

Community Structure Detection from Networks with Weighted Modularity

Nandinee Fariah Haq^a, Mehdi Moradi^b, Z. Jane Wang^a

^aDepartment of Electrical and Computer Engineering, University of British Columbia, Vancouver, Canada

^bIBM Almaden Research Center, San Jose, USA

Published in *Pattern Recognition Letters* 2019

DOI : <https://doi.org/10.1016/j.patrec.2019.02.005>

ABSTRACT

Community detection from networks is an emerging topic in modern network science. Communities are defined as clusters of nodes or vertices that share higher concentration of edges among themselves than sharing with other nodes in the network. Community structure is an important property of real systems and detecting communities enables us to better understand the underlying structure of the system. The most widely used method for community detection is modularity maximization which works by optimizing a quality function named modularity of the network partition. However, traditional modularity-based approaches generally have a resolution limit that prevents them from detecting communities that are sufficiently smaller compared to the whole network. In this work, we target to overcome the resolution limit of the modularity function by incorporating a weight term in the modularity formulation. We propose a community detection approach based on a community quality metric, named as *weighted modularity*. We validate the performance of the proposed method in several benchmark networks and show that the proposed method is promising in different settings.

1. Introduction

The study of networks has received much attention over the past few years and many researches currently focus on developing and applying algorithms targeted to analyze network structure in the fields of natural, social and computer sciences. Networks consist of nodes or vertices and edges or links, where typically an edge connects a pair of nodes. An important feature of the networks representing real systems is the inhomogeneity in their distribution of edges. The inhomogeneous distribution of the number of edges or the degree distribution of real networks reveals the existence of *communities*, where vertices are organized into groups and the edges are organized in high concentration within groups and in low concentration between these groups. This feature of real networks is known as *community structure* (Girvan and Newman, 2002). Many systems show community structures such as scientific collaboration network (Newman, 2001), online friendship network (Traud et al., 2011), citation network (Rosvall and Bergstrom, 2008), metabolic network (Holme et al., 2003) etc.

Community detection aims to identify the sub-groups, and in some cases their hierarchical organizations, within a network based on the information encoded in the graph topology. Current approaches for detecting communities can be divided into a few major categories (Fortunato, 2010; Fortunato and Hric, 2016): divisive algorithms that target to remove inter-community edges (Girvan and Newman, 2002; Radicchi et al., 2004); spectral clustering methods based on eigenvectors of graph representative matrices (Donetti and Munoz, 2004; Jin, 2015); methods based on statistical inference that fits a generative network model to the data (Guimerà and Sales-Pardo, 2009; Karrer and Newman, 2011; Peixoto, 2014); methods based on optimization of a quality functions (Newman, 2004; Huang et al., 2011; Lancichinetti et al., 2011); and methods based on dynamics (Rosvall and Bergstrom, 2008; Pons and Latapy, 2005). Among these methods, the modularity maximization-based approach proposed by Newman (2004) is by far the most popular method for community structure detection. This method works by maximizing a partition quality measure named modularity which is defined as the difference between the internal link density of a community from what one expects to find within the same group of vertices in a ran-

e-mail: nandinee@ece.ubc.ca (Nandinee Fariah Haq)

dom graph. The reason for the popularity and wide-acceptance of this method lies partly in its ability to automatically detect the number of communities. In addition, modularity is the best-known quality function and it embeds the fundamental ingredients of network community structure in its compact form - from the definition of community as a sub-graph with high internal density, to its comparison with a similar random network. Hence it generates an expression of the 'distance' of the community structure present in the network from a random network without any community structure. Therefore there has been extensive research on modularity-based techniques. Variants of the method are emerged from other choices of the null model to generate the random graph (Barber, 2007; MacMahon and Garlaschelli, 2013; Traag and Bruggeman, 2009) and different choices of the optimization techniques (Blondel et al., 2008; Guimera et al., 2004; Clauset et al., 2004; Newman, 2006; Danon et al., 2006; Pujol et al., 2006; Wakita and Tsurumi, 2007; Arenas et al., 2008).

Despite its popularity, modularity-maximization based approaches suffer from a resolution limitation that prevents them from detecting communities which are comparatively small with respect to the network as a whole, even when they are well-defined complete subgraphs like cliques (Fortunato and Barthelemy, 2007). However, real networks usually contain communities which are very diverse in sizes (Clauset et al., 2004; Danon et al., 2005; Palla et al., 2005; Guimera et al., 2003). Hence many small communities remain undetected with modularity-based approaches. Moreover, modularity-based approaches are extremely sensitive to individual connections- if two complete subgraphs are connected to each other with a few edges, the modularity-based approaches tend to cluster them together (Fortunato and Barthelemy, 2007). To overcome these limitations of modularity maximization methods, several multi-resolution approaches are proposed in the literature. One approach considers a hierarchical subdivision of detected communities by repeated application of the modularity maximization algorithm (Fortunato and Barthelemy, 2007; Ruan and Zhang, 2008). However, the communities detected using this approach is not consistent with each other since different null models are used to sub-divide each of the communities. Moreover, we need to define the stopping criteria to determine when to stop the algorithm. Another approach is to incorporate a tunable resolution parameter in the definition of modularity (Reichardt and Bornholdt, 2006; Arenas et al., 2008). By tuning the parameter one can detect communities of different sizes - going from very large to very small communities. However, for real networks, one usually has no information about the underlying community sizes, and hence tuning the parameter to the actual scale of the communities may not be feasible. Another approach is to include external degrees of communities in the definition of modularity (Li et al., 2008; Chen et al., 2015). Besides being NP-hard, the method proposed in (Li et al., 2008) was shown to be affected by a resolution limit and a tunable parameter was proposed, and methods proposed in (Chen et al., 2015, 2014) were shown to cluster unlinked nodes together (Chen et al., 2017). Therefore there is still a need to develop new approaches to overcome the limitations of modularity maximization meth-

ods.

In this work, to solve the aforementioned limitations of modularity maximization based community detection methods, we propose a community detection approach based on a modification of the traditional modularity metric. The proposed community quality metric is named as *weighted modularity*, and it does not contain any tunable parameters. Hence it is applicable for real networks where communities of different sizes co-exist and the number and size of communities are unknown. We show that by maximizing the weighted-modularity we can detect communities of different sizes. We propose a community detection approach by optimizing the weighted modularity metric and show that the proposed approach outperforms other modularity-based approaches on benchmark graphs.

2. Method

In this section, we describe the traditional modularity formulation and the motivation for proposing the weighted modularity. We then propose the weighted modularity formulation and prove that the metric does not suffer from resolution limitation. We then finally present a community detection approach based on the maximization of the weighted modularity metric.

2.1. Traditional Modularity

The idea of modularity was originally proposed by Newman and Girvan (2004) as a quality function for communities and is defined as the difference between the fraction of edges that exist within the members of a community and the expected such fraction if the edges were distributed at random. Modularity is positive if the number of edges within a community exceeds the expected such fraction in a random network. If an unweighted network with N nodes and L edges are divided into c communities, then mathematically the network modularity for this partition, Q , can be expressed as:

$$Q = \frac{1}{2L} \sum_{x,y} (A_{xy} - P_{xy}) \delta(C_x, C_y) \quad (1)$$

where A is the adjacency matrix and $A_{xy} = \{0, 1\}$ where 1 denotes the existence of an edge between nodes x and y . P_{xy} is the expected number of edges between x and y if the edges are distributed at random, C_x and C_y are the communities of x and y respectively and the δ function is defined as:

$$\delta(C_x, C_y) = \begin{cases} 1, & \text{if } C_x = C_y \\ 0, & \text{otherwise} \end{cases}$$

The expected number of edges in a random network is generally computed by using the configuration model and is defined as $P_{xy} = k_x k_y / 2L$ where k_x and k_y are the total degrees of vertices x and y . Hence Eq. 1 can be written as:

$$Q = \frac{1}{2L} \sum_{x,y} \left(A_{xy} - \frac{k_x k_y}{2L} \right) \delta(C_x, C_y). \quad (2)$$

Since the only contribution to the modularity term comes from the pair of vertices that belong to the same community, the total modularity of the network at this particular partition is expressed as a summation of the modularity term for each community as follows:

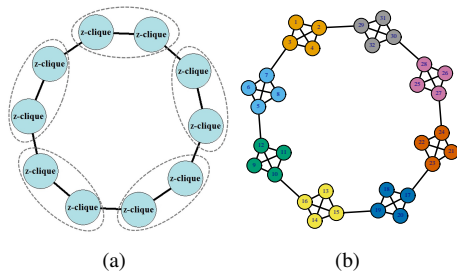


Fig. 1. Illustration of a representative network where the modularity maximization method fails to detect underlying communities. (a) Ring of z -cliques, where each clique represents a community. The dotted line shows the communities detected by traditional modularity-based methods. (b) A realization of ring of cliques network with 32 vertices divided into eight communities. Each community is a clique of four vertices where each vertex is connected to all other vertices of its community. Each 4-clique is connected with its neighbouring cliques with a single edge.

$$Q = \sum_{i=1}^c \left[\frac{l_i}{L} - \left(\frac{d_i}{2L} \right)^2 \right] = \sum_{i=1}^c (e_{ii} - a_i^2) = \sum_{i=1}^c q_i \quad (3)$$

where l_i is total number of edges within community- i and d_i is the sum of degrees of vertices in community- i . Hence $e_{ii} = l_i/L$ is the fraction of edges within community i , and $a_i = d_i/2L$ is the fraction of edges with at least one end in community- i . Modularity maximization based methods target to find such a partition of the network for which the modularity in Eq. 3 is maximum.

2.2. Weighted Modularity

The modularity maximization methods suffer from a resolution limit. A well-known example where modularity-based methods fail to detect true communities is shown in Fig. 1 (Fortunato and Barthelemy, 2007). The network consists of identical z -cliques where the cliques are connected only by single edges. Although the z -clique communities are well-defined complete subgraphs where each of the z nodes are connected with all other nodes inside its community, the traditional modularity maximization based methods fail to detect such communities. Instead, the methods prefer to partition the network where a group of cliques are put together to form a community, as shown by the dotted line in Fig. 1.

The classical modularity of a network at a particular partition is defined as the summation of the modularity term of all of its communities, q_i , as seen from Eq. 3. The total modularity formula treats contributions of all communities equally. Hence a modularity contribution of q_i gets the same *weight* irrespective of whether it comes from a strongly connected well-separated community as in Fig. 1, or from a weakly connected community. This aspect of Eq. 3 favours joining two small communities together to achieve a higher Q , especially when the size of the true community is small compared to the whole network. When the community sizes falls below a resolution limit with respect to the whole network, even a single edge triggers an increase in modularity and the traditional modularity maximization method combines communities together to form a bigger community, even when the true communities are well-separated and fully connected sub-graphs.

To avoid this resolution limitation, we propose a community quality metric by incorporating a weight term in the modularity formula that measures how *strong* a community is. We name the proposed metric *Weighted Modularity* and define the metric as:

$$Q_w = \sum_{i=1}^c \lambda_i q_i = \sum_{i=1}^c \lambda_i (e_{ii} - a_i^2) \quad (4)$$

$$\text{where } \lambda_i = 1 + \frac{2l_i}{n_i(n_i - 1)} \quad (5)$$

Here Q_w is the weighted modularity of the network and n_i is the total number of nodes within community- i . The weight term λ_i represents how strong a community is in terms of its conductance, i.e. the ratio of the edges within a community to the maximum number of possible edges. The term $\lambda_i q_i$ represents how community-like each cluster is. Incorporating λ in the modularity formula ensures the modularity contribution from densely connected communities are given more weight whereas the same modularity contribution coming from a loosely connected community is weighted less. λ denotes the strength of the community with respect to the ideal community structure, where every vertex is connected to every other vertex in its community. Note that the traditional modularity term q_i does not consider the number of nodes of a community, hence if two communities with n_1 and n_2 nodes have the same modularity where $n_1 < n_2$, the traditional modularity treats them equally, $q_1 = q_2$. Since larger communities are more likely to have higher modularity values, the traditional modularity formula favours larger communities over smaller communities. However in the weighted modularity formula the number of nodes n_i is incorporated into the weight term, hence using weighted modularity ensures $\lambda_1 q_1 > \lambda_2 q_2$, which eliminates merging two strongly connected smaller communities. If we set $\lambda_i = 1$, Eq. 4 becomes the traditional modularity equation.

2.3. Proof of Solving Resolution Limitation

In this section, we mathematically prove that the proposed weighted modularity metric, Q_w solves the resolution limitation on the examples from (Fortunato and Barthelemy, 2007). Here we prove that maximizing Q_w neither divides a complete subgraph into two or more parts, nor it merges two or more adjacent complete subgraphs.

2.3.1. Proof: Weighted Modularity Does Not Divide Cliques

Given a clique with m nodes where $m \geq 3$ let us consider a partition \mathcal{P} that divides the clique into two communities c_1 and c_2 with nodes m_1 and m_2 . Let Q_w^s be the weighted modularity when the whole clique is considered a single community, and Q_w^p be the weighted modularity of the partition \mathcal{P} . We have to show that higher value of Q_w is achieved when the clique is considered a single community, or $Q_w^s > Q_w^p$. Here the total number of edges in the network is, $L = m(m-1)/2$ and the number of edges between c_1 and c_2 is $m_1 m_2$. By definitions, $Q_w^s = 0$ and $Q_w^p = \sum_{i=1}^2 \lambda_i \left[\frac{l_i}{L} - \left(\frac{d_i}{2L} \right)^2 \right]$, where λ_i , l_i and d_i denote the λ , l and d terms of community- i as defined in Table.

1. For community $i = 1$ generated by dividing a clique, $\lambda_1 = 2$, $l_1 = m_1(m_1 - 1)/2$, $d_1 = 2l_1 + m_1m_2$. Similar equations can be derived for community $i = 2$. By replacing these values in Q_w^p we get (see appendix for details) :

$$Q_w^p = \frac{-4m_1m_2}{m^2(m-1)} < 0 \text{ for } m \geq 3 \quad (6)$$

So, $Q_w^p < Q_w^s$. Hence for any clique, the maximum weighted modularity will reach its maximum value when the whole clique is one single community, and maximizing weighted modularity will not divide the cliques.

2.3.2. Proof: Weighted Modularity Does Not Merge Cliques

In this section we prove that maximizing weighted modularity does not merge adjacent complete subgraphs as shown in Fig. 1. Given a ring of $n \geq 3$ cliques where each clique has $m \geq 3$ nodes and $m(m-1)/2$ edges, the total number of nodes and edges in the network is $N = nm$ and $L = nm(m-1)/2 + n$, where cliques are connected with adjacent cliques with a single edge. Let Q_w^s be the weighted modularity of the partition \mathcal{P}^s with n -communities when one single clique forms a single community, and Q_w^p be the weighted modularity of the partition \mathcal{P}^p with $n/2$ -communities when two adjacent cliques are merged into one community as depicted in Fig. 1. We have to show that, $Q_w^s > Q_w^p$, or

$$Q_w^s - Q_w^p > 0 \quad (7)$$

Let λ_s, q_s and λ_p, q_p be the λ and q terms of the communities of the partitions \mathcal{P}^s and \mathcal{P}^p respectively. By definition, $Q_w^s = n\lambda_s q_s$ and $Q_w^p = (n/2)\lambda_p q_p$. It can be written as:

$$\lambda_s = 2; \quad nq_s = 1 - \frac{2}{m(m-1)+2} - \frac{1}{n}$$

$$\lambda_p = \frac{3}{2} - \frac{m-2}{2m(2m-1)}; \quad \frac{n}{2}q_p = 1 - \frac{1}{m(m-1)+2} - \frac{2}{n}$$

By using these values the left side of Eq. 7 can be written as (see appendix for details):

$$Q_w^s - Q_w^p = 0.5 + \frac{m^3 - 13m^2 + 8m - 2}{2m(2m-1)[m(m-1)+2]} + \frac{2[(m-1)^2 + m]}{nm(2m-1)} \quad (8)$$

The first and third terms of Eqn. 8 are always positive for $m \geq 3$. The second term reaches its lowest value at $m = 3$ and the lowest value is -0.2833 , which when combined with the first term of the equation, generates a positive value. Hence overall, $Q_w^s - Q_w^p > 0$, for all $m \geq 3$, $n \geq 3$. Since inequality 7 holds, maximizing weighted modularity will not merge adjacent cliques, and it will always detect the smallest clique as a single community.

2.4. Community Detection by the Maximization of Weighted Modularity

In this section, we describe a community detection method by maximizing the weighted modularity. We propose an agglomerative clustering approach based on greedy optimization to maximize weighted modularity. The process starts by putting each node of the network into their own separate communities. So at the first step we start from N number of communities, where N is the total number of nodes in the network. Then the

Table 1. Definition of variables.

N	Total number of nodes in the network
L	Total number of edges in the network
c	Total number of communities
n_i	Total number of vertices in community- i
d_i	Sum of degrees of vertices in community- i
l_i	Total number of edges within community- i
l_{ij}^{ext}	Total number of edges between communities i and j
q_i	Modularity term of community- i . $q_i = [(l_i/m) - (d_i/2m)^2]$
λ_i	Weight term for community- i as defined in Eq.5
Q_w	Weighted modularity as defined in Eq.4
$\mathbf{n}^{c \times 1}$	Community size vector with elements $[n_1, n_2, \dots, n_c]^T$
$\mathbf{d}^{c \times 1}$	Internal degree vector with elements $[d_1, d_2, \dots, d_c]^T$
$\mathbf{l}^{c \times 1}$	Internal edge vector with elements $[l_1, l_2, \dots, l_c]^T$
$\mathbf{l}^{c \times c}$	External edge matrix with elements $\{l_{ij}^{ext}\}$
$\mathbf{Q}^{c \times 1}$	Modularity vector with elements $[q_1, q_2, \dots, q_c]^T$
$\mathbf{\Lambda}^{c \times 1}$	Weight vector with elements $[\lambda_1, \lambda_2, \dots, \lambda_c]^T$
$\mathbf{C}^{N \times 1}$	Community label vector

increase in weighted modularity is calculated if any two communities are merged. Finally, a new community is formed by merging those two communities for which the increase in terms of weighted modularity is maximum. At each iteration step, this process is repeated and communities are merged repeatedly. The process terminates when no increase in weighted modularity is possible, i.e. no increment in weighted modularity is observed by merging any two of the remaining communities.

We maintain a few variables and matrices to compute the changes in weighted modularity efficiently. The variables and matrices are defined in Tab. 1. If community- i is merged with community- j then the gain in weighted modularity can be expressed as:

$$\Delta Q_w(i, j) = \lambda_{com} \times q_{com} - [\lambda_i \times q_i + \lambda_j \times q_j] \quad (9)$$

where q_{com} and λ_{com} are the modularity and the weight term for the new community generated by merging i and j . At each iteration step, the maximum increase in terms of weighted modularity is calculated. Other variables for the new community are calculated as follows:

$$n_{com} = n_i + n_j; \quad d_{com} = d_i + d_j; \quad l_{com} = l_i + l_j + l_{ij}^{ext};$$

$$\lambda_{com} = 1 + \frac{2l_{com}}{n_{com}(n_{com} - 1)}; \quad q_{com} = \frac{l_{com}}{m} - \left(\frac{d_{com}}{2m}\right)^2 \quad (10)$$

At each iteration step, we solve for the communities i and j where merging the communities i and j generates the maximum increase of weighted modularity: $u, v = \arg \max_{i,j} \Delta Q_w(i, j)$.

We then update the matrices by removing the j^{th} entries and replacing the i^{th} entries with the updated values, i.e., $n_i = n_{com}$, $l_i = l_{com}$, $d_i = d_{com}$, $q_i = q_{com}$, $\lambda_i = \lambda_{com}$. The process terminates when no further increase in weighted modularity is observed and the partition with the maximum weighted modularity represents the detected communities. The greedy optimization approach for community detection by maximizing weighted modularity is outlined in Algorithm 1.

The community structure detected by the greedy optimization can further be refined by applying a complete refinement step at the end of the algorithm. The refinement approach is outlined in Algorithm 2. At the refinement step, weighted modularity change is calculated if a node is moved to its neighbouring

Algorithm 1 Community detection by maximization of weighted modularity : greedy optimization

```

1: Input : Network,  $\mathcal{G}$ 
2: function WEIGHTED_MODULARITY_MAXIMIZATION( $\mathcal{G}$ )
3:   Initialize :  $\mathbf{C} = [1, 2, \dots, N]^T$ ,  $c = N$ 
4:   Calculate :  $\mathbf{A}$ ,  $\mathbf{n}$ ,  $\mathbf{d}$ ,  $\mathbf{l}$ ,  $\mathbf{l}_{\text{ext}}$ ,  $\mathbf{Q}$  and  $Q_w$ 
5:   Set  $Q_w^{\text{best}} = Q_w$  and  $\mathbf{C}^{\text{best}} = \mathbf{C}$ 
6:   while  $c > 1$  do
7:     for  $i, j \in \{1, \dots, c\}$ ;  $i < j$ ;  $l_{ij}^{\text{ext}} > 0$  do
8:        $u, v = \arg \max_{i, j} \Delta Q_w(i, j)$ 
9:     end for
10:     $\Delta Q_w^{\text{max}} = \Delta Q_w(u, v)$ 
11:    Update :  $\mathbf{C}$ ,  $\mathbf{A}$ ,  $\mathbf{n}$ ,  $\mathbf{d}$ ,  $\mathbf{l}$ ,  $\mathbf{l}_{\text{ext}}$ ,  $\mathbf{Q}$ 
12:     $Q_w = \mathbf{A}^T \mathbf{Q}$ 
13:    if  $Q_w > Q_w^{\text{best}}$  then
14:       $Q_w^{\text{best}} = Q_w$ 
15:       $\mathbf{C}^{\text{best}} = \mathbf{C}$ 
16:    end if
17:  end while
18: end function
19: Output :  $\mathbf{C}^{\text{best}}$ 

```

vertices' communities. If a vertex, u from community $i = \mathbf{C}[u]$ is moved to the community $j = \mathbf{C}[v]$ of its neighbouring vertex, v , the change in weighted modularity due to this movement can be calculated from the changes in n_i, n_j and l_i, l_j terms associated with communities- i and j . The change in weighted modularity can be written as:

$$Q_{u \rightarrow v}^{\text{move}} = [\lambda_i^{\text{move}} \times q_i^{\text{move}} + \lambda_j^{\text{move}} \times q_j^{\text{move}}] - [\lambda_i \times q_i + \lambda_j \times q_j] \quad (11)$$

where, $Q_{u \rightarrow v}^{\text{move}}$ is the change in weighted modularity if node, u is moved from its community- i to the community- j of node, v . λ^{move} and q^{move} denote the changed values of λ and q due to this movement. This process is repeated for all pair of nodes, and finally a node is moved to a new community for which the maximum $Q_{u \rightarrow v}^{\text{move}}$ is reached. This process is repeated until there is no further increase in weighted modularity.

Algorithm 2 Community detection by maximization of weighted modularity : refinement

```

Input :  $\mathcal{G}$ ,  $\mathbf{C}^{\text{best}} = \text{Weighted\_Modularity\_Maximization}(\mathcal{G})$ 
2: function COMPLETE_GREEDY_REFINEMENT( $\mathcal{G}$ ,  $\mathbf{C}^{\text{best}}$ )
   Calculate :  $\mathbf{A}$ ,  $\mathbf{n}$ ,  $\mathbf{d}$ ,  $\mathbf{l}$ ,  $\mathbf{l}_{\text{ext}}$ ,  $\mathbf{Q}$  and  $Q_w$ 
4:   Set  $Q_w^{\text{best}} = Q_w$ ,  $Q_w^{\text{move}} = 0$ 
   while 1 do
6:     for  $u, v \in \{1, \dots, N\}$ ;  $l_{uv}^{\text{ext}} > 0$  do
        $u^{\text{best}}, v^{\text{best}} = \arg \max_{u, v} Q_{u \rightarrow v}^{\text{move}}$ 
8:     end for
     if  $Q_{u^{\text{best}} \rightarrow v^{\text{best}}}^{\text{move}} > 0$  then ▷ move u to the community of v
9:        $\mathbf{C}^{\text{best}}[u^{\text{best}}] = \mathbf{C}^{\text{best}}[v^{\text{best}}]$ 
10:      Update :  $\mathbf{A}$ ,  $\mathbf{n}$ ,  $\mathbf{d}$ ,  $\mathbf{l}$ ,  $\mathbf{l}_{\text{ext}}$ ,  $\mathbf{Q}$ ,  $Q_w^{\text{best}}$ 
12:      continue
     else break
14:   end if
   end while
16: end function
Output :  $\mathbf{C}^{\text{best}}$ 

```

3. Experimental Results

In this section, we report the performances of the proposed community detection method for several benchmark networks with known community structures. We compared the proposed method with a widely used modularity maximization based approach by Clauset et al. (2004), which is a faster implementation of the traditional method (Newman, 2004) that has been used to compare the performances of other modularity-based algorithms in literature (Chen et al., 2014; Li et al., 2008; Chen et al., 2015; Xiang et al., 2016). We also compared the proposed method with a multi-resolution based modularity maximization approach proposed in Reichardt and Bornholdt (2006) and a more recent method proposed in Chen et al. (2015) whenever the results are available. The generated partitions are compared with the true communities using two widely used measures for comparing the performances of community detection methods, namely *normalized mutual information* and *variation of information*. For two partitions of a network \mathcal{X} with n_X communities and \mathcal{Y} with n_Y communities, the normalized mutual information (NMI) is defined as (Danon et al., 2005): $NMI_{\mathcal{X}, \mathcal{Y}} = 2I_{\mathcal{X}, \mathcal{Y}} / (H_{\mathcal{X}} + H_{\mathcal{Y}})$, where X and Y denote the community labels of the nodes in partitions \mathcal{X} and \mathcal{Y} respectively. $I_{\mathcal{X}, \mathcal{Y}}$ is the mutual information between X and Y and $H_{\mathcal{X}}$ and $H_{\mathcal{Y}}$ are their entropy. The variation of information (VI) between partitions \mathcal{X} and \mathcal{Y} is defined as (Meila, 2003): $VI_{\mathcal{X}, \mathcal{Y}} = H_{\mathcal{X}|\mathcal{Y}} + H_{\mathcal{Y}|\mathcal{X}}$, where $H_{\mathcal{X}|\mathcal{Y}}$ is the conditional entropy of X given Y and $H_{\mathcal{Y}|\mathcal{X}}$ is the conditional entropy of Y given X .

3.1. Planted ℓ -Partition Networks

We evaluated the performance of the proposed method on the benchmark networks generated with the planted ℓ -partition model (Condon and Karp, 2001). The model partitions a network with N vertices into c equal-size groups with n_c nodes each. Edges are placed at random with a probability p_{in} between vertices of the same community and with a probability p_{out} between vertices in different communities, where $p_{in} > p_{out}$. The probabilities are calculated from the average degree of the nodes k . We generated several networks by changing the average external degree of vertices, k_{out} . In this work we used a network with $N = 128$ nodes and divided them into $c = \{8, 16, 32\}$ communities. Tab. 2 reports the performance of the weighted modularity method in terms of NMI and VI for different settings. We report the average performance over 20 realizations of each settings. As can be seen from the table, for benchmark networks generated with planted ℓ -partition model, the proposed method performs better than (Clauset et al., 2004) and generates comparable performance with the other method. Note that the multi-resolution method by Reichardt *et al.* (Reichardt and Bornholdt, 2006) has a tunable parameter that needs to be tuned to the resolution of the underlying communities, and it cannot detect disconnected communities ($k_{out} = 0$). On the other hand, the proposed weighted modularity method does not need to tune any parameter and still generates comparable results with the other literature-based approaches. Moreover, as we increased the number of communities (for $c = 16, 32$), many communities became unconnected

Table 2. Performances of the proposed method on networks generated with the planted ℓ -partition model. The networks are generated with $N = 128$ number of nodes which are divided into different number of communities. The table reports the average normalized mutual information (NMI) and variation of information (VI) over 20 realizations of the network.

c^*	$k^\#$	k_{out}^+	NMI			VI		
			Weighted Modularity	Clauset et al. (2004)	Reichardt and Bornholdt (2006)	Weighted Modularity	Clauset et al. (2004)	Reichardt and Bornholdt (2006)
8	8	0	1.00	1.00	-	0.00	0.00	0.00
		1	1.00	0.97	0.99	0.01	0.10	0.03
		2	0.98	0.89	0.98	0.09	0.45	0.07
		3	0.86	0.68	0.91	0.59	1.22	0.38
		4	0.59	0.48	0.65	1.94	1.97	1.43
		5	0.40	0.25	0.34	2.91	2.89	2.75
		6	0.29	0.13	0.16	3.46	3.38	3.24
		8	0.22	0.09	0.12	3.76	3.51	3.61
16	4	0	0.18	0.07	0.07	3.98	3.65	3.61
		1	0.99	1.00	-	0.08	0.01	-
		2	0.88	0.80	-	0.67	1.02	-
		3	0.66	0.51	-	2.03	2.47	-
32	2	4	0.50	0.31	-	2.97	3.42	-
		1	0.40	0.21	-	3.54	3.90	-
		0	0.96	0.96	-	0.29	0.29	-
		1	0.78	0.68	-	1.65	2.09	-
		2	0.63	0.50	-	2.60	3.17	-

* c -Number of Communities # k -Average Degree + k_{out} - Average External Degree

and the multi-resolution method could not work in these settings.

3.2. LFR Benchmark

The planted ℓ -partition model divides the network into equal size communities and the vertices also have approximately the same degree. These two features are at odds with real networks. A more realistic benchmark is the LFR benchmark networks (Lancichinetti et al., 2008) where the heterogeneity of both the degrees and the community sizes are taken into account that is observed in networks of real systems. In LFR benchmark networks the community sizes and the degrees are distributed according to power law and hence the generated networks have a combination of small and big communities which is observed in real networks. We applied the proposed weighted modularity method on LFR benchmark networks and compared its performance with existing methods. Tab. 3 shows the performance of the methods in terms of NMI and VI over different values of the mixing parameter, μ . The LFR benchmark network reported here has 1000 nodes with an average degree of 15 and a maximum degree of 50. The minimum and maximum community sizes are 10 and 50 respectively. The degree sequence exponent is 2 and community size distribution exponent is 1. The mixing parameter, μ is varied from 0 to 0.5. The low values of the mixing parameter denote strong community structure and higher values denote the existence of weak communities. For each μ , 20 realizations of the LFR networks are generated and the average NMI and VI are reported. From the table, it is clear that the proposed method gives consistent performance in practical networks, where communities of different sizes are present and the degrees of nodes varies. The proposed method performs better than (Clauset et al., 2004) and (Chen et al., 2015) and generates comparable performance with (Reichardt and Bornholdt, 2006). For the multi-resolution method in (Reichardt and Bornholdt, 2006) the parameter was tuned to the underlying resolution of the communities.

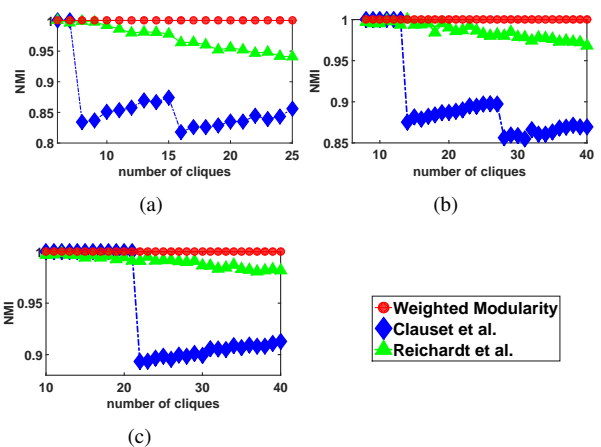


Fig. 2. Performance of the proposed method on the ring of cliques network. The plots show the performance of different algorithms in terms of normalized mutual information over the total number of cliques, n_c in the network. (a) Performance for networks with 3-cliques. (b) Performance for networks with 4-cliques. (c) Performance for networks with 5-cliques.

3.3. Ring of Cliques

One of the challenging networks where most modularity-maximization based methods fail to detect the underlying communities is the ring of cliques network (Fortunato and Barthelemy, 2007). The ring of cliques network consists of identical z -cliques where z -nodes of a community are connected with each other, and each clique is connected with its neighbours with only one edge, forming a ring-like structure as shown in Fig. 1. We generated several ring of cliques networks using cliques of three, four and five nodes and compared the performances of the modularity-based algorithms in these networks. Fig. 2 report the performances of the proposed method along with other literature-based methods against the total number of cliques in the network. Each point on these plots is an average over 20 realizations of the networks. Although cliques form the strongest possible community, the literature-

Table 3. Performance comparisons between the proposed method and other literature-based algorithms on LFR benchmark networks. The table reports the normalized mutual information (NMI) and variation of information (VI) for different values of the mixing parameter, μ .

μ^*	NMI				VI			
	Weighted Modularity	Clauset et al. (2004)	Reichardt and Bornholdt (2006)	Chen et al. (2015, 2014)	Weighted Modularity	Clauset et al. (2004)	Reichardt and Bornholdt (2006)	Chen et al. (2015, 2014)
0.05	1.00	0.98	0.79	0.91	0.00	0.10	0.03	0.81
0.10	1.00	0.95	0.99	0.94	0.00	0.32	0.04	0.58
0.15	1.00	0.91	0.99	0.94	0.00	0.59	0.05	0.68
0.20	1.00	0.87	0.98	0.93	0.00	0.82	0.08	0.78
0.25	1.00	0.81	0.98	0.90	0.00	1.21	0.10	1.02
0.30	1.00	0.77	0.98	0.89	0.00	1.36	0.07	1.13
0.35	1.00	0.73	0.99	0.91	0.01	1.61	0.06	0.94
0.40	1.00	0.67	0.97	0.91	0.01	1.82	0.15	0.95
0.45	1.00	0.63	0.98	0.89	0.01	2.06	0.13	1.15
0.50	0.98	0.59	0.98	0.80	0.11	2.23	0.08	1.97

* μ -Mixing Parameter

based methods fail to detect them when the total number of cliques in the network increases, i.e. the network size increases, as can be seen from Fig. 2. On the other hand, the proposed method correctly identifies the true communities from the ring of clique networks, even for the large graphs, hence $NMI=1$ for all cases.

3.4. Real Networks

We applied the proposed method on real networks to extract community structures. The first network considered here is the American college football network (Girvan and Newman, 2002) that represents the network of United States football games between Division IA colleges during the season of Fall 2000. The vertices represent teams and edges represent games between the teams. The network consists of 115 nodes and 613 edges and it incorporates a known community structure. The teams (i.e. nodes of the network) are divided into twelve conferences where games are more frequent between teams that belong to the same conference. For this network we see the teams played on average seven intra-conference games as opposed to four inter-conference games, hence generating a higher number of edges between nodes within the conferences. These conferences are treated as ground truth communities for the network. We applied our proposed method on this network to find communities, and the proposed method achieved a normalized mutual information of 0.91 when compared with the true communities. We applied literature-based approaches on the same network and the NMI between the true communities and the communities detected by (Clauset et al., 2004) and (Reichardt and Bornholdt, 2006) was 0.70 and 0.88 respectively. The proposed method detected twelve communities, which matches with the number of underlying true communities, whereas the number communities detected by (Clauset et al., 2004) and (Reichardt and Bornholdt, 2006) was six and ten respectively.

Another network that we considered here is a network of 105 books about the United States politics (Krebs). The nodes represent books published around the time of the 2004 presidential election and sold by the online bookseller Amazon. Edges represent frequent co-purchasing of books by the same buyers, as indicated by the *customers who bought this book also bought these other books* feature on the Amazon website. The books were later labeled by Mark Newman into three categories as *liberal*, *conservative* and *neutral* based on a reading of the descriptions and reviews of the books posted on Amazon. The

application of the proposed method on this network revealed seven communities as shown in Fig. 3. If we consider the three categories as the true community structure of the network, the NMI between the these communities and the communities detected by the proposed method is 0.50. However we observed better modular structure with the detected communities in terms of average clustering coefficient and modularity. The modularity value with the three categories is 0.42, whereas with the proposed method it is 0.52, and average clustering coefficient with the proposed method is 0.69, whereas with the three groups it was 0.33. Hence with the seven communities, a higher modularity and clustering coefficient value is achieved, which denotes that the seven communities better represent the modular structure of the network. We observed that the *neutral* books are mainly divided into three communities. Careful observation of the network reveals that a few *neutral*-labeled nodes do not share edges with other neutral nodes, and after applying the proposed method these nodes are separated from other *neutral*-labeled nodes and forms their own communities with a few *liberal* and *conservative* nodes. We also observed existence of two small communities inside the *liberal* and *conservative* groups.

4. Conclusion

In this paper we present a community detection method based on the maximization of a community quality metric named as *weighted modularity*. We demonstrated that exploring weighted modularity can overcome the resolution limitation of the traditional modularity based approaches for community detection. We proposed a greedy optimization based approach to detect communities from networks by maximizing the weighted modularity. We applied the proposed method on several standard benchmark networks and compared the performance of the method with other literature-based algorithms. We also applied the method to real-world networks and reported the detected communities.

The strength of the proposed weighted-modularity based approach to community detection is that it needs no prior knowledge on the number and sizes of the underlying communities, and hence it is applicable for real networks where prior information about the communities may not always be available. The method can successfully detect complete subgraphs from any network, irrespective of the network size. One limitation of the proposed method is that, in its current implementation, the

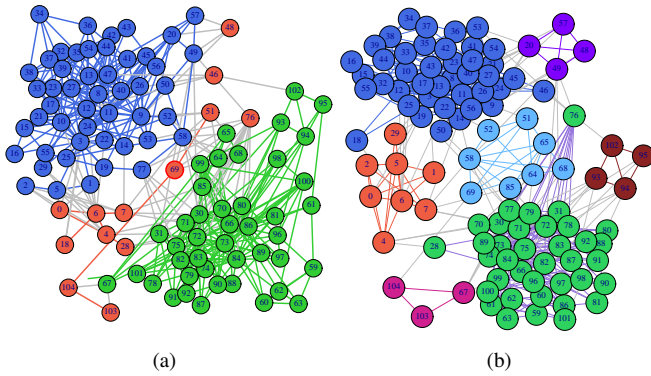


Fig. 3. Community structure of the political books network. (a) Network of books on US politics labeled as conservative (blue), neutral (orange) and liberal (green). (b) Detected community structure. Existence of small communities are observed inside conservative and liberal groups. The neutral-labeled group is divided into two communities, and one group is formed with few neutral, conservative and liberal nodes.

community detection approach is not able to assess the significance of the detected communities. Moreover, the proposed method, in its current implementation, can detect communities from unweighted and undirected networks. The proposed method can be further modified to detect communities from other types of networks such as weighted, directed and/or binary networks. We plan to extend the method for these types of networks in the near future.

References

- Arenas, A., Fernandez, A., Gomez, S., 2008. Analysis of the structure of complex networks at different resolution levels. *New Journal of Physics* 10, 053039.
- Barber, M.J., 2007. Modularity and community detection in bipartite networks. *Physical Review E* 76, 066102.
- Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008, P10008.
- Chen, M., Kuzmin, K., Szymanski, B.K., 2014. Community detection via maximization of modularity and its variants. *IEEE Transactions on Computational Social Systems* 1, 46–65.
- Chen, M., Nguyen, T., Szymanski, B.K., 2015. A new metric for quality of network community structure. *arXiv preprint arXiv:1507.04308*.
- Chen, T., Singh, P., Bassler, K.E., 2017. Network community detection using modularity density measures. *arXiv preprint arXiv:1708.06810*.
- Clauset, A., Newman, M.E., Moore, C., 2004. Finding community structure in very large networks. *Physical review E* 70, 066111.
- Condon, A., Karp, R.M., 2001. Algorithms for graph partitioning on the planted partition model. *Random Structures and Algorithms* 18, 116–140.
- Danon, L., Díaz-Guilera, A., Arenas, A., 2006. The effect of size heterogeneity on community identification in complex networks. *Journal of Statistical Mechanics: Theory and Experiment* 2006, P11010.
- Danon, L., Díaz-Guilera, A., Duch, J., Arenas, A., 2005. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment* 2005, P09008.
- Donetti, L., Munoz, M.A., 2004. Detecting network communities: a new systematic and efficient algorithm. *Journal of Statistical Mechanics: Theory and Experiment* 2004, P10012.
- Fortunato, S., 2010. Community detection in graphs. *Physics Reports* 486, 75–174.
- Fortunato, S., Barthelemy, M., 2007. Resolution limit in community detection. *Proceedings of the National Academy of Sciences* 104, 36–41.
- Fortunato, S., Hric, D., 2016. Community detection in networks: A user guide. *Physics Reports* 659, 1–44.

- Girvan, M., Newman, M.E., 2002. Community structure in social and biological networks. *Proceedings of the national academy of sciences* 99, 7821–7826.
- Guimera, R., Danon, L., Diaz-Guilera, A., Giralt, F., Arenas, A., 2003. Self-similar community structure in a network of human interactions. *Physical review E* 68, 065103.
- Guimerà, R., Sales-Pardo, M., 2009. Missing and spurious interactions and the reconstruction of complex networks. *Proceedings of the National Academy of Sciences* 106, 22073–22078.
- Guimera, R., Sales-Pardo, M., Amaral, L.A.N., 2004. Modularity from fluctuations in random graphs and complex networks. *Physical Review E* 70, 025101.
- Holme, P., Huss, M., Jeong, H., 2003. Subnetwork hierarchies of biochemical pathways. *Bioinformatics* 19, 532–538.
- Huang, J., Sun, H., Liu, Y., Song, Q., Wengner, T., 2011. Towards online multiresolution community detection in large-scale networks. *PLoS one* 6, e23829.
- Jin, J., 2015. Fast community detection by SCORE. *The Annals of Statistics* 43, 57–89.
- Karrer, B., Newman, M.E., 2011. Stochastic blockmodels and community structure in networks. *Physical Review E* 83, 016107.
- Krebs, V., unpublished. <http://www.orgnet.com/>.
- Lancichinetti, A., Fortunato, S., Radicchi, F., 2008. Benchmark graphs for testing community detection algorithms. *Physical review E* 78, 046110.
- Lancichinetti, A., Radicchi, F., Ramasco, J.J., Fortunato, S., 2011. Finding statistically significant communities in networks. *PLoS one* 6, e18961.
- Li, Z., Zhang, S., Wang, R.S., Zhang, X.S., Chen, L., 2008. Quantitative function for community detection. *Physical review E* 77, 036109.
- MacMahon, M., Garlaschelli, D., 2013. Community detection for correlation matrices. *arXiv preprint arXiv:1311.1924*.
- Meila, M., 2003. Comparing clusterings by the variation of information, in: *Colt*. volume 3, pp. 173–187.
- Newman, M.E., 2001. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences* 98, 404–409.
- Newman, M.E., 2004. Fast algorithm for detecting community structure in networks. *Physical review E* 69, 066133.
- Newman, M.E., 2006. Modularity and community structure in networks. *Proceedings of the national academy of sciences* 103, 8577–8582.
- Newman, M.E., Girvan, M., 2004. Finding and evaluating community structure in networks. *Physical review E* 69, 026113.
- Palla, G., Derényi, I., Farkas, I., Vicsek, T., 2005. Uncovering the overlapping community structure of complex networks in nature and society. *Nature* 435, 814–818.
- Peixoto, T.P., 2014. Hierarchical block structures and high-resolution model selection in large networks. *Physical Review X* 4, 011047.
- Pons, P., Latapy, M., 2005. Computing communities in large networks using random walks, in: *ISCIS*. volume 3733, pp. 284–293.
- Pujol, J.M., Béjar, J., Delgado, J., 2006. Clustering algorithm for determining community structure in large networks. *Physical Review E* 74, 016107.
- Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D., 2004. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America* 101, 2658–2663.
- Reichardt, J., Bornholdt, S., 2006. Statistical mechanics of community detection. *Physical Review E* 74, 016110.
- Rosvall, M., Bergstrom, C.T., 2008. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* 105, 1118–1123.
- Ruan, J., Zhang, W., 2008. Identifying network communities with a high resolution. *Physical Review E* 77, 016104.
- Traag, V.A., Bruggeman, J., 2009. Community detection in networks with positive and negative links. *Physical Review E* 80, 036115.
- Traud, A.L., Kelsic, E.D., Mucha, P.J., Porter, M.A., 2011. Comparing community structure to characteristics in online collegiate social networks. *SIAM review* 53, 526–543.
- Wakita, K., Tsurumi, T., 2007. Finding community structure in mega-scale social networks, in: *Proceedings of the 16th international conference on World Wide Web*, pp. 1275–1276.
- Xiang, J., Hu, T., Zhang, Y., Hu, K., et al., 2016. Local modularity for community detection in complex networks. *Physica A: Statistical Mechanics and its Applications* 443, 451–459.

Appendix

A1. Proof: Weighted Modularity Does Not Divide Cliques

Given a clique with m nodes where $m \geq 3$ let us consider a partition \mathcal{P} that divides the clique into two communities c_1 and c_2 with nodes m_1 and m_2 . Let Q_w^s be the weighted modularity when the whole clique is considered a single community, and Q_w^p be the weighted modularity of the partition \mathcal{P} . We have to show that higher value of Q_w is achieved when the clique is considered a single community, or $Q_w^s > Q_w^p$. Here the total number of edges in the network is, $L = m(m-1)/2$ and the number of edges between c_1 and c_2 is m_1m_2 . Then by definitions,

$$Q_w^s = 0 \quad (1)$$

$$Q_w^p = \sum_{i=1}^2 \lambda_i \left[\frac{l_i}{L} - \left(\frac{d_i}{2L} \right)^2 \right] \quad (2)$$

where λ_1 and λ_2 are the weight terms, l_1 and l_2 are the total number of edges within community, d_1 and d_2 denote the sum of degrees of vertices for community-1 and 2 respectively. For community-1 and 2 generated by dividing a clique,

$$l_1 = m_1(m_1 - 1)/2 \quad (3)$$

$$l_2 = m_2(m_2 - 1)/2 \quad (4)$$

$$d_1 = 2l_1 + m_1m_2 \quad (5)$$

$$d_2 = 2l_2 + m_1m_2 \quad (6)$$

$$\lambda_1 = 1 + \frac{2l_1}{m_1(m_1 - 1)} = 1 + \frac{2m_1(m_1 - 1)/2}{m_1(m_1 - 1)} = 2 \quad (7)$$

$$\lambda_2 = 1 + \frac{2l_2}{m_2(m_2 - 1)} = 1 + \frac{2m_2(m_2 - 1)/2}{m_2(m_2 - 1)} = 2 \quad (8)$$

By replacing the values from Eqn. 3-8 in Eqn. 2 we get :

$$\begin{aligned} Q_w^p &= 2 \left[\frac{m_1(m_1 - 1)}{m(m - 1)} + \frac{m_2(m_2 - 1)}{m(m - 1)} \right. \\ &\quad \left. - \left(\frac{m_1(m_1 - 1) + m_1m_2}{m(m - 1)} \right)^2 - \left(\frac{m_2(m_2 - 1) + m_1m_2}{m(m - 1)} \right)^2 \right] \\ &= 2 \left[\frac{m_1(m_1 - 1) + m_2(m_2 - 1)}{m(m - 1)} \right. \\ &\quad \left. - \left(\frac{m_1(m - 1)}{m(m - 1)} \right)^2 - \left(\frac{m_2(m - 1)}{m(m - 1)} \right)^2 \right] \\ &= 2 \left[\frac{m_1(m_1 - 1) + m_2(m_2 - 1)}{m(m - 1)} - \frac{m_1^2 + m_2^2}{m^2} \right] \\ &= \frac{-4m_1m_2}{m^2(m - 1)} \end{aligned}$$

For $m \geq 3$, $Q_w^p = \frac{-4m_1m_2}{m^2(m-1)} < 0$.

So, $Q_w^p < Q_w^s$. Hence for any clique, the maximum weighted modularity will reach its maximum value when the whole clique is one single community, and maximizing weighted modularity will not divide the cliques.

A2. Proof: Weighted Modularity Does Not Merge Cliques

In this section we prove that maximizing weighted modularity does not merge adjacent complete subgraphs as shown in Fig. 1. Given a ring of $n \geq 3$ cliques where each clique has $m \geq 3$ nodes and $m(m-1)/2$ edges and cliques are connected with adjacent cliques with a single edge. Then the total number of nodes and edges in the network is,

$$N = nm \quad (1)$$

$$L = nm(m-1)/2 + n \quad (2)$$

Let Q_w^s be the weighted modularity of the partition \mathcal{P}^s with n -communities when one single clique forms a single community, and Q_w^p be the weighted modularity of the partition \mathcal{P}^p with $n/2$ -communities when two adjacent cliques are merged into one community. We have to show that,

$$Q_w^s > Q_w^p \quad (3)$$

$$\text{or, } Q_w^s - Q_w^p > 0 \quad (4)$$

Let λ_s and λ_p be the weight terms, q_s and q_p be the modularity term, l_s and l_p be the total number of edges within community, d_s and d_p denote the sum of degrees of vertices for the communities of the partitions \mathcal{P}^s and \mathcal{P}^p respectively, where,

$$l_s = m(m-1)/2 \quad (5)$$

$$l_p = 2l_s + 1 = m(m-1) + 1 \quad (6)$$

$$d_s = 2l_s + 2 = m(m-1) + 2 \quad (7)$$

$$d_p = 2d_s = 2[m(m-1) + 2] \quad (8)$$

By definition,

$$Q_w^s = n\lambda_s q_s \quad (9)$$

$$\text{and } Q_w^p = (n/2)\lambda_p q_p \quad (10)$$

Using Eqn. 1-2 and 5- 8, for partition \mathcal{P}^s we can write:

$$\lambda_s = 1 + \frac{2l_s}{m(m-1)} = 1 + \frac{m(m-1)}{m(m-1)} = 2 \quad (11)$$

$$\begin{aligned} nq_s &= n \left[\frac{l_s}{L} - \left(\frac{d_s}{2L} \right)^2 \right] \\ &= \frac{m(m-1)}{m(m-1) + 2} - \frac{1}{n} \left[\frac{m(m-1) + 2}{m(m-1) + 2} \right]^2 \\ &= 1 - \frac{2}{m(m-1) + 2} - \frac{1}{n} \end{aligned} \quad (12)$$

And for partition \mathcal{P}^p we can write:

$$\begin{aligned} \lambda_p &= 1 + \frac{2l_p}{2m(2m-1)} \\ &= 1 + \frac{m(m-1) + 1}{m(2m-1)} \\ &= \frac{3m^2 - 2m + 1}{m(2m-1)} \\ &= \frac{3}{2} - \frac{m-2}{2m(2m-1)} \end{aligned} \quad (13)$$

$$\begin{aligned}
\frac{n}{2}q_p &= \frac{n}{2} \left[\frac{l_p}{L} - \left(\frac{d_p}{2L} \right)^2 \right] \\
&= \frac{m(m-1)+1}{m(m-1)+2} - \frac{2}{n} \left[\frac{m(m-1)+2}{m(m-1)+2} \right]^2 \\
&= 1 - \frac{1}{m(m-1)+2} - \frac{2}{n}
\end{aligned} \tag{14}$$

Replacing values from Eqn. 11-12 in Eqn. 9:

$$Q_w^s = 2 - \frac{4}{m(m-1)+2} - \frac{2}{n} \tag{15}$$

Replacing values from Eqn. 13-14 in Eqn. 10:

$$\begin{aligned}
Q_w^p &= \frac{3}{2} - \frac{m-2}{2m(2m-1)} - \frac{1}{m(m-1)+2} \left[\frac{3}{2} - \frac{m-2}{2m(2m-1)} \right] \\
&\quad - \frac{2}{n} \left[\frac{3}{2} - \frac{m-2}{2m(2m-1)} \right]
\end{aligned} \tag{16}$$

Hence the left side of Eqn. 4 can be written as:

$$\begin{aligned}
Q_w^s - Q_w^p &= 0.5 - \frac{1}{m(m-1)+2} \left[\frac{5}{2} + \frac{m-2}{2m(2m-1)} \right] \\
&\quad + \frac{m-2}{2m(2m-1)} + \frac{2}{n} \left[\frac{1}{2} - \frac{m-2}{2m(2m-1)} \right] \\
&= 0.5 + \frac{m^3 - 13m^2 + 8m - 2}{2m(2m-1)[m(m-1)+2]} + \frac{2}{n} \frac{m^2 - m + 1}{m(2m-1)} \\
&= 0.5 + \frac{m^3 - 13m^2 + 8m - 2}{2m(2m-1)[m(m-1)+2]} + \frac{2[(m-1)^2 + m]}{nm(2m-1)}
\end{aligned} \tag{17}$$

The first and third terms of Eqn. 17 are always positive for $m \geq 3$. The second term reaches its lowest value at $m = 3$ and the lowest value is -0.2833 , which when combined with the first term of the equation, is still positive. Hence overall, $Q_w^s - Q_w^p > 0$, for all $m \geq 3$, $n \geq 3$. Since inequality 4 holds, maximizing weighted modularity will not merge adjacent cliques, and it will always detect the smallest clique as a single community.