Algorithm and Complexity for a Network Assortativity Measure

Sarah J. Kunkler*  M. Drew LaMar†  Rex K. Kincaid‡  David Phillips§

July 2, 2013

Abstract

We show that finding a graph realization with the minimum Randić index for a given degree sequence is solvable in polynomial time by formulating the problem as a minimum weight perfect b-matching problem. However, the realization found via this reduction is not guaranteed to be connected. Approximating the minimum weight b-matching problem subject to a connectivity constraint is shown to be NP-Hard. For instances in which the optimal solution to the minimum Randić index problem is not connected, we describe a heuristic to connect the graph using pairwise edge exchanges that preserves the degree sequence. In our computational experiments, the heuristic performs well and the Randić index of the realization after our heuristic is within 3% of the unconstrained optimal value.

1 Introduction

Networks are ubiquitous in the sciences. 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 particular graph property may be more important than another. Oftentimes, a desired property is to have a connected graph or to optimize a particular metric while constrained to connected graphs [10].

One of these measures, the Randić index of a graph, developed by Milan Randić, was originally used in chemistry [13]. The Randić index

*Department of Mathematics, College of William and Mary, Williamsburg, VA 23187-8795, USA, sjkunkler@email.wm.edu
†Department of Biology, College of William and Mary, Williamsburg, VA 23187-8795, USA, mdlama@wm.edu
‡Department of Mathematics, College of William and Mary, Williamsburg, VA 23187-8795, USA, rrkinc@wm.edu
§Mathematics Department, United States Naval Academy, Annapolis, MD 21401, USA, dphillip@usna.edu
of a graph can be thought of as an assortativity measure. A network is described as disassortative if high-degree nodes are predominantly attached to low-degree nodes [11]. Minimizing the Randić index, in many instances, will produce a graph with disassortativity [11]. Why is this optimization problem of interest? Li et al. [11], in the design of an internet router network, found that networks that maximized throughput also had small values for the Randić index. In addition, the Randić index has been shown [9] to correlate with synchronization, an important property in many network applications. Our focus is to investigate algorithms that minimize the Randić index of a graph over all connected realizations while keeping the degrees of the nodes fixed.

1.1 Notation and Definitions

We assume the reader to have a knowledge of graph theory (see, e.g., [18]). We consider an undirected graph, $G = (N, E)$, which consists of nodes, $N$, and edges, $E$.

We assume that our graph is undirected and simple, i.e., there are no self-loops and no multi-edges. The degree of a node is defined as $d_i(G) := |\{j : (i,j) \in E\}|$. We denote the node-node adjacency matrix with $A(G)$. When the particular graph is clear from context, we omit $G$ in the previous definitions.

The degree sequence is the list of the degrees of all the nodes in a graph, which we represent as $d(G) = (d_1, d_2, ..., d_n)$ where each $d_i$ is the degree of each node $i \in N$. Any sequence of non-negative integers is a potential degree sequence, but the sequence is considered graphic if it can produce a graph. Degree sequences can correspond to more than one adjacency matrix or graph. We call these graphs different realizations of the degree sequence.

Let nodes $u,v \in G$. We say that $u$ and $v$ are connected if there exists a path from $u$ to $v$. A graph is connected if for all $u \in N$ there exists a path to every other node.

**Definition 1.1.** The Randić index of a graph $G = (N, E)$ is defined as

$$R_\alpha(G) = \sum_{(i,j) \in E} (d_i \cdot d_j)^\alpha,$$

where $\alpha \in \mathbb{R} - \{1\}$.

A popular $\alpha$ used in chemistry is $\alpha = -\frac{1}{2}$ [6, 13]. For our purposes, we let $\alpha = 1$ and omit the subscript by denoting the Randić index of a graph by $R(G)$. A natural optimization problem is:

**Minimum Randić Index Problem.** Given a graphic 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.
Definition 1.2. For a graph $G = (N, E)$ 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 $i \in N$, the degree of $i$ in the graph $(N, M)$ is $b_i$. 

An associated optimization problem is:

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

In section 3, we will see that 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. We will also show that by constraining the matchings to be connected, for an arbitrary graph $G$, the minimum weight perfect $b$-matching problem becomes NP-Hard.

### 1.2 Network Measures of Assortativity

The Randić index of a graph was originally defined in chemistry. In 1975, the chemist Milan Randić [13] proposed the topological index $R_\alpha(N)$ for $\alpha = -1$ or $-1/2$ under the name branching index. He explained the utility of $R$ in measuring the extent of branching of the carbon-atom skeleton of saturated hydrocarbons. His approach "reveals some inherent relationships between [structures] which can be traced to connectivity" [13]. It is sometimes referred to as the connectivity index by scholars in chemistry [6]. Bollobás and Erdős [2] generalized this index by allowing $\alpha$ to take on any real number. A survey of results for the Randić index can be found in [12]. Around the same time as Randić was proposing his index, Gutman developed an almost identical measure called the Zagreb index, which is also used in chemistry [4]. There is a first Zagreb index which is defined as

$$M_1(G) = \sum_{u \in N} d_u^2$$

for a graph $G$. The second Zagreb index is defined as

$$M_2(G) = \sum_{(u,v) \in E} d_u \cdot d_v.$$

This is analogous to the Randić index with $\alpha = 1$ [4].

In 2005, Li et al. [11] introduced the $s$-metric of a graph which, unknown to them, is the Randić index when $\alpha = 1$. They defined the $s$-metric for a graph $G$ as

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

They used the $s$-metric to differentiate between graph realizations of a given degree sequence following a power law distribution [11] in the design of internet router networks. For a fixed degree distribution they plotted
the s-metric versus throughput for hundreds of realizations. They noted
that \( s(G) \) measures the “hub-like core” of a graph and is maximized when
high-degree nodes are connected to other high degree nodes (assortative).
Conversely, the minimum values of \( s(G) \) were predominantly associated
with networks that maximized throughput and were dissassortative.

In 2008, Beichl and Cloteaux 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-metric re-
sulted in better models than the ones that used simple uniform sampling
[1]. Randić index, Zagreb index and the s-metric are all variations of the
same basic measure. We will use the term Randić index, recognizing that
the use of either Zagreb index or s-metric would also be appropriate.

1.3 Random Graphs Classes

Our computational experiments require random graphs. We make use of
three types of graphs, Erdős-Rényi, geometric and scale-free. The struc-
ture of these graphs depends on the parameters chosen.

**Erdős-Rényi Graphs.** A number of nodes \( n \) and a probability of con-
nection \( p \) are chosen. A random probability is generated for each possible
edge. If the probability generated is less than the \( p \) then the edge is added.

**Geometric Graphs.** A number of nodes \( n \) is chosen and placed on a
unit square at random. This gives each node \( i \) coordinates \( x, y \). A radius
\( r \) is chosen. We connect nodes \( i \) and \( j \) if \( (x_i - x_j)^2 + (y_i - y_j)^2 \leq r^2 \) [16].

**Scale-Free 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 you reach \( n \) nodes, making it more
likely that a new node will be connected to a higher degree node [16]. The
algorithm allows a minimum node degree to be specified.

2 Algorithms

Our primary goal is to devise an algorithm to solve the minimum Randić
index problem. An algorithm that is useful when creating graphs with
a specified degree sequence is the Havel-Hakimi algorithm. The Havel-
Hakimi algorithm can be used to check if a degree sequence is graphic
and to find a realization of that sequence.
2.1 Havel-Hakimi Algorithm

The Havel-Hakimi algorithm can be useful when we have a degree sequence and we want to know: Given a non-negative integer sequence, is it graphic, and if so, what is a realization of the sequence? (See Figure 1.)

**Havel-Hakimi [7, 5]**

**Inputs:** $d$, a non negative integer sequence

**Outputs:** $G$, a graph realization of $d$ (if graphic)

Order $d$ so that it is non-increasing.
Create an empty $|d| \times |d|$ adjacency matrix.
while $d$ is not the 0 sequence
    do
        Pick a random node $i$ from $d$, with $d_i = x$.
        Subtract 1 from $x$ and 1 from $x$ nodes (starting with the largest degree) until $x = 0$.
        Put 1s in the adjacency matrix to represent the $x$ edges connected to $i$ and the $x$ largest nodes.
        Re-order $d$ so it is non-increasing
        if $|d| == 1$
            $d$ is not graphic

**return** $G = $ Adjacency matrix of $d$ (if graphic)

Figure 1: Havel-Hakimi Algorithm

2.2 Two-Switches and the Metagraph

One way to generate a collection of realizations for a degree sequence is to move from one realization to another by doing a two-switch. (See Figure 2.) When doing a two-switch, we examine two edges, $(a, b), (c, d) \in E$. If $(a, d) \notin E$ and $(b, c) \notin E$ then we can remove edges $(a, b)$ and $(c, d)$ and create edges $(a, d)$ and $(b, c)$. This is not a unique move, since we could also use $(a, c)$ and $(b, d)$ if $(a, c) \notin E$ and $(b, d) \notin E$. Two-switching is an easy way to obtain a different graph with the same degree sequence after a graph is created using the Havel-Hakimi algorithm.
An example of a two-switch.

| Two-Switch |
|------------|
| **Inputs:** A, an adjacency matrix with degree sequence, d. |
| **Outputs:** G, the new adjacency matrix with degree sequence, d. |
| Pick a random edge \((a, b) \in A\). |
| Find a node \(c\) that is not connected to \(b\). |
| Find a node \(d\) that is connected to \(c\), but not to \(a\). |
| Remove edges \((a, b), (c, d)\) from \(A\) and add \((a, d), (c, b)\) to create \(G\). |
| **return** \(G\) |

Figure 2: Two-Switch Algorithm

**Theorem 2.1.** (Ryser’s Theorem [14].) Given graphs \(G\) and \(G'\) such that \(d(G) = d(G')\), there exists a sequence of two-switches going from \(G\) to \(G'\).

Using this theorem, we can construct a metagraph of a degree sequence. The metagraph is an undirected graph where each node represents a graph realization of a degree sequence and each edge represents a two-switch. Note that, by Ryser’s Theorem, the metagraph is always a connected graph [14].

### 3 Formulation and Complexity

In this section, we formulate the minimum Randić index problem as a minimum weight perfect \(b\)-matching problem, which is solvable in polynomial time [15]. Note that this problem does not enforce connectivity. We then show that approximating the minimum weight perfect \(b\)-matching problem with connectivity is NP-Hard.

#### 3.1 The \(b\)-Matching Problem

Consider a graph \(G = (N, E)\), a positive integer vector \(b = (b_1, \ldots, b_n) \in \mathbb{Z}^n\) and \(M \subseteq E\), a perfect \(b\)-matching. For a given \(b\)-matching, \(M\), the graph induced by \(M\) is \((N, M)\). We denote the set of perfect \(b\)-matchings of a graph \(G\) by \(P_b(G)\). For edge weights \(w : E \rightarrow \mathbb{R}\), the minimum
weight perfect b-matching problem is finding the perfect b-matching with minimum weight, i.e., to calculate

$$M^*(G) := \arg \min \{ \sum_{e \in M} w(e) : M \in \mathcal{P}_b(G) \}. \quad (1)$$

For example, let G be the undirected, weighted graph below

and let $b = [2 \ 1 \ 1 \ 2]$ for nodes $v_1, v_2, v_3$ and $v_4$ respectively. We select $b_i$ edges that will connect to the $i$th node and that will produce the minimum weight. Therefore, the matching induces the graph $G'$ below.

Note that for this example the solution $G'$ is the only perfect b-matching for $G$.

We now show that the minimum Randić index problem is equivalent to the minimum weight perfect b-matching problem when the input graph for the b-matching problem is a complete graph. Thus, to formulate an instance of the minimum Randić index problem as a minimum weight perfect b-matching problem, set

$$w_{ij} = b_i \cdot b_j. \quad (2)$$

For these weights, solve (1) to obtain $M^*(G)$. We claim that $G^* = (N, M^*(G))$ is an optimal solution to the minimum Randić index problem instance given. Note first that it is feasible since the degree of a node $i \in N$ is $b_i$ by the definition of the perfect b-matching problem. Note 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, (2) 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, we can create an instance of a minimum weight perfect b-matching on a complete graph 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. Note that this method does not enforce connectivity.

Subject to a connectivity constraint, we show that approximating the minimum weight perfect $b$-matching problem on an arbitrary graph $G$ is NP-Hard. This leaves open the complexity of the case when the input graph $G$ is complete and, by equivalence, the connected minimum Randić index problem. We first define approximation algorithms (see [17] for further details about approximation algorithms). Let $S \subset \mathbb{R}^n$ and $f : S \to \mathbb{R}$ be a given feasibility set and objective function, respectively. Define an $\alpha$-approximation algorithm for the minimization problem $v^* = \min_{x \in S} f(x)$ as a polynomial time algorithm that finds a solution $y \in S$ with $f(y) \leq \alpha v^*$. We say that we can approximate a minimization problem if there exists an $\alpha$ such that an $\alpha$-approximation algorithm exists. Note that $\alpha \geq 1$ is implicit with $\alpha = 1$ only if an exact algorithm exists.

**Theorem 3.1.** Approximating the minimum weight perfect $b$-matching problem subject to a connectivity constraint is NP-Hard.

**Proof.** Recall that a Hamiltonian cycle on $G$ is a tour (set of adjacent edges or, equivalently, nodes in $G$) that visits each node exactly once, except for the start node which is equal to the last node on the tour. We claim the existence of a Hamiltonian cycle on a given graph is equivalent to the feasibility of the minimum weight perfect $b$-matching with connectivity on a related instance. Recall that an instance of Hamiltonian cycle consists of a graph, so let such an instance be given with $G = (N, E)$. Now define the vector $b \in \mathbb{R}^{|N|}$ by setting $b_i = 2$ for $i \in \{1, \ldots, |N|\}$ and consider the resulting connected minimum weight perfect $b$-matching instance using the graph $G$ and vector $b$.

We first show that if the minimum weight perfect $b$-matching instance $(G, b)$ is feasible, then there is a Hamiltonian cycle on $G$. Suppose there is a feasible solution $H = (N, F)$, which means $F \subseteq E$, each node $u \in N$ has degree 2, and $H$ is connected. As each node has even degree and $H$ is connected, there is an Eulerian cycle, $T$, on $H$. We claim that $T$ is a Hamiltonian cycle on $G$, which means each node is visited exactly once by $T$ except the start node which is visited exactly twice. If $T$ does not visit some node $u \in N$, this is a contradiction as $T$ traverses every arc and two arcs are adjacent to $u$. Denote the start node of $T$ by $s \in N$ and consider traversing $T$ beginning at $s$. If the traversal visits a node $u \in N \setminus \{s\}$ more then once then an edge was traversed into $u$, a second distinct edge was traversed out of $u$, and a third distinct edge was traversed into $u$, a contradiction as there are exactly two distinct edges adjacent to $u$ in $H$. The same argument applies if $s$ is visited more then once before the traversal is complete. So each node is visited exactly once by $T$ except the start node, which is returned to when the traversal is complete, i.e., $T$ is a Hamiltonian cycle. Thus, if the instance $(G, b)$ is feasible, then $G$ possess a Hamiltonian cycle.

We now show that if there is a Hamiltonian cycle on $G$, then the minimum weight perfect $b$-matching problem on $(G, b)$ is feasible. Consider a Hamiltonian cycle, $C$ on $G$ and the subgraph induced by $C$. Such a subgraph is connected as each node is visited. Also, each node has degree 2 as each node $u \in N$ has one arc used to enter $u$ and exactly one distinct
arc u to exit. Thus, if G possess a Hamiltonian cycle, then \((G, b)\) must be feasible to the given connected minimum weight perfect b-matching instance.

Now suppose there were an \(\alpha\)-approximation algorithm to the connected minimum weight perfect b-matching problem for some \(\alpha \geq 1\). If the algorithm returns a solution to the instance \((G, b)\) then \(G\) possesses a Hamiltonian cycle. If it does not, then the instance \((G, b)\) was not feasible and \(G\) does not possess a Hamiltonian cycle. Note that the argument does not rely on what value \(\alpha\) is. 

### 3.2 Example Transformation

Given the degree sequence \(d = (3, 2, 2, 2, 2, 1)\), what is a graph realization with the minimum Randić index? We let nodes \(v_1, v_2, v_3, v_4, v_5, v_6 \in N\) with \(b = [3 \ 2 \ 2 \ 2 \ 2 \ 1]\). Now we can form the complete graph \(G\), with weights corresponding to \(b_i \cdot b_j\) for every node \(i, j \in N\).

Now we solve the minimum weight perfect b-matching for \(G\) and obtain \(G'\):

\[
\begin{bmatrix}
0 & 6 & 6 & 6 & 6 & 3 \\
6 & 0 & 4 & 4 & 4 & 2 \\
6 & 4 & 0 & 4 & 4 & 2 \\
6 & 4 & 4 & 0 & 4 & 2 \\
6 & 4 & 4 & 4 & 0 & 2 \\
3 & 2 & 2 & 2 & 2 & 0
\end{bmatrix}
\]
$G'$ is a 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.

### 4 Solving the Minimum Randić index Problem

In this section, we focus on the case where the input graph $G$ is the complete graph. To solve the minimum Randić index problem we used code that solves a minimum weight perfect $b$-matching problem. The code used is for generalized matching problems and was written by Vlad Schogolev, Bert Huang, and Stuart Andrews. Their code uses the GOBLIN graph library (http://goblin2.sourceforge.net/). Huang’s paper on loopy belief propagation for bipartite maximum weight $b$-matching uses this code [8]. The code solves a maximum weight perfect $b$-matching problem given the weight matrix, $b$ vector and the exact solution algorithm choice. After transforming our minimum Randić index problem instance into a minimum weight perfect $b$-matching instance, the $b$ vector will correspond to the degree constraints and the weight matrix to the possible degree products (see Section 3.2). But, since the code solves the maximum matching we transform our problem so that solving for the maximum yields the
solution for the minimum.

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 1s in all positions except for the diagonal which has 0s. We then multiply every entry by one more than the maximum entry of $H$. Then $H$ is subtracted from $M$ yielding $H_2$. Now we can state an algorithm that will solve the minimum Randić index problem for a given degree sequence.

**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 Randić 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 3: Solving minimum Randić index with $b$-matching

The algorithm in Figure 3 returns the minimum Randić index of a graph and a realization. We know that the $b$-matching code runs in polynomial time ([15]). It is easy to see that the transformation steps are done in polynomial time as well. We used three types of randomly generated graphs to test the algorithm performance: Erdős-Rényi, geometric and scale-free. We limited our computational experiments to graphs for which connected realizations were known to exist. The Randić index before and after the optimization was recorded. After the optimization we check if the graph realization with the minimum Randić index is connected. We generated graphs with 25, 50 and 100 nodes. In addition, 100 of each graph type and size were generated. Tables 1, 2 and 3 present results from the runs. 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   | 67        | 1            | 32                       |
| Geometric     | 61        | 2            | 37                       |
| Scale-Free    | 93        | 7            | 0                        |

Table 1: 25 node graphs
The MATLAB functions used to generate the geometric and scale-free graphs are from CONTEST: A Controllable Test Matrix Toolbox for MATLAB [16]. 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 [3].

In general from our runs, the realization of the minimum Randić index was connected. There are minimum Randić index graph realizations that are disconnected and we do not know if there are other realizations with this Randić index that are connected since the \( b \)-matching solver only produces one solution. But there were often a large proportion of graphs that had no connected realization at all. This largely depends on parameters chosen for the randomly generated graphs. If the random graph produced has most nodes with large degrees then it is unlikely that any graph realization would be disconnected. We were interested in generating graphs that have both connected and disconnected realizations and investigating whether the realization generated with the minimum Randić index was connected or not.

The following parameters were chosen after extensive experimentation so that the number of instances with no connected realizations was small. For the Erdős-Rényi graphs we used an average degree per node of 4.25. The corresponding \( p \) values used were calculated using \( p = \frac{4.25}{n} \) where \( n \) is the number of nodes in the graph. Thus \( p = .17 \) for \( n = 25 \), \( p = .085 \) for \( n = 50 \), and \( p = .043 \) for \( n = 100 \). For the geometric graphs we used an average degree per node of 6. The radii were calculated using \( r = \sqrt{\frac{6}{\pi n}} \). Our corresponding radii were \( r = .276 \) for \( n = 25 \), \( r = .195 \) for \( n = 50 \), and \( r = .138 \) for \( n = 100 \). We used scale-free graphs with a minimum node degree of 2.

The left box plots for each of 25, 50 and 100 node graphs in Figures 4, 5 and 6 show the percent difference between the graph’s original Randić

| Graph type | connected | disconnected | no connected realizations |
|------------|-----------|--------------|--------------------------|
| Erdős-Rényi | 50        | 5            | 45                       |
| Geometric  | 57        | 3            | 40                       |
| Scale-Free | 85        | 15           | 0                        |

Table 2: 50 node graphs

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

Table 3: 100 node graphs
index and the minimum Randić index. The percent difference is calculated from \( \frac{\text{original} - \text{minimum}}{\text{minimum}} \times 100 \). The right box plots in these figures describe the performance of the heuristic. The right box plots and the heuristic are described in the next section.

![Figure 4: Comparing percent differences for Erdős-Rényi graphs.](image)

![Figure 5: Comparing percent differences for geometric graphs.](image)

5 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 7 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. The right box plots for each of 25, 50 and 100 node graphs in Figures 4, 5 and 6 show the percent difference between the minimum Randić index and the Randić index after the heuristic. This percent difference is calculated with $\frac{\text{after heuristic} - \text{minimum}}{\text{minimum}} \times 100$. 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 Tables 1, 2 and 3 for those 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 $\geq 2$  
do a two switch with components 1 and 2 to connect them  
using two randomly chosen edges from each component

return $A$

Figure 6: Comparing percent differences for scale-free graphs.

Figure 7: 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.

6 Conclusions and Future Work

We have shown that the minimum Randić index problem can be solved in polynomial time. With use of available b-matching code we have developed an algorithm that produces a graph realization with the minimum Randić index for a given degree sequence. Not all optimal solutions are connected. A two-switch heuristic was developed to connect disconnected optimal solutions. Although the graph structure of these new connected graphs is fragile, the Randić index changed relatively little.

From our experiments, when generating Erdős-Rényi, geometric and scale-free graphs, the realization with the minimum Randić index is generally connected. Undoubtedly, this result is influenced by the parameters chosen for the randomly generated graphs, but many graphs that will have a disconnected minimum Randić index realization have no connected realizations at all.

There are a number of future topics to explore. We want to develop a better way to connect graphs using the two-switch heuristic so that the structure of the graph is less fragile. Further experiments with the parameters of randomly generated graphs are needed to understand the conditions under which the number of graphs that are disconnected or have no connected realizations changes. A natural next step is to consider directed graphs and explore the properties of an extension of the Randić index there. We also are interested in determining the complexity of the connected Randić index problem.

References

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

[2] B. Bollobás, P. Erdős, and A. Sarkar. Extremal graphs for weights. *Discrete Mathematics*, 200(1-3):5–19, 1999.

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

[4] I. Gutman and N. Trinajstic. Graph theory and molecular orbitals. total φ-electron energy of alternant hydrocarbons. *Chemical Physics Letters*, 20(3):257–260, 1973.
[5] S.L. 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*, pages 496–506, 1962.

[6] P. Hansen and H. Mélot. Variable neighborhood search for extremal graphs. 6. analyzing bounds for the connectivity index. *Journal of Chemical Information and Computer Sciences*, 43(1):1–14, 2003.

[7] V. Havel. A remark on the existence of finite graphs. *Casopis Pest. Mat.*, 80:477–480, 1955.

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

[9] R.K. Kincaid, N. Alexandrov, and M. Holroyd. An investigation of synchrony in transport networks. *Complexity*, 14(4):34–43, 2009.

[10] R.K. Kincaid and D. Phillips. Network topology measures. *Wiley Interdisciplinary Reviews: Computational Statistics-Operations*, 3(6):557–565, 2011.

[11] L. Li, D. Alderson, J.C. Doyle, and W. Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, 2(4):431–523, 2005.

[12] 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.

[13] M. Randić. Characterization of molecular branching. *Journal of the American Chemical Society*, 97(23):6609–6615, 1975.

[14] H.J. Ryser. Combinatorial properties of matrices of zeros and ones. *Canad. J. Math*, 9(371-377):6, 1957.

[15] A. Schrijver. *Combinatorial optimization*. Springer, 2003.

[16] A. Taylor and D.J. Higham. Contest: A controllable test matrix toolbox for matlab. *ACM Transactions on Mathematical Software (TOMS)*, 35(4), 2009.

[17] V. V. Vazirani. *Approximation algorithms*. Springer-Verlag, Berlin, 2001.

[18] D.B. West. *Introduction to graph theory, 2nd ed.* Prentice Hall Upper Saddle River, NJ, 2001.

16