

On the Conduciveness of Random Network Graphs for Maximal Assortative or Maximal Dissortative Matching

Natarajan Meghanathan¹

¹ Department of Computer Science, Jackson State University, USA

Correspondence: Natarajan Meghanathan, Department of Computer Science, Mailbox 18839, Jackson State University, Jackson, MS 39217, USA. Tel: 1-601-979-3661. E-mail: natarajan.meghanathan@jsums.edu

Received: October 25, 2015

Accepted: November 20, 2015

Online Published: December 15, 2015

doi:10.5539/cis.v9n1p21

URL: <http://dx.doi.org/10.5539/cis.v9n1p21>

The research is financed by the Massie Chair Grant (#: DE-NA0000654) at Jackson State University.

Abstract

A maximal matching of a graph is the set of edges such that the addition of an edge to this set violates the property of matching (i.e., no two edges of the matching share a vertex). We use the notion of assortative index (ranges from -1 to 1) to evaluate the extent of similarity of the end vertices constituting the edges of a matching. A maximal matching of the edges whose assortative index is as close as possible to 1 is referred to as maximal assortative matching (MAM) and a maximal matching of the edges whose assortative index is as close as possible to -1 is referred to as maximal dissortative matching (MDM). We present algorithms to determine the MAM and MDM of the edges in a network graph. Through extensive simulations, we conclude that random network graphs are more conducive for maximal dissortative matching rather than maximal assortative matching. We observe the assortative index of an MDM on random network graphs to be relatively more closer to the targeted optimal value of -1 compared to the assortative index of an MAM to the targeted optimal value of 1.

Keywords: assortative index, maximal assortative matching, maximal dissortative matching, random network graphs

1. Introduction

Maximal matching is one of the classical problems of graph theory. A matching for a graph is a set of edges such that no two edges share a common vertex. A maximal matching for a graph is a set of edges such that the inclusion of an additional edge to this set violates the above property of matching. A maximum matching for a graph is the largest maximal matching for a graph. Most of the algorithms in the literature for determining a maximal matching for a graph use the notion of augmenting paths (Cormen et al., 2009) and the focus is on determining the maximal node matching (i.e., largest set of edges such that no two edges have a common end vertex). The extent of the similarity of the end vertices constituting the edges of a maximal matching has not been typically taken into consideration as part of the design of the maximal matching algorithms. In this paper, we emphasize that the similarity of the end vertices of the edges constituting a maximal matching needs to be taken into consideration in the design of matching algorithms for complex network graphs. This is because, when vertices in a complex network graph are to be paired to obtain a maximal matching, it is imperative that we pair vertices that are either very similar or very dissimilar from each other (depending on the nature of the application). For example, a maximal matching that is arbitrary with respect to the similarity of the vertices being matched need not be preferred in peer-to-peer interaction and collaborative networks: Students would prefer to be paired with students of similar characteristics when they are asked to share a dormitory, whereas faculty would prefer to collaborate with peers who have different strengths and could complement each other on a research project. Thus, we hypothesize that for complex network graphs, maximal matching of the vertices in the network graph should be conducted with the objective of maximizing either the similarity or dissimilarity of the end vertices constituting the matching and should not be arbitrary or done simply with the objective of maximizing the number of vertices paired.

We propose to use the notion of assortative index (Newman, 2010) as a measure of the similarity of the end vertices constituting a maximal matching. The assortativity index of a set of edges (with respect to any particular

measure of node weight - like the node degree) is a quantitative measure of the similarity between the end vertices of the edges that are part of the set (Newman, 2010). The assortativity index values can range from -1 to 1. If the assortativity index of a set of edges calculated with respect to a particular measure of node weight is close to 1, then it implies the end vertices of the edges that form the set are very similar to each other with respect to the particular measure of node weight (for example, a high degree vertex matched to another high degree vertex, a low degree vertex matched to another low degree vertex, etc). If the assortativity index is close to 0, then the pairing of the vertices in the edge set is arbitrary with respect to the node weight. On the other hand, if the assortativity index of the set of the edges with respect to a measure of node weight is close to -1, then it implies that most of the node pairs constituting the edge set are not similar to each other with respect to the node weight (for example, if node degree is used as the node weight, then an assortativity index of -1 of a set of edges implies that most of the node pairings in this set involve a high degree vertex matched to a low degree vertex and vice-versa).

Random network graphs are a category of complex network graphs for which there exists an edge between *any* two vertices with a probability. To vindicate this characteristic, we observe the assortative index of the entire set of edges in a random network graph to be close to 0, indicating that there is no particular preference for a node to have an edge to any other node. Given the above observation, we wanted to explore whether it would be indeed possible to determine a maximal matching of the edges in a random network graph such that the assortative index of the matching is either close to 1 or -1 (depending on whether the targeted matching is an assortative matching or disassortative matching). In this paper, we show that it is possible to determine both maximal assortative matching (MAM) and maximal disassortative matching (MDM) on random network graphs. Further, we make an interesting observation that the assortative index of an MDM is relatively more closer to the targeted optimal value of -1 compared to the proximity of the assortative index of an MAM to the targeted optimal value of 1. We observe the above phenomenon for random network graphs generated with the well-known Erdos-Renyi model (for various fixed values of the probability of a link between any two nodes) with node degree as node weights and random node weights.

The rest of the paper is organized as follows: Section 2 presents the algorithms for maximal assortative matching (MAM) and maximal disassortative matching (MDM). Section 3 presents the results of the execution of the MAM and MDM algorithms on random network graphs generated according to the Erdos-Renyi model with the node degree as node weights and random node weights. Section 4 discusses related work on maximal matching and Section 5 concludes the paper. Throughout the paper, the terms 'node' and 'vertex', 'link' and 'edge' as well as 'pair' and 'match' are used interchangeably. They mean the same.

2. Algorithms for Maximal Assortative Matching and Maximal Disassortative Matching

2.1 Network Model and Assumptions

We model the input network graph $G = (V, E)$ as a set of vertices V and undirected edges E wherein each vertex $v \in V$ have a weight $w(v) \in \mathbb{R}$. We say an edge (p, q) is adjacent to an edge (r, s) if $p, q, r, s \in V$ and either $p = r$ or $p = s$ or $q = r$ or $q = s$. That is, two edges (p, q) and (r, s) are said to be adjacent to each other if they have one common end vertex. The degree of a vertex $u \in V$ is the number of edges incident on u (i.e., the number of edges that have vertex u as one of the two end vertices). Though the edges are undirected, for the sake of discussion, we refer to the first vertex (vertex u) indicated in an edge (u, v) as the upstream vertex and the second vertex (vertex v) indicated in an edge (u, v) as the downstream vertex. Also, since the edges are undirected, we conveniently adopt a convention to represent the edges: the ID of the upstream vertex of an edge (u, v) is always less than the ID of the downstream vertex of the edge (i.e., $u < v$).

For a set of edges M constituting a matching of the vertices V in the graph G , the assortativity index of M is a quantitative measure of the similarity (or equivalently the dissimilarity) of the end vertices of the edges in M (Newman, 2010). The assortativity index for a set M of edges (AI_M) with respect to the node weights $w(v)$ for every vertex $v \in V$ is calculated using the formula (1) given below, where \bar{U} and \bar{D} are respectively the average weight of the upstream and downstream vertices of the edges constituting the set M .

$$AI_M = \frac{\sum_{(p,q) \in M} [w(p) - \bar{U}][w(q) - \bar{D}]}{\sqrt{\sum_{(p,q) \in M} [w(p) - \bar{U}]^2} \sqrt{\sum_{(p,q) \in M} [w(q) - \bar{D}]^2}} ; \quad \bar{U} = \frac{1}{|M|} \sum_{(p,q) \in M} w(p) ; \quad \bar{D} = \frac{1}{|M|} \sum_{(p,q) \in M} w(q) \quad (1)$$

We say an edge (u, v) included in a matching covers itself as well as covers the edges that are adjacent to it in the original graph. An uncovered edge is an edge in the graph that is not yet covered by an edge in the matching. We define the assortativity weight of an edge (u, v) to be the product of the number of uncovered edges that are

adjacent to it and the absolute value of the difference in the weights of the end vertices u and v . The number of uncovered edges adjacent to an edge (u, v) is the sum of the number of uncovered edges incident on each of the end vertices u and v . If $w(u)$ and $w(v)$ indicate respectively the weights of the vertices u and v , and $ue(u)$ and $ue(v)$ indicate respectively the number of uncovered edges incident on vertices u and v , then: Assortativity Weight of edge $(u, v) = \{ue(u) + ue(v)\} * \{|w(u) - w(v)|\}$.

2.2 Description of the Algorithm for Maximal Assortative Matching

The MAM algorithm employs a greedy strategy; at the beginning of each iteration, the algorithm chooses the uncovered edge with the smallest assortativity weight. The pseudo code for the algorithm to determine maximal assortative matching (MAM) is outlined in Figure 1; the pseudo code for the two sub routines used in the algorithm is given in Figure 2. The algorithm maintains the set of uncovered edges (*UncoveredEdges*) that are yet to be covered by an edge in the MAM. The set *UncoveredEdges* is initialized to the set of all edges E for the input graph G .

Input: Graph $G = (V, E)$, where weight $w(v) \in \mathbb{R}$ for every $v \in V$
Output: Maximal Assortativity Matching, *MAM*
Auxiliary Variables: *UncoveredEdges*
Initialization: *UncoveredEdges* = E ; *MAM* = ϕ
Begin MAM Algorithm
1 *FindAssortativityWeights*(*UncoveredEdges*)
2 **while** (*UncoveredEdges* $\neq \phi$) **do**
3 $Edge(u, v) = \left\{ (p, q) \mid \underset{(p, q) \in UncoveredEdges}{Min} \{ AssortativityWeight(p, q) \} \right\}$
4 *MAM* = *MAM* $\cup \{ (u, v) \}$
5 *UncoveredEdges* = *UncoveredEdges* - $\{ (u, v) \}$
6 *RemoveEdges*((u, v) , *UncoveredEdges*)
7 *FindAssortativityWeights*(*UncoveredEdges*)
8 **end while**
9 **return** *MAM*
End MAM Algorithm

Figure 1. Pseudo code for the maximal assortativity matching (MAM) algorithm

To start with, the assortativity weight of the edges in the set *UncoveredEdges* is determined and the edge (u, v) that has the smallest assortativity weight among the edges in *UncoveredEdges* is selected for inclusion in the MAM. An edge (u, v) selected for inclusion to the MAM is said to cover itself as well as cover its adjacent edges; accordingly, all these newly covered edges are removed from the set *UncoveredEdges*. The assortativity weight of the edges in the updated set of *UncoveredEdges* is recalculated and the edge with the smallest assortativity weight is selected for inclusion in the MAM. The above procedure is repeated as a sequence of iterations until the set of *UncoveredEdges* is empty. At this stage, we have found a maximal matching of the vertices in the graph.

The run-time complexity of the MAM algorithm depends on the time complexity to update the set of *UncoveredEdges* in each iteration. As the algorithm proceeds, with each edge added to the MAM, we expect the size of the set of *UncoveredEdges* to reduce significantly. For optimal run-time, we suggest maintaining the set of *UncoveredEdges* as a minimum heap (Cormen et al., 2009) that can be constructed in $O(|E|)$ time for the $|E|$ edges of the graph. Each update to the minimum heap (like removing an edge or updating the assortativity weight of an edge) takes $O(\log|E|)$ time. The MAM algorithm runs at most for $|V|/2$ iterations for a graph of $|V|$ vertices. During each such iteration, there would have to be at most $|E|$ updates to the heap (one update or removal for each edge, depending on the case), incurring a worst-case time complexity of $O(|E| * \log|E|)$ per iteration. Considering that there could be at most $|V|/2$ iterations, the overall run-time complexity of the MAM algorithm is $O(|E| * |V| * \log|E|)$. For sparse graphs ($|E| = O(|V|)$), the run-time complexity of the MAM algorithm would be $O(|V|^2 * \log|V|)$; for dense graphs ($|E| = O(|V|^2)$), the run-time complexity of the MAM algorithm would be $O(|V|^3 * \log|V|)$.

```

Subroutine FindAssortativityWeights(UncoveredEdges)
  for every edge  $(u, v) \in \text{UncoveredEdges}$  do
     $\text{numUncoveredAdjacentEdges}(u, v) = 0$ 
    for every edge  $(u, p)$  or  $(p, u) \in \text{UncoveredEdges}$  AND  $p \neq v$  do
       $\text{numUncoveredAdjacentEdges}(u, v) = \text{numUncoveredAdjacentEdges}(u, v) + 1$ 
    end for
    for every edge  $(v, q)$  or  $(q, v) \in \text{UncoveredEdges}$  AND  $q \neq u$  do
       $\text{numUncoveredAdjacentEdges}(u, v) = \text{numUncoveredAdjacentEdges}(u, v) + 1$ 
    end for

     $\text{AssortativityWeight}(u, v) = \text{numUncoveredAdjacentEdges}(u, v) * |w(u) - w(v)|$ 

  end for
End Subroutine
|
Subroutine RemoveEdges(Edge  $(u, v)$ , UncoveredEdges)
  for every edge  $(u, v) \in \text{UncoveredEdges}$  do
    for every edge  $(u, p)$  or  $(p, u) \in \text{UncoveredEdges}$  AND  $p \neq v$  do
       $\text{UncoveredEdges} = \text{UncoveredEdges} - \{(u, p) \text{ or } (p, u)\}$ 
    end for
    for every edge  $(v, q)$  or  $(q, v) \in \text{UncoveredEdges}$  AND  $q \neq u$  do
       $\text{UncoveredEdges} = \text{UncoveredEdges} - \{(v, q) \text{ or } (q, v)\}$ 
    end for
  end for
End Subroutine

```

Figure 2. Pseudo code for the subroutines used by the maximal assortativity matching algorithm

2.3 Description of the Algorithm for Maximal Dissortative Matching

The MAM algorithm has to be only slightly modified to determine an MDM: Instead of preferring to include edges with a lower assortative weight (to maximize the assortative index of the maximal matching), we need to include the uncovered edge with the largest assortative weight (to minimize the assortative index of the maximal matching) in each iteration. The definition of the assortative weight remains the same as before: that is, the assortative weight of an uncovered edge (u, v) is the product of the number of uncovered edges adjacent to (u, v) and the absolute value of the difference in the node weights for the end vertices u and v . The pseudo code for the MDM algorithm to minimize the assortative index is shown in Figure 3. The sub routines *FindAssortativeWeights* and *RemoveEdges* remain the same as before (see Figure 2). For the MDM algorithm, the set *UncoveredEdges* will have to be maintained as a maximum heap; nevertheless, the time complexity to construct a maximum heap of $|E|$ edges would be $O(|E|)$ and the time complexity to update a maximum heap of at most $|E|$ edges during an iteration would be $O(|E| * \log|E|)$. As a result, the overall run-time complexity of the MDM algorithm would be the same as that of the MAM algorithm: $O(|V|^2 * \log|V|)$ for sparse graphs and $O(|V|^3 * \log|V|)$ for dense graphs.

Input: Graph $G = (V, E)$, where weight $w(v) \in \mathbb{R}$ for every $v \in V$
Output: Maximal Dissortativity Matching, MDM
Auxiliary Variables: $UncoveredEdges$
Initialization: $UncoveredEdges = E$; $MDM = \phi$
Begin MDM Algorithm

- 1 *FindAssortativityWeights*($UncoveredEdges$)
- 2 **while** ($UncoveredEdges \neq \phi$) **do**
- 3 $Edge(u, v) = \left\{ (p, q) \mid \underset{(p, q) \in UncoveredEdges}{Max} [AssortativityWeight(p, q)] \right\}$
- 4 $MDM = MDM \cup \{(u, v)\}$
- 5 $UncoveredEdges = UncoveredEdges - \{(u, v)\}$
- 6 *RemoveEdges*($(u, v), UncoveredEdges$)
- 7 *FindAssortativityWeights*($UncoveredEdges$)
- 8 **end while**
- 9 **return** MDM

End MDM Algorithm

Figure 3. Pseudo code for the maximal dissortative matching (MDM) algorithm

2.4 Example for MAM and MDM

Figure 4 presents an example to illustrate the execution of the maximal assortative matching algorithm on a graph wherein the node weights are random numbers generated in the range 0 to 1. All the edges in the input graph and the initialization graph are uncovered edges. The initialization graph displays the assortative weight of the edges as a tuple. For an edge (u, v) , we indicate a tuple representing (number of uncovered adjacent edges and the absolute value of the difference in the node weights of the end vertices u and v) as well as the assortativity weight of the edge, which is the product of the two entries in the tuple. In the first iteration, the algorithm encounters a tie between edges (3, 6) and (4, 7) - both of which have the lowest assortative weight of 0.6; the algorithm breaks the tie arbitrarily by including edge (3, 6) to the maximal assortative matching (MAM). As part of the inclusion of the edge (3, 6) into the MAM, all its adjacent edges are considered to be covered and are removed from the graph. We reevaluate the assortativity weight of the uncovered edges in the graph; edge (4, 7) with the currently lowest assortative weight of 0.3 is the second edge to be picked for inclusion to the MAM and all its adjacent edges are removed from the graph. At the end of the second iteration, all edges in the graph are either in the MAM or covered by an edge in the MAM. The node weights of the end vertices that are included into the MAM are (0.9, 0.8) and (0.3, 0.4) for the edges (3, 6) and (4, 7) respectively. The difference in the node weights of the end vertices for both the edges in the MAM is the bare minimum that we could get for the input graph considered (as one can notice, all the nodes in the input graph have unique weights). The % of nodes matched in the MAM is $(2*2)/7 = 57\%$ and the assortative index of the matching (based on node weights) is 1.0; the calculations are illustrated as part of Figure 4.

Figure 5 presents an example for the execution of the MDM algorithm on the same graph as in Figure 4. The initial values for the assortative weight of the edges are the same as in Figure 4. As the MDM algorithm prefers to include edges with a larger assortative weight, the first edge to get selected for inclusion to the MDM is edge (5, 6) of assortativity weight 3.5. Due to this inclusion, all the five adjacent edges of (5, 6) are considered to be covered and are removed from the graph. In the second iteration, the MDM algorithm chooses the edge (3, 4) with the currently largest assortativity weight (2.4) to be part of the matching; as a result of this inclusion, all the four adjacent edges of the edge (3, 4) are considered to be covered. At the end of the second iteration, the edges

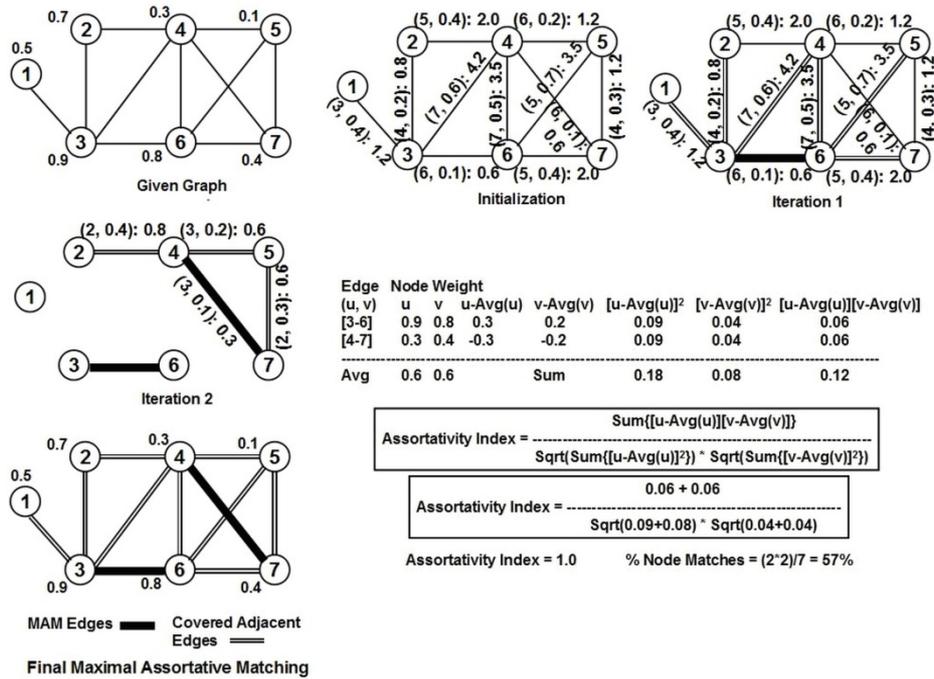


Figure 4. Example to illustrate the execution of the algorithm for maximal assortative matching

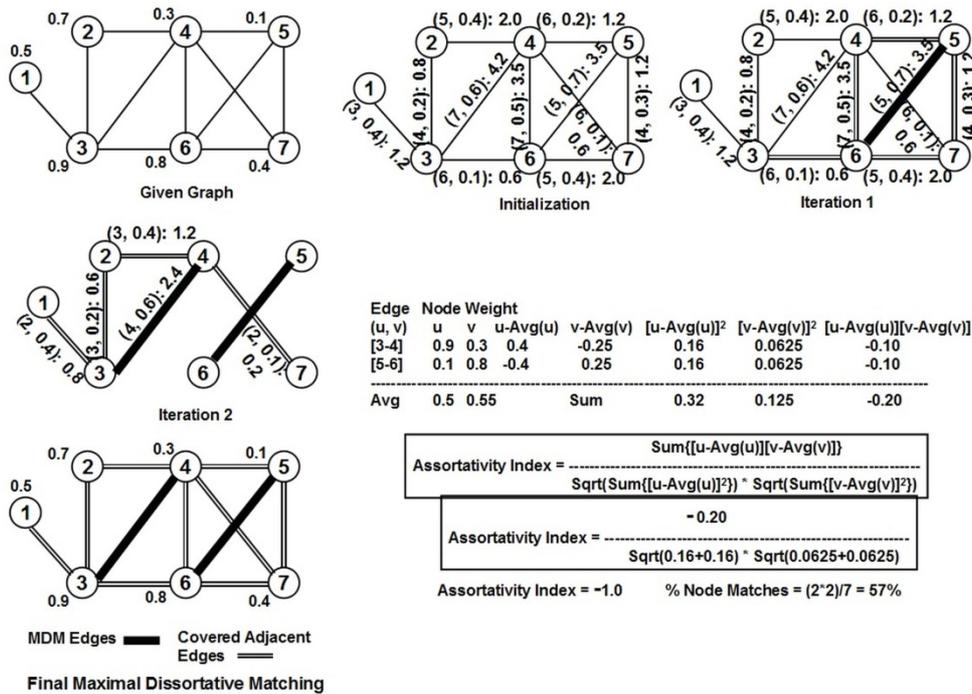


Figure 5. Example to illustrate the execution of the algorithm for maximal dissortative matching

In the graph are either covered or included to the MDM. One can notice that the weights of the pairs of vertices constituting the edges (3, 4) and (5, 6) that are part of the MDM are respectively (0.9, 0.3) and (0.1, 0.8). Given the choice of weights of the end vertices in the toy example graph, this is the best combination of edges that one could come up with even after trying all permutations and combinations. The assortativity index of the above MDM matching evaluates to -1, vindicating our assertion. Like the case of MAM, the % of node matches could not be maximized and only 2 of the 3 edges could be selected (the maximum number of edges that could be selected for a matching in the chosen toy example of 7 vertices is 3 edges).

3. Analysis of Random Network Graphs

In this section, we simulate the evolution of random network graphs generated using the well-known Erdos-Renyi model (Erdos & Renyi, 1959). The model inputs two parameters: the total number of nodes (N) and the probability of a link (p_{link}) between any two nodes in the graph. Since we simulate the evolution of an undirected random network, the links are bi-directional and we could assume that the end vertices of each link could be represented as an ordered pair (u, v) where u and v are the node IDs and $u < v$. We assume there are no self-loops and there is no more than an edge between any two nodes in the network. For an N node network, the maximum number of undirected links possible in the network is $N(N-1)/2$. We consider every such possible link in the network and generate a random number (in the range 0...1) to decide whether or not to include the link in the network. If the random number generated for a pair (u, v) is less than or equal to p_{link} , then we include the link (u, v) in the network; otherwise, not. As it is obvious, the larger the value of p_{link} , the larger the number of links in the random network graph as well as larger the chances for the network to have a degree distribution wherein the degree of each node is closer to the average node degree. The total number of nodes considered in the simulations for this section is $N = 100$ nodes. The values used for the probability of link between any two nodes in the network (p_{link}) are: 0.05, 0.07, 0.10, 0.15, 0.20, 0.30, 0.40 and 0.50. We observe that the random networks for all the 100 trials generated with $p_{link} \geq 0.05$ to be connected. Even though the number of links in the network increases with increasing p_{link} values, the assortativity of the set of all edges in a random network remains close to 0 for all the p_{link} values. This vindicates the random nature of the distribution of the edges among the vertices as per the Erdos-Renyi model.

In section 3.1, we use node degree as node weight for the assortativity calculations and in section 3.2, we use randomly generated numbers in the range (0...1) as node weight for the assortativity calculations. Accordingly, for each of the two sections, we run 100 trials of the network evolution and analyze the assortativity of the network as well as evaluate the % of node matches and assortativity index (both the average and standard deviation) of the maximal matching obtained with the MAM and MDM algorithms for each p_{link} value. The values reported for the % of node matches and assortative index in Figures 6 and 7 are the average (the values corresponding to the markers) and standard deviation (the values corresponding to the error bars) obtained from the 100 trial runs for each p_{link} value. We observe the variation in the assortative index and % of node matches to decrease with increase in the p_{link} values.

3.1 Analysis with Node Degree as Node Weights

The values reported in Figure 6 are the average values obtained from the 100 trial runs for each p_{link} value with node degree considered as node weights. With regards to the % of node matches, the % of node matches for the MAM and MDM algorithms are respectively 85% and 84% for p_{link} value of 0.05, and reaches respectively 99% and 98% for p_{link} value of 0.5; the % of node matches for MAM crosses 95% when p_{link} is 0.15 and the % of node matches for the MDM crosses 95% when p_{link} is 0.20. However, the tradeoff is quite high with respect to the assortativity index (A.Index).

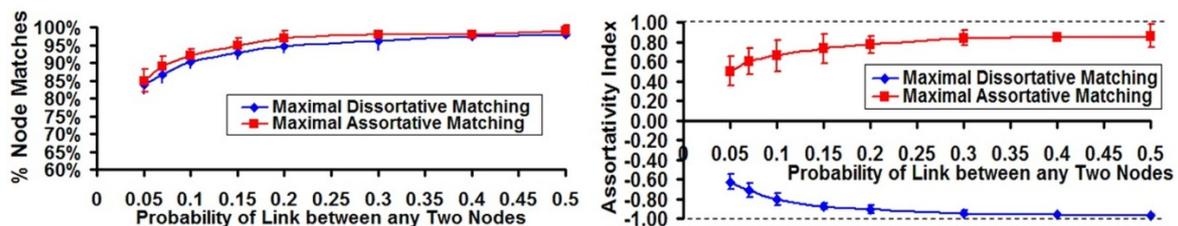


Figure 6. Random networks with node degree as node weights: Distribution of the percentage of node matches and assortativity index as a function of the probability of link between any two nodes

The A.Index values of MAM and MDM are 0.5 and -0.63 when p_{link} is 0.05 and reaches 0.85 and -0.96 when p_{link} value is 0.3. The A.Index does not increase appreciably for MAM as we further increase the p_{link} value. The average A.Index value observed for MAM is 0.90 when the p_{link} value is 0.90. On the other hand, the A.Index values for the MDM continue to vary (i.e., decrease) at a relatively faster rate and reaches -0.99 when the p_{link} value is 0.90. This is a significant observation that has been hitherto not reported in the literature for random networks. Figure 6 illustrates the nature of increase in the % of node matches and the assortativity index values as we increase the p_{link} values from 0.05 to 0.50 as explained above.

Overall, the results presented for random network graphs with node degree as node weights illustrate that the assortative index values obtained with Maximal Dissortative Matching (MDM) are more closer to the targeted optimal value (-1) compared to the closeness of the assortative index values obtained with the Maximal Assortative Matching (MAM) to the targeted optimal value (1). On the other hand, though the % of node matches obtained with Maximal Dissortative Matching appears to be less than that obtained with the Maximal Assortative Matching, the difference in the % of node matches is within 2-3% for all values of p_{link} and by observing the nature of the increase in the % of node matches with the two maximal matching strategies (MAM and MDM), we could say that the difference in the % of node matches would only further narrow down with increase in the p_{link} value. The results of Figure 6 thus illustrate that when node degree is considered for node weights, random network graphs are more conducive for a maximal dissortative matching compared to a maximal assortative matching on the basis of the proximity of the assortative index to the targeted optimal value (-1 for MDM and +1 for MAM).

3.2 Analysis with Random Node Weights

In this sub section, we present the results for the percentage of node matches and assortativity index incurred with the MAM and MDM algorithms for random networks generated under the Erdos-Renyi model wherein the node weights are random numbers generated from 0 to 1. We conducted the simulations with 100 trials for each p_{link} value and averaged the results for the network assortativity as well as the % of node matches and assortativity index for the MAM and MDM. The results presented in Figure 7 indicate the average values for these metrics from the 100 trials.

For a given p_{link} value, we observe the assortativity index of the maximal assortative matching (with random node weights) to be slightly higher (the difference is as large as 0.1 in a scale of 0 to 2) than the assortativity index of the maximal assortative matching with node degree as node weights. Though the difference in the assortativity index values for maximal assortative matching with the above two categories of node weights could be observed for all p_{link} values, the difference is relatively more prominent for random networks with lower p_{link} values and reduces as the p_{link} value increases. As can be observed from Figure 7, the curve for the assortativity index for maximal assortative matching with random node weights becomes flat starting from p_{link} value of 0.40 (the assortativity index curve for the MAM with node degree as node weights became flat starting from p_{link} value of 0.30).

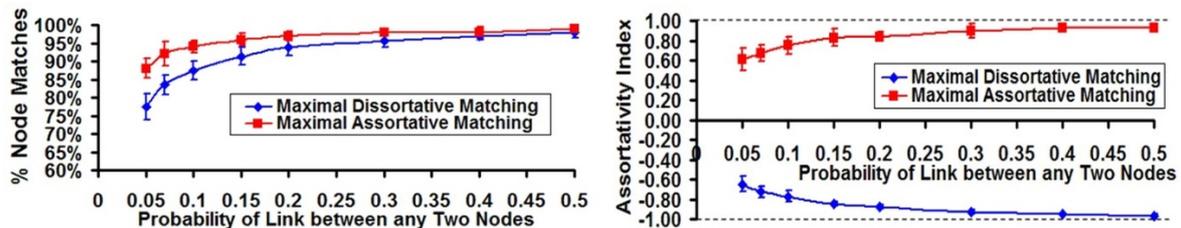


Figure 7. Random networks with random node weights: Distribution of the percentage of node matches and assortativity index as a function of the probability of link between any two nodes

An interesting observation is that (in addition to incurring a relatively larger assortativity index) the % of node matches obtained with the MAM algorithm for random network graphs with random node weights is even slightly larger than the % of node matches obtained with the MAM algorithm for random network graphs with node degree as node weights, especially for networks formed with lower p_{link} values. Overall, the maximal assortative matching algorithm could give even relatively better optimal results (with respect to both assortativity index and % of node matches) for random network graphs with random node weights and the tradeoff in the values incurred for the above two metrics is relatively less pronounced than what is observed in random network graphs with node degree as node weights. As we expect node weights in social networks to be not only a measure of the node degree, the MAM algorithm could be very useful to match vertices with any measure of node weights, especially in social network graphs that are not very dense. This vindicates the wider scope of application of the proposed maximal assortative matching (MAM) algorithm; the algorithm could give even better optimal results (with respect to assortativity) for random graphs with node weights that are independent of node degree.

With regards to the performance of the MDM algorithm on random network graphs (that evolved using the

Erdos-Renyi model) with randomly generated node weights in the range 0 to 1, we observe (from Figure 7) the assortativity index values for the maximal assortative matching to be very close to that of the assortativity index values illustrated in Figure 6 for the maximal assortative matching obtained on random network graphs with node degree as node weights (the difference in A.Index is within ± 0.03); the % of node matches obtained for the maximal assortative matching with random node weights is at most 7% lower than that obtained for the maximal assortative matching with node degree as node weights. While comparing the results obtained for the maximal assortative matching and maximal assortative matching obtained on random network graphs with random node weights, we observe the assortativity index values of the maximal assortative matching to be relatively more closer to the targeted optimal value of -1 compared to that of the closeness of the assortativity index values of the maximal assortative matching to its targeted optimal value of 1. Thus (like in the case of random network graphs with node degree as node weights), we could still say that for random network graphs with random node weights, it would be more apt to aim for a maximal assortative matching compared to a maximal assortativity matching on the basis of the proximity of the assortative index to the targeted optimal value.

4. Related Work

The problem of determining a maximal matching with minimum cardinality for the set of edges constituting the matching is an NP-hard problem (Yannakakis & Gavril, 1980). It is equivalent to the problem of finding a minimum edge dominating set (Horton & Kilakos, 1993) - to find the smallest set of edges of the graph such that each edge in the set covers itself and covers one or more adjacent edges as well as satisfies the matching constraint (no two edges in the set have a common end vertex). The problem of focus in our paper is the maximal independent edge set problem (Cormen et al., 2009) wherein we want to find the largest set of independent edges such that no two edges have a common end vertex. Note that heuristics (Horton & Kilakos, 1993) for the minimum edge dominating set problem cannot be applied to determine the maximal a(di)assortative matching. Because, heuristics for the minimum edge set problem are more likely to determine the set of edges such that each edge in the set covers a larger number of adjacent edges. The maximal matching algorithms developed in this paper take the approach of preferring to include edges that cover a smaller number of adjacent edges so that the number of independent edges determined could be as large as possible. To the best of our knowledge, we have not come across a maximal matching algorithm that is aimed at simultaneously maximizing the a(di)assortativity of the matching as well as maximizing the cardinality of the matching for complex network graphs. In this perspective, the maximal assortative matching and maximal assortative matching algorithms proposed in this paper are significant contributions to the literature for complex network graphs and analysis.

Wang et al (2011) showed that for networks with binomial degree distribution, the maximum and minimum assortativity vary with the density of the networks. Motivated by this observation, Winterbach et al (2012) introduced an algorithm to compute a network with maximal or minimal assortativity given a vector of valid node degrees using degree-preserving rewiring (Maslov & Sneppen, 2002) and weighted b-matching (Muller-Hannemann & Schwartz, 1999). Degree-preserving link rewiring is effective in decreasing or increasing the assortativity of a network graph without affecting the degree distribution of the vertices (Van Mieghem et al., 2010). Holme & Zhao (2007) also found that an increase in the assortativity of a graph (accomplished through degree-preserving rewiring) also contributes to an increase in the maximum modularity, average hop count, effective graph resistance as well as a decrease in the number of clusters. However, neither the work of Wang et al (2011) nor the work of Winterbach et al (2012) could be extended to determine a maximal a(di)assortative matching of the edges of the graph. It was also shown by Wang et al (2011) that for networks whose degree distribution is binomial (like the Erdos-Renyi model-based random network graphs), the maximum assortativity and minimum assortativity are asymptotically anti-symmetric. This observation correlates well with our observation in Section 3 that the values for the assortative index for maximum assortative matching are comparable enough to the absolute values of the assortative index for maximum assortative matching.

5. Conclusions

We explored the feasibility of determining maximal matching in random network graphs with the objective of maximizing the assortative index (targeted optimal value of 1) or minimizing the assortative index (targeted optimal value of -1). In this pursuit, we have developed two greedy strategy-based maximal matching algorithms for maximizing the assortative index (maximal assortative matching, MAM) and minimizing the assortative index (maximal assortative matching, MDM). We showed that it is possible to determine maximal assortative matching as well as maximal assortative matching for random network graphs, despite the observation that the assortative index of the set of all edges of a random network graph is 0. For a probability of link value of 0.30 or above, we observe that it is possible to determine maximal assortative matching as well as maximal assortative

matching with assortative index values equal to or above 0.85 and equal to or below -0.95 respectively. With regards to the relative proximity of the assortative index values to the targeted optimal values, we observe the assortative index of a maximal dissortative matching for a random network graph to be consistently more closer to the targeted optimal value of -1 (in comparison to the proximity of the assortative index values of a maximal assortative matching to the targeted optimal value of 1). We could thus conclude that random network graphs are more conducive for maximal dissortative matching. The methodology and the algorithms described in this paper could also be used to decide the conduciveness of other types of complex network graphs (like scale-free network graphs (Barabasi & Albert, 1999) as well as real-world network graphs (Krebs, 2000; Zachary, 1977) for maximal assortative or maximal dissortative matching. We intend to study this as part of future work.

Acknowledgments

The research is financed by the Massie Chair Grant (#: DE-NA0000654) at Jackson State University.

References

- Barabasi, A. L., & Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, 286(5439), 509-512. <http://dx.doi.org/10.1126/science.286.5439.509>
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3rd ed.) MIT Press.
- Erdos, P., & Renyi, A. (1959). On Random Graphs I. *Publicationes Mathematicae*, 6, 290-297.
- Holme, P., & Zhao, J. (2007). Exploring the Assortativity-Clustering Space of a Network's Degree Sequence. *Physics Review E*, 75(4), 1-12. <http://dx.doi.org/10.1103/PhysRevE.75.046111>
- Horton, J. D., & Kilakos, K. (1993). Minimum Edge Dominating Sets. *SIAM Journal on Discrete Mathematics*, 6(3), 375-387. <http://dx.doi.org/10.1137/0406030>
- Krebs, V. (2000). Working in the Connected World: Book Network. *Journal of the Institute of Health Record Information and Management*, 4(1), 87-90.
- Maslov, S., & Sneppen, K. (2002). Specificity and Stability in Topology of Protein Networks. *Science*, 296(5569), 910-913. <http://dx.doi.org/10.1126/science.1065103>
- Muller-Hannemann, M., & Schwartz, A. (1999). Implementing Weighted b-Matching Algorithms: Insights from a Computational Study. *Algorithm Engineering and Computation, Lecture Notes in Computer Science*, 1619, 18-36. http://dx.doi.org/10.1007/3-540-48518-X_2
- Newman, M. E. J. (2010). *Networks: An Introduction* (1st ed.) Oxford University Press.
- Van Mieghem, P., Wang, H., Ge, X., Tang, S., & Kuipers, F. (2010). Influence of Assortativity and Degree-Preserving Rewiring on the Spectra of Networks. *The European Physical Journal B*, 76(4), 643-652. <http://dx.doi.org/10.1140/epjb/e2010-00219-x>
- Wang, H., Winterbach, W., & Van Mieghem, P. (2011). Assortativity of Complementary Graphs. *The European Physical Journal B*, 83(2), 203-214. <http://dx.doi.org/10.1140/epjb/e2011-20118-x>
- Winterbach, W., de Ridder, D., Wang, H. J., Reinders, M., & Van Mieghem, P. (2012). Do Greedy Assortativity Optimization Algorithms Produce Good Results? *The European Physical Journal B*, 5, 151-160. <http://dx.doi.org/10.1140/epjb/e2012-20899-2>
- Yannakakis, M., & Gavril, F. (1980). Edge Dominating Sets in Graphs. *SIAM Journal on Applied Mathematics*, 38(3), 364-372.
- Zachary, W. W. (1977). An Information Flow Model for Conflict and Fission in Small Groups. *Journal of Anthropological Research*, 33(4), 452-473.

Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).