U_1 and U_2 as specified in the construction of C, as well as be representative samples from marginal distributions F_1 and F_2, respectively.

The next two sub-sections describe work in sampling networks with a fixed degree sequence and specified node-degree and/or edge-degree correlation.

4.2. **Node-degree correlation.** The *node-degree correlation* [47] of a finite network can be quantified in several ways, perhaps most intuitively using the Pearson correlation coefficient

$$\rho = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{d_i^+ - \mu^+}{\sigma^+} \right) \left(\frac{d_i^- - \mu^-}{\sigma^-} \right) \equiv \frac{\text{cov}(d^+, d^-)}{\sigma^- \sigma^+},$$

where μ^+ and σ^+ are the mean and standard deviation of the out-degrees of the nodes (similarly for μ^- and σ^-) and $\text{cov}(d^+, d^-)$ represents the covariance between d^+ and d^-. As described in the previous section, sampling networks from degree sequences with a prescribed node-degree correlation only require specification of the correlation coefficient between the in and out-degrees, as well as the marginal in and out-degree distributions.

In [31], the relationship between node-degree correlation and synchronization of pulse-coupled oscillators was explored. Examples in nature of pulse-coupled oscillators include fireflies in southeast Asia, as well as tonically firing neurons. Of particular interest in these situations is synchronization of the phases of each oscillator (see Section 5.1.2).

4.3. **Edge-degree correlation.** A natural next step is to sample networks with a fixed degree sequence and desired assortativity (3), or edge-degree correlation [27, 41]. To define edge-degree correlation, it is easiest to start with the directed case. Thus, for a directed graph, we define the edge-degree correlation as the Pearson correlation coefficient between the in-degrees (out-degrees) at the tail and in-degrees (out-degrees) at the head of every arc. This is given by

$$\rho_e(G) = \frac{1}{M} \sum_{(i,j) \in A} \left(\frac{d_i^p - \mu_1^p}{\sigma_1^p} \right) \left(\frac{d_j^q - \mu_2^q}{\sigma_2^q} \right) = \frac{\frac{1}{M} \sum_{(i,j) \in A} d_i^p d_j^q - \mu_1^p \mu_2^q}{\sigma_1^p \sigma_2^q}, \tag{5}$$

where M is the number of arcs, $p, q \in \{-, +\}$ and μ_k^p, σ_k^q are the mean and standard deviation of d^p or d^q for vertices at the tail ($k = 1$) or head ($k = 2$) of all arcs. In the

undirected case, by transforming every edge into a bidirectional arc, it can be shown that the edge-degree correlation in (5) is equivalent to (3), in other words, $\rho_e(G) = r(G)$.

Due to the relationship mentioned in Section 2 between $s(G)$ and $\rho_e(G)$, many people choose to use the metric $s(G)$ when working with edge-degree correlations, which we do as well. One of the techniques [43, 47] to increase or decrease the edge-degree correlation is to do 2-swaps between two edges (i,j) and $(m,n) \in E$ when the *edge-degree increment* $\Delta\big((i,j),(m,n)\big) = d_i d_j + d_m d_n - d_i d_n - d_m d_j = (d_i - d_m)(d_j - d_n)$ is positive (decrease edge-degree correlation) or negative (increase edge-degree correlation).

In the directed case, we have four different measures of edge-degree correlation [57] given by

$$s^{pq}(G) = \sum_{(i,j)\in A} d_i^p d_j^q,$$

where $p, q \in \{-,+\}$ (see (2)). This can be seen as a natural extension of $s(G)$ to the directed case, and has a similar relationship to (5) as edge-degree correlation and assortativity have in the undirected case. Now we have four edge-degree increments given by

$$\Delta^{pq}\big((i,j),(m,n)\big) = (d_i^p - d_m^p)(d_j^q - d_n^q), \tag{6}$$

where $p, q \in \{-,+\}$.

We will now illustrate an algorithm on directed graphs, similar to the undirected version [43, 47], which attempts to sample networks with a fixed degree sequence and desired edge-degree correlations $\{s^{--}, s^{-+}, s^{+-}, s^{++}\}$. The key observation is that certain arc-swaps modify only one of the four edge-degree correlations at a time. To see this, considering (6), if you want to vary correlation s^{-+} and leave the others fixed, for example, then you only consider swaps between an arc $(i,j) \in A$ with arcs in the set $A_{ij}^{-+} \equiv \{(m,n) \in A \mid d_m^+ = d_i^+ \text{ and } d_n^- = d_j^-\}$. Thus, for $p = +$ or $q = -$, we have $\Delta^{pq}((i,j), A_{ij}^{-+}) = 0$, and thus any two-swap between (i,j) and an arc in A_{ij}^{-+} leaves s^{+-}, s^{++} and s^{--} fixed. For general s^{pq}, we construct the set A_{ij}^{pq} as $A_{ij}^{pq} \equiv \{(m,n) \in A \mid d_m^q = d_i^q \text{ and } d_n^p = d_j^p\}$. The general algorithm then cycles through the four edge-degree correlations and performs an arc-swap between a random arc (i,j) and an arc in A_{ij}^{pq} if the supposed swap moves the edge-degree correlation s^{pq} towards the desired value. Similar algorithms exist (see [57], for example) to sample networks with desired edge-degree correlations. Note that there is a dependency between the node-degree and edge-degree correlations [47], so that we may or may not be able to achieve a network with our desired correlation structure.

5. Applications

5.1. **Synchronization on networks.** Synchronization of processes is ubiquitous in the biological sciences, for example the synchronization of neurons in the preBötzinger complex which drives the breathing rhythm [15], the synchronization of repressilator networks in

gene transcription [18], and the synchronous release of oxytocin from magnocellular hypothalamic neurons in neuroendocrinology [48]. Network structure and oscillator dynamics play fundamental roles in the dynamics of the system. There are many models for the oscillator dynamics that are considered, ranging from the complexity of Hodgkin-Huxley neuron models to phase models where we track only the phase of the oscillators and ignore their positions. The three phase models that have received the most attention are the Kuramoto oscillator, the Laplacian oscillator (4), and pulse-coupled oscillators. In the next two subsections, we discuss the effect of network structure on the synchronous bursting of neurons in the preBötzinger complex and the phase synchronization of pulse-coupled oscillators.

5.1.1. *Neuronal networks.* In mammals, a small group of neurons in the brain stem, the pre-Bötzinger complex [15], is responsible for generating a regular rhythmic output to motor cells that initiate a breath. Disconnected, these neurons are unable to provide sufficient output to activate the motor neurons, but their interconnected network structure allows them to synchronize without any external influence and produce regular bursts. This is a clear example of an important neuronal network where robustness (and synchronization [27]) is essential.

In [49], Del Negro et al. developed a physiologically realistic mathematical model of neurons in the preBötzinger complex that demonstrates the capability of the breathing rhythm to be an emergent phenomena of the network and not explicitly controlled by central pattern generators. Although there is very little known regarding the network structure of the preBötzinger complex, it is possible, using the model in [49], to test various network structures. Two geometric graphs with extreme values of λ_2 were tested in [23] and [27]. One of the geometic networks had a value of $\lambda_2 = 0.025$ and a second had a value of $\lambda_2 = 0.974$. The rhythmic output from the the network with $\lambda_2 = 0.025$ was ragged with fuzzy bursts, while outputs from the network with $\lambda_2 = 0.974$ was sharp with clear, regular bursts. The results of the two simulations, depicted in Figure 6, provide compelling evidence for the utility of λ_2 as a predictor of synchronization. It is easy to see that the network with higher λ_2 synchronizes more strongly than the other network. These experiments provide further evidence that λ_2 can be used to identify graphs (networks) that are not likely to synchronize.

5.1.2. *Pulse-coupled oscillators.* In [31], the effect of synchronization of homogeneous pulse-coupled oscillators on node-degree correlation (see Section 4.2) was studied. The dynamics of pulse-coupled oscillators is given by

$$\frac{d\phi_i}{dt} = 1 + \frac{k}{n}\sum_{j=1}^{n} A_{ji} H(\phi_j)\Delta(\phi_i), \quad \phi_i \in [0,1]$$

with $\Delta(\phi)$ the sensitivity function (phase response curve) and $H(\phi_j) = \delta(t - t_j)$ the pulsatile interaction function. In this notation, δ is a delta function with infinite point mass at 0 and t_j is the firing time for oscillator j. The matrix A_{ji} is the node-node adjacency matrix

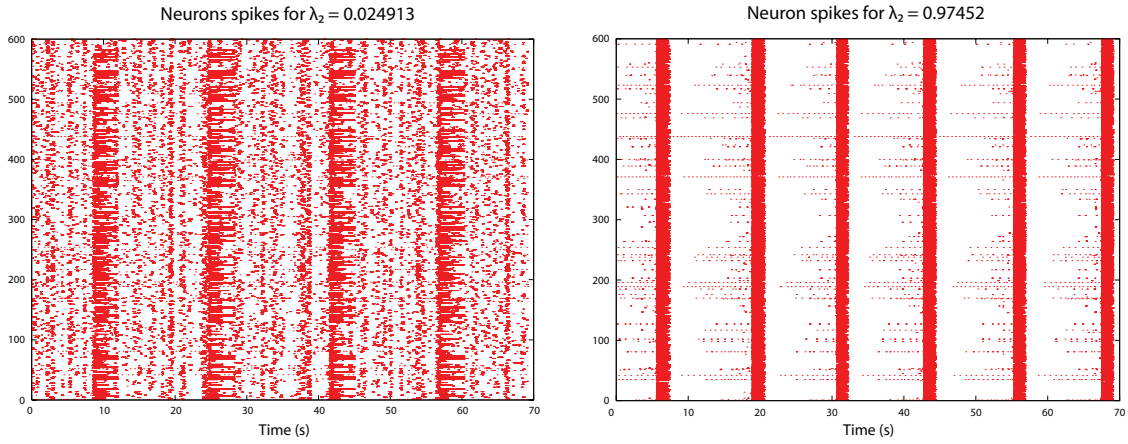


FIGURE 6. Raster plots of neuron output for two networks with disparate λ_2 values. A point at (x, y) indicates neuron x is spiking at time y. The higher λ_2 network displays much stronger synchronization among all nodes as predicted, as well as a quicker breath frequency.

for the network (not the Laplacian as in (4)), while k/n is the coupling constant, which includes the $1/n$ so that it is well-behaved in the thermodynamic limit $n \to \infty$. Strogatz and Mirollo [37] showed that in the case of all-to-all coupling, there is a globally synchronous state $\phi_1 = \cdots = \phi_n$ for almost all initial conditions, i.e. the initial conditions that do not reach global synchrony are a set of measure zero.

The top panel of Figure 7(a) shows the phases $\phi_k(t)$ for 200 oscillators on a network with negative node-degree correlation $\rho \approx -1$. The global synchronization measure (or coherence measure) given by $r(t) = |\sum_{k=1}^{n} \exp(2\pi i \phi_k(t))|/n$ is displayed in the bottom panel. Note that the global synchronization measure is 1 when the oscillators are globally synchronized. Contrary to Figure 7(a) is a similar plot in Figure 7(b) for a network with positive node-degree correlation $\rho \approx 1$. In this case, complete synchronization occurs at approximately $T = 500$ when $r(500) \approx 1$.

Numerical experiments in [31] demonstrated that the proportion of initial conditions resulting in a globally synchronous state is an increasing function of node-degree correlation. For those networks observed to globally synchronize, both the mean and standard deviation of time to synchronization decrease as node-degree correlation increases. Many networks with negatively correlated node degree exhibited multiple coherent attracting states, with trajectories performing fast transitions between them. A similar phenomenon was reported in [53, 28] in networks of pulse-coupled oscillators with delay.

As stated in Section 3, the algebraic connectivity λ_2 is known to have an effect on the rate of convergence to the globally synchronous state when identical oscillators are coupled as in the Kuramoto model or via an undirected Laplacian [25, 1]. Similarly, the eigenvalue

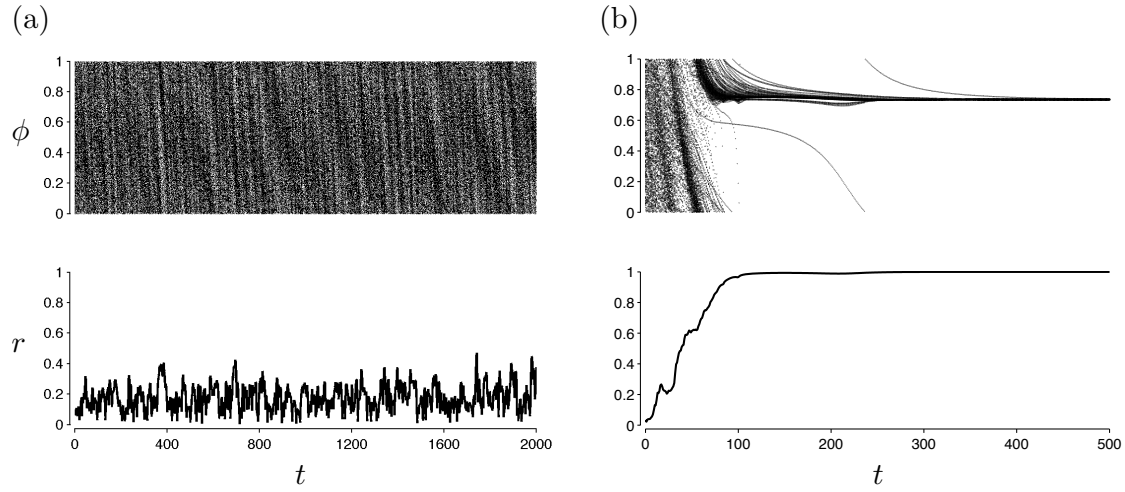


FIGURE 7. (a) The top panel shows the phase ϕ of 200 oscillators as a function of time for a network with node-degree correlation $\rho \approx -1$, with the corresponding global synchronization measure $r(t)$ shown in the botton panel. (b) Same as in (a) except with $\rho \approx 1$.

ratio λ_N/λ_2 is related to the propensity of identical oscillators coupled via an undirected Laplacian to have a stable synchronous state [4]. Figure 8 shows the mean Laplacian eigenvalue λ_2 (panel (a), solid line) and the ratio λ_N/λ_2 (panel (b), dashed line) are monotone increasing and decreasing functions of node-degree correlation, respectively (average over 1,000 random networks for each ρ). For pulse-coupled networks with negatively correlated node degree, slow synchronization and reduced percentage of initial conditions that reached global synchronization is observed.

5.2. **Learning from Internet Design.** The problems faced by designers of air transport networks share some aspects with the design of an Internet router network. Many authors have contributed to investigations of how a router network is constructed. Two references in this field, [32] and [33], contain ideas central to our consideration of the design of air transport networks. At one level of resolution, Table 1 points out the analogies between these two network design problems. With regard to bandwidth, the Internet router designer must weigh the trade-offs between many low bandwidth connections and fewer high bandwidth connections. These trade-offs are akin to choosing between a few hub airports in a hub-and-spoke system and choosing lower frequency airports that might arise in a direct route system. Of course, there are many differences as well. The variation in the size of the packets for the Internet is not nearly as great as the number of passengers on planes of different sizes. In addition, although the FAA[3] clearly defines the routes allowed between airports, the links are as not hard-wired as they are in the Internet model. Still

[3]Federal Aviation Administration

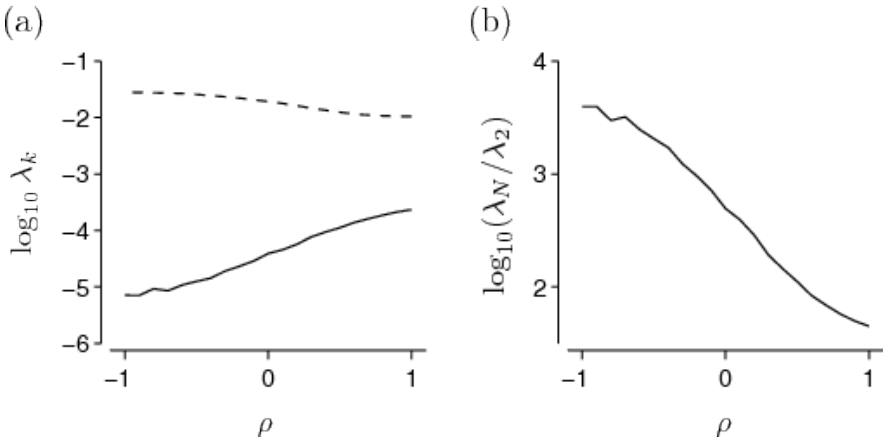


FIGURE 8. Laplacian eigenvalues λ_N (dashed line) and λ_2 (solid line) in (a) and ratio λ_N/λ_2 in (b). Each point on the curves shows an average over 1,000 random networks.

there is much to be learned from the research efforts on the design of effective Internet router networks.

	Internet	**Air Transport**
product	packets	planes (loaded)
constraint	bandwidth	airport capacity
competitors	ISPs	airlines
links	hardwired	FAA/Airlines
distributors	routers	airports

TABLE 1. Analogy between Internet router and air transport networks.

There has been an inordinate amount of interest in networks whose complementary cumulative degree distributions have a fat tail or follow a power law. In [32] a figure is presented which plots $s(G)$ versus thruput when the degree distribution is fixed. The figure highlights the error in focusing exclusively on the form of the degree distribution. A normalized $s(G)$ value is plotted along the x-axis and a thruput metric is plotted along the y-axis. Each data point represents the performance of a network, each of which has an identical degree distribution following a power law. An unexpectedly wide variance in the thruput performance of these networks in which the degree distribution is an invariant is observed. Moreover, low (disassortative) $s(G)$ instances lead, in general, to better thruput performance. The authors in [32] point out that when sampling from this invariant degree distribution it was much more likely to draw an instance in which $s(G)$ is large (assortative).

6. Optmization: Perfect b-matching and $s(G)$

There has been little work done with regard to classifying optimization problems associated with graph invariants. In this section we address optimization problems associated with $s(G)$ [30]. In this context, a natural optimization problem is:

Minimum Randić Index Problem. Given a degree sequence what is a graph realization with the minimum Randić index?

We define the *connected minimum Randić index problem* as the minimum Randić index problem with the additional constraint that the graph realization is connected. For a graph $G = (V, E)$ and a positive integer vector $b = (b_1, \cdots, b_n) \in \mathbb{Z}^n$, a *perfect b-matching* is a subset of edges $M \subseteq E$ such that for node $i \in V$, the degree of i in the graph (V, M) is b_i.

An associated optimization problem is:

Minimum Weight Perfect b-Matching Problem. Given a positive integer vector b, a graph $G = (V, E)$ and a set of edge weights $w : E \to \mathbb{R}$, find a perfect b-matching with minimum weight.

The minimum Randić index problem is equivalent to the minimum weight perfect b-matching problem on a complete graph G with an appropriate choice of weights [30]. In [30] it was shown that by constraining the matchings to be connected, for an arbitrary graph G, the minimum weight perfect b-matching problem becomes NP-Hard. In 2008, Beichl and Cloteaux [5] investigated how well random networks generated with a chosen $s(G)$ can model the structure of real networks such as the Internet. The graphs produced optimizing the $s(G)$ resulted in better models than the ones that used simple uniform sampling.

6.1. Formulation and Complexity.

The minimum Randić index problem can be formulated as a *minimum weight perfect b-matching problem*, which is solvable in polynomial time [51]. Note that the perfect b-matching problem does not enforce connectivity. When connectivity of solutions is desired, in [30] it is shown that even approximating the minimum weight perfect b-matching problem with connectivity is NP-Hard.

Consider a graph $G = (V, E)$, a positive integer vector $b = (b_1, \cdots, b_n) \in \mathbb{Z}^n$ and $M \subseteq E$, a perfect b-matching. For a given b-matching, M, the graph induced by M is (V, M). Denote the set of perfect b-matchings of a graph G by $\mathcal{P}_b(G)$. For edge weights $w : E \to \mathbb{R}$, the *minimum weight perfect b-matching problem* requires finding the perfect b-matching with

minimum weight, i.e., to calculate

$$M^*(G) := \arg\min \left\{ \sum_{e \in M} w(e) : M \in \mathcal{P}_b(G) \right\}. \tag{7}$$

To formulate an instance of the minimum Randić index problem as a minimum weight perfect b-matching problem, set

$$w_{ij} = b_i \cdot b_j. \tag{8}$$

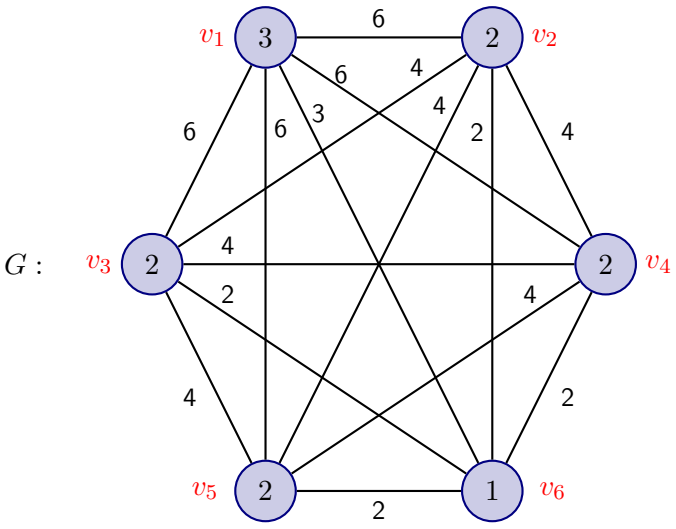
For these weights, solve (7) to obtain $M^*(G)$. $G^* = (V, M^*(G))$ is an optimal solution to the minimum Randić index problem instance. Note first that it is feasible since the degree of a node $i \in V$ is b_i by the definition of the perfect b-matching problem. Note also that any feasible graph to the minimum Randić index problem is also a perfect b-matching because the degree of any node i is equal to b_i. Moreover, (8) implies

$$R(G^*) = \sum_{(i,j) \in M^*(G)} b_i \cdot b_j = \sum_{(i,j) \in M^*(G)} w_{ij}.$$

Since any graph that is feasible to the minimum Randić index is also a b-matching, the optimality of $M^*(G)$ implies the optimality of G^*.

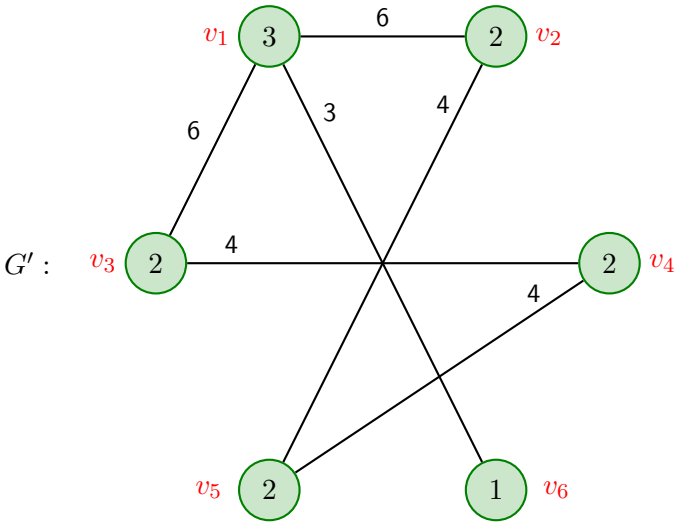
Therefore, an instance of a minimum weight perfect b-matching on a complete graph can be constructed to solve the minimum Randić index problem. Since the b-matching problem can be solved in polynomial time, finding the minimum Randić index of a graph can also be done in polynomial time. Optimal solutions, however, are not necessarily connected.

Consider the following example. Given the degree sequence $d = (3, 2, 2, 2, 2, 1)$, what is a graph realization with the minimum Randić index? Let $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ with $b = \begin{bmatrix} 3 & 2 & 2 & 2 & 2 & 1 \end{bmatrix}$ be given. Next form the complete graph G, with weights corresponding to $b_i \cdot b_j$ for every node $v_i, v_j \in V$.



$$\begin{bmatrix} 0 & 6 & 6 & 6 & 6 & 3 \\ 6 & 0 & 4 & 4 & 4 & 2 \\ 6 & 4 & 0 & 4 & 4 & 2 \\ 6 & 4 & 4 & 0 & 4 & 2 \\ 6 & 4 & 4 & 4 & 0 & 2 \\ 3 & 2 & 2 & 2 & 2 & 0 \end{bmatrix}$$

Solve the minimum weight perfect b-matching for G and obtain G':



$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$R(G') = 6 + 6 + 4 + 4 + 4 + 3 = 27$$

G' is an optimal solution for the minimum weight perfect b-matching. The sum of the weights is the minimum Randić index and the unweighted adjacency matrix is the corresponding graph realization. Note that there are other solutions to the matching that will produce the minimum Randić index and a different realization. That is, the solution is not unique.

6.2. **Solving the Minimum Randić index Problem.** In this section, the input graph G is assumed to be complete. A code written by Vlad Schogolev, Bert Huang, and Stuart Andrews [24] that makes use of the GOBLIN graph library (http://goblin2.sourceforge.net/) is designed to solve a *maximum* weight perfect b-matching problem. Given a weight matrix H we transform these weights into a matrix H_2 such that the maximum matching using

H_2 will yield the same solution as the minimum matching using H. To do this we take a matrix M with ones in all positions except for the diagonal which has zeros. We then multiply every entry by one more than the maximum entry of H. Then H is subtracted from M yielding H_2. An algorithm that will solve the minimum Randić index problem for a given degree sequence is given below.

ALGORITHM TO SOLVE MINIMUM RANDIĆ INDEX WITH b-MATCHING

Inputs: A, an adjacency matrix with degree sequence, d.
Outputs: G, the new adjacency matrix with degree sequence, d and minimized Randic index, r.

Create a complete graph H of degree products
Transform H to H_2 for b-matching code
Use b-match solver to get adjacency matrix, G of optimal solution
Calculate $r = R(G)$
return G and r

FIGURE 9. Solving minimum Randić index with b-matching

The algorithm in Figure 9 returns the minimum Randić index of a graph and a realization. The b-matching code runs in polynomial time ([51]) and it is easy to see that the transformation steps are done in polynomial time as well. Three types of randomly generated graphs to test the algorithm performance: Erdős-Rényi, geometric and scale-free. The computational experiments are limited to graphs for which connected realizations are known to exist. The Randić index before and after the optimization is recorded. After the optimization the graph realization with the minimum Randić index is checked to see if it is connected. In [30], computational experiments for a number of graph sizes and types are reported. Here we include the results for 100 replications of three types of 100 node graphs in table 2. Note that the number of graphs connected after the run plus the number of graphs disconnected plus the number of graphs with no connected realizations is 100 for each graph type.

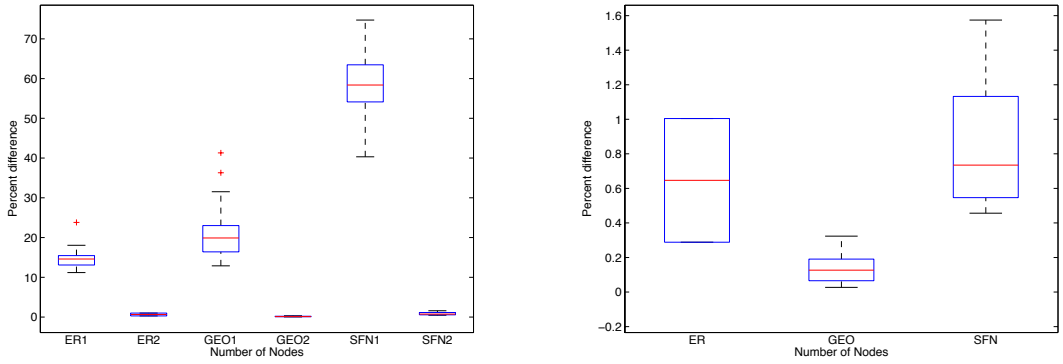
Graph type	**connected**	**disconnected**	**no connected realizations**
Erdős-Rényi	16	2	82
Geometric	30	6	64
Scale-Free	91	8	1

TABLE 2. 100 node graphs

The MATLAB functions used to generate the geometric and scale-free graphs are from CONTEST: A Controllable Test Matrix Toolbox for MATLAB [52]. In addition, the necessary and sufficient conditions for a set $\{a_i\}$ to be realizable (as the degrees of the nodes

of a connected graph) are that $a_i \neq 0$ for all i and the sum of the integers a_i is even and not less than $2(n-1)$. This condition was used to discard graphs with a degree sequence that had no connected realizations [9].

The left box plots for each of the 100 node graphs in Figure 10 show the percent difference between the graph's original Randić index and the minimum Randić index. The percent difference is calculated from $\frac{original-minimum}{minimum} \times 100$. In each pair of plots, the right box plot describes the performance of the heuristic. The right box plots and the heuristic are described in the next section.



(a) Comparing percent differences for Erdős-Rényi , geometric and scale-free graphs.

(b) Comparing percent differences between optimal and heuristic solutions

FIGURE 10. 100 node graph results

6.3. **Heuristic for Disconnected Realizations.** The complexity of the minimum Randić index problem subject to a connectivity constraint is not known. However, since some of the graph realizations with the minimum Randić index were disconnected, we developed a heuristic using two-switches to connect these realizations. (See Figure 11 for the algorithm.) The heuristic performs a two-switch between every component until all the components are connected. We know that a two-switch exists between any two-connected components because they do not share any edges. Any edge can be used.

The heuristic was applied to all optimal solutions that were disconnected. In general, the difference in Randić index from the minimum was not significant. The Randić index changes the least after the heuristic in the Erdős-Rényi graphs. This percent difference is calculated with $\frac{after\ heuristic-minimum}{minimum} \times 100$. The number of graphs that used the heuristic depended on the number of optimal graph realizations that were disconnected. Note that

this is a different number for each graph type and size. See Table 2 for the 100 node graph numbers.

Two-switch Heuristic

Inputs: A, an adjacency of disconnected graph
Outputs: A, the new adjacency matrix of connected graph

while the number of connected components in A is ≥ 2
do a two switch with components 1 and 2 to connect them
using two randomly chosen edges from each component

return A

FIGURE 11. Connecting disconnected graph with two-switch heuristic

Note that the method to connect the disconnected realizations may not produce graphs with the best structure since there is only 1 edge connecting one component to another. Also note that we do not need to check whether the randomly chosen edges are adjacent or not since they are in separate connected components. In addition, once we connect components 1 and 2, component 2 becomes part of component 1 and component 3 becomes the new component 2. Therefore we can always connect components 1 and 2.

7. Conclusion

The importance of how network invariants affect the dynamics of a process on the network (e.g. respiration, current, traffic) has been highlighted. In studying the potential effect of incremental changes in network invariants on network dynamics, the degree distribution of a network was held constant while other network invariants were examined. In particular, results demonstrating the effects of assortativity, $s(G)$, and eigenvalues of the Laplacian on network dynamics were presented.

In section 3, the connection between λ_2 and other network dynamics is studied. Research supporting the link between λ_2 and synchronization are provided as well as an inverse relationship between λ_2 and the diameter of the network. Section 4 summarizes algorithms that allow sampling (sometimes uniform) from the family of networks with a fixed degree sequence. The ability to sample uniformly is critical in any research attempting to discover the effects of network invariants. In section 5 a number of network applications are described including neuronal networks for respiration in mammals, pulse-coupled oscillators, internet router networks and air transportation route networks. Section 6 focuses on optimizing $s(G)$, a network assortativity metric. A novel connection with minimum weight perfect b-matching problem as well as computational results is given.

References

[1] J. Almendral and A. Díaz-Guilera, *Dynamical and spectral properties of complex networks*, New Journal of Physics, 9 (2007), pp. 187–187.

[2] F. Atay, T. Biyikoglu, and J. Jost, *Synchronization of networks with prescribed degree distributions*, Circuits and Systems I: Regular Papers, IEEE Transactions on, 53 (2006), pp. 92–98.

[3] F. Atay, J. Jost, and A. Wende, *Delays, connection topology, and synchronization of coupled chaotic maps*, Physical review letters, 92 (2004), p. 144101.

[4] M. Barahona and L. Pecora, *Synchronization in small-world systems*, Physical Review Letters, 89 (2002), p. 54101.

[5] I. Beichl and B. Cloteaux, *Measuring the effectiveness of the s-metric to produce better network models*, in Simulation Conference, 2008. WSC 2008. Winter, IEEE, 2008, pp. 1020–1028.

[6] A. Berger and M. Müller-Hannemann, *Uniform sampling of undirected and directed graphs with a fixed degree sequence*, arXiv, math.DM (2009).

[7] J. Blitzstein and P. Diaconis, *A Sequential Importance Sampling Algorithm for Generating Random Graphs with Prescribed Degrees*, Internet Mathematics, 6 (2011), pp. 489–522.

[8] V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky, *Convex graph invariants*, SIAM Review, 54 (2012), pp. 513–541.

[9] W. Chen, *Graph theory and its engineering applications*, World Scientific Pub. Co. Inc., 1997.

[10] F. Chung, *Laplacians and the cheeger inequality for directed graphs*, Annals of Combinatorics, 9 (2005), pp. 1–19.

[11] G. W. Cobb and Y.-P. Chen, *An application of markov chain monte carlo to community ecology*, The American Mathematical Monthly, 110 (2003), pp. 265–288.

[12] V. Colizza, J. Banavar, A. Maritan, and A. Rinaldo, *Network structures from selection principles*, Physical review letters, 92 (2004), p. 198701.

[13] L. Donetti, F. Neri, and M. Muñoz, *Optimal network topologies: Expanders, Cages, Ramanujan graphs, Entangled networks and all that*, Journal of Statistical Mechanics: Theory and Experiment, 2006 (2006), p. P08007.

[14] P. Erdös and A. Rényi, *On the evolution of random graphs*, in PUBLICATION OF THE MATHEMATICAL INSTITUTE OF THE HUNGARIAN ACADEMY OF SCIENCES, Jan. 1960, pp. 17–61.

[15] J. L. Feldman and C. A. D. Negro, *Looking for inspiration: new perspectives on respiratory rhythm*, Nat Rev Neurosci, 7 (2006), pp. 232–42.

[16] M. Fiedler, *Algebraic connectivity of graphs*, Czechoslovak Math. J., 23(98) (1973), pp. 298–305.

[17] L. R. Ford and D. R. Fulkerson, *Maximal flow through a network*, Canad. J. Math., 8 (1956), pp. 399–404.

[18] J. Garcia-Ojalvo, M. B. Elowitz, and S. H. Strogatz, *Modeling a synthetic multicellular clock: repressilators coupled by quorum sensing*, Proc. Natl. Acad. Sci. U.S.A., 101 (2004), pp. 10955–60.

[19] C. I. D. Genio, H. Kim, Z. Toroczkai, and K. E. Bassler, *Efficient and exact sampling of simple graphs with given arbitrary degree sequence*, PLoS ONE, 5 (2010), pp. 1–7.

[20] E. N. Gilbert, *Random graphs*, Annals of Mathematical Statistics, 30 (1959), pp. 1141–1144.

[21] S. Hakimi, *On realizability of a set of integers as degrees of the vertices of a linear graph. i*, Journal of the Society for Industrial and Applied Mathematics, (1962), pp. 496–506.

[22] V. Havel, *A remark on the existence of finite graphs*, Casopis Pest. Mat, 80 (1955), pp. 477–480.

[23] M. Holroyd and R. K. Kincaid, *Synchronizability and connectivity of discrete complex systems*, in Proccedings of the International Conference on Complex Systems 2006, New England Complex Systems Institute, June 2006.

[24] B. Huang and T. Jebara, *Loopy belief propagation for bipartite maximum weight b-matching*, Artificial Intelligence and Statistics (AISTATS), (2007).

[25] A. Jadbabaie, N. Motee, and M. Barahona, *On the stability of the kuramoto model of coupled nonlinear oscillators*, American Control Conference, 5 (2004), pp. 4296– 4301 vol.5.

[26] H. Kim, C. I. Del Genio, K. E. Bassler, and Z. Toroczkai, *Constructing and sampling directed graphs with given degree sequences*, New Journal of Physics, 14 (2012), p. 023012.
[27] R. Kincaid, N. Alexandrov, and M. Holroyd, *An investigation of synchrony in transport networks*, Complexity, 14 (2009), pp. 34–43.
[28] C. Kirst and M. Timme, *From networks of unstable attractors to heteroclinic switching*, Phys. Rev. E, 78 (2008), pp. 065201–065204.
[29] D. Kleitman and D. Wang, *Algorithms for constructing graphs and digraphs with given valences and factors*, Discrete Math., 6 (1973), pp. 79–88.
[30] S. J. Kunkler, M. D. LaMar, R. K. Kincaid, and D. Phillips, *Algorithm and Complexity for a Network Assortativity Measure*, arXiv, (2013).
[31] M. D. LaMar and G. D. Smith, *Effect of node-degree correlation on synchronization of identical pulse-coupled oscillators*, Phys. Rev. E, 81 (2010), p. 046206.
[32] L. Li, D. Alderson, J. Doyle, and W. Willinger, *Towards a theory of scale-free graphs: Definition, properties, and implications*, Internet Mathematics, 2 (2005), pp. 431–523.
[33] L. Li, D. Alderson, W. Willinger, and J. Doyle, *A first-principles approach to understanding the internet's router-level topology*, ACM SIGCOMM Computer Communication Review, 34 (2004), pp. 3–14.
[34] X. Li, Y. Shi, and L. Wang, *An updated survey on the Randić index*, Recent Results in the Theory of Randic index, Mathematical Chemistry Monographs, 6 (2008).
[35] F. Liljeros, C. R. Edling, L. A. Amaral, H. E. Stanley, and Y. Aberg, *The web of human sexual contacts*, Nature, 411 (2001), pp. 907–8.
[36] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabasi, *Controllability of complex networks*, Nature, 473 (2011), pp. 167–173.
[37] R. Mirollo and S. Strogatz, *Synchronization of pulse-coupled biological oscillators*, SIAM J. Appl. Math., 50 (1990), pp. 1645–1662.
[38] B. Mohar, *Some applications of Laplace eigenvalues of graphs*, Graph Symmetry: Algebraic Methods and Applications, 497 (1997), pp. 227–275.
[39] A. Motter, *Bounding network spectra for network design*, New Journal of Physics, 9 (2007), p. 182.
[40] R. B. Nelsen, *An Introduction to Copulas*, Springer, 1998.
[41] M. Newman, *J. 2003.The Structure and Function of Complex Networks.*, SIAM Review, 45 (2003), p. F00.
[42] M. Newman and M. Girvan, *Finding and evaluating community structure in networks*, Physical review E, 69 (2004), p. 26113.
[43] M. E. J. Newman, *Assortative mixing in networks*, Phys. Rev. Lett., 89 (2002), p. 208701.
[44] T. Nishikawa and A. Motter, *Maximum performance at minimum cost in network synchronization*, Physica D: Nonlinear Phenomena, 224 (2006), pp. 77–89.
[45] M. Pósfai, Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, *Effect of correlations on network controllability.*, Sci Rep, 3 (2013), p. 1067.
[46] M. Randić, *Characterization of molecular branching*, Journal of the American Chemical Society, 97 (1975), pp. 6609–6615.
[47] J. G. Restrepo, E. Ott, and B. R. Hunt, *Approximating the largest eigenvalue of network adjacency matrices*, Phys. Rev. E, 76 (2007), p. 056119.
[48] E. Rossoni, J. Feng, B. Tirozzi, D. Brown, G. Leng, and F. Moos, *Emergent synchronous bursting of oxytocin neuronal network*, PLoS Comput Biol, 4 (2008), p. e1000123.
[49] J. E. Rubin, J. A. Hayes, J. L. Mendenhall, and C. A. D. Negro, *Calcium-activated nonspecific cation current and synaptic depression promote network-dependent burst oscillations*, Proc. Natl. Acad. Sci. U.S.A., 106 (2009), pp. 2939–44.
[50] H. Ryser, *Combinatorial properties of matrices of zeros and ones*, Canad. J. Math, 9 (1957), p. 6.
[51] A. Schrijver, *Combinatorial optimization*, Springer, 2003.

[52] A. Taylor and D. Higham, *Contest: A controllable test matrix toolbox for matlab*, ACM Transactions on Mathematical Software (TOMS), 35 (2009).
[53] M. Timme, F. Wolf, and T. Geisel, *Prevalence of Unstable Attractors in Networks of Pulse-Coupled Oscillators*, Phys. Rev. Lett., 89 (2002), p. 154105.
[54] D. West, *Introduction to graph theory, 2nd ed.*, Prentice Hall Upper Saddle River, NJ, 2001.
[55] C. Wu, *Perturbation of coupling matrices and its effect on the synchronizability in arrays of coupled chaotic systems*, Physics Letters A, 319 (2003), pp. 495–503. Preprint 0307052 (2006) available at http://arXiv/pdf/nlin.CD/.
[56] C. Wu, *On rayleigh-ritz ratios of a generalized laplacian matrix of directed graphs*, Linear Algebra and Its Applications, 402 (2005), pp. 207–227.
[57] G. Zamora-López, C. Zhou, V. Zlatić, and J. Kurths, *The generation of random directed networks with prescribed 1-node and 2-node degree correlations*, Journal of Physics A: Mathematical and Theoretical, 41 (2008), p. 224006.