Generalized Modularity for Community Detection

Mohadeseh Ganji[1,3], Abbas Seifi[1], Hosein Alizadeh[2], James Bailey[3], and Peter J. Stuckey[3]

[1] Amirkabir University of Technology, Tehran, Iran,
aseifi@aut.ac.ir,
[2] Iran University of Science and Technology, Tehran, Iran,
halizadeh@iust.ac.ir,
[3] NICTA, Victoria laboratory, Department of Computing and Information Systems, University of Melbourne, Melbourne, Victoria
sghasempour@student.unimelb.edu.au, {baileyj,pstuckey}@unimelb.edu.au

Abstract. Detecting the underlying community structure of networks is an important problem in complex network analysis. Modularity is a well-known quality function introduced by Newman, that measures how vertices in a community share more edges than what would be expected in a randomized network. However, this limited view on vertex similarity leads to limits in what can be resolved by modularity. To overcome these limitations, we propose a generalized modularity measure called GM which has a more sophisticated interpretation of vertex similarity. In particular, GM also takes into account the number of longer paths between vertices, compared to what would be expected in a randomized network. We also introduce a unified version of GM which detects communities of unipartite and (near-)bipartite networks without knowing the structure type in advance. Experiments on different synthetic and real data sets, demonstrate GM performs strongly in comparison to several existing approaches, particularly for small-world networks.

Keywords: Community detection, Modularity, Generalized modularity, Vertex similarity, Resolution limit

1 Introduction

As many real-world systems can be represented by networks, much research has focused on analysing networks and finding underlying useful structural patterns. Examples include social and biological networks [1, 2], in which vertices represent individuals or proteins and edges represent communications or interactions.

Among complex network analysis approaches, community detection is an important task which aims to find groups of vertices which could share common properties and/or have similar roles within the network [3]. This might reveal friendship communities in a social network or an unexpected hard-to-predict community structure in a biological dataset.

Two important network structures covered in the literature are unipartite and bipartite networks. In unipartite networks like social networks [1], the assumption is connections within communities are dense and connections between communities are sparse. However, some real networks are bipartite which means they can be partitioned into two clusters such that no two vertices within the same cluster are adjacent [4]. People attending events [5] is one example of a bipartite network. In addition, there are some real networks with near-bipartite properties. In these networks, there are some connections inside the two communities but they are fewer than between-community connections. Networks of sexual relationships are an example of near-bipartite networks.

Among community detection criteria, *modularity* [6] is one of the most important because according to [7], "Modularity has the unique privilege of being at the same time a global criterion to define a community, a quality function and the key ingredient of the most popular method of graph clustering." After its introduction, modularity was rapidly adopted and physicists, computer scientists, and sociologists have all developed a variety of heuristic algorithms to optimize modularity. They are based on greedy algorithms [8] spectral methods [9], mathematical optimization [10] and other strategies [11, 7].

Given an un-weighted undirected network $G(V, E)$, let d_i be degree of vertex i, m be total number of edges and A_{ij} be an element of the adjacency matrix which takes value 1 if vertices i and j are connected and 0 otherwise. Suppose the vertices are partitioned into communities such that vertex i belongs to community C_i. Then the modularity of the partition is defined by equation (1).

$$Q = \frac{1}{2m} \sum_{i,j} [A_{ij} - \frac{d_i d_j}{2m}] \delta(C_i, C_j) \tag{1}$$

The matrix of elements $A_{ij} - \frac{d_i d_j}{2m}$ is called the modularity matrix which is denoted by W. The modularity matrix records the difference between the number of the edges connecting each pair of vertices and the expected number of edges in a randomly distributed network of the same size with the same vertex degree sequence (in the rest of this paper we call it a randomized network). If the number of edges between i and j is the same as what is expected in the randomized network, the corresponding element of the modularity matrix is zero. Hence, nonzero values of the modularity matrix represent deviation from randomness. The coefficient $\frac{1}{2m}$ normalizes modularity to the interval [-1,1].

For calculating the modularity of a network partition, one adds up the modularities between each pair of vertices that lie in the same community. In equation (1), $\delta(C_i, C_j)$, the Kronecker delta function performs this task by limiting the summation to just over vertex pairs of the same community. The Kronecker function has the value 1 if its arguments are equal and 0 otherwise.

Brandes *et al* [12] showed that finding a clustering with maximum modularity is an NP-hard problem. However, researchers have tackled community detection using exact and approximation methods for modularity maximization. Among exact methods, Aloise *et al* [10] introduced a column generation model which can find communities of optimal modularity value for problems of up to 512

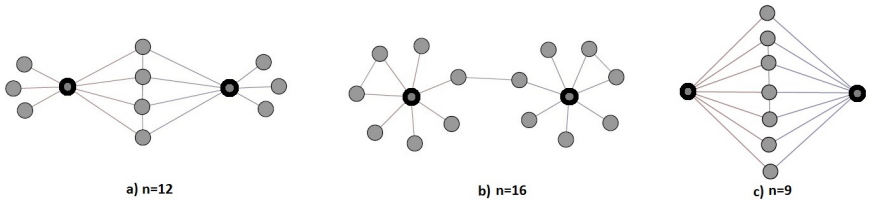

Fig. 1. Examples for neglecting neighbours by Modularity

vertices. Among the wide range of approximation algorithms for modularity maximization, the hierarchical iterative two phase method of Blondel *et al* [8] is one of the best (see [7]). In the first phase, communities are merged together only if this improves the modularity value of the partition, whilst the second phase reconstructs the network whose nodes are communities of the previous phase.

Limitations of Modularity Although modularity performs effectively in many cases, some limitations have been noted about its performance [7, 13]. First, modularity has a restricted interpretation of vertex similarity. Figure 1 illustrates this problem using hand-made examples. All structures shown in this figure have 17 edges but the number of vertices are different. Degrees of the bold vertices are the same and equal to 7 in all cases and they do not share any edges together. Hence, in all three cases, the modularity value between the two bold vertices is equal while one can clearly see that the structures are very different.

Figure 1 illustrates that modularity's interpretation of vertex similarity is limited to sharing common edges. While in reality, in addition to sharing an edge, pairs of vertices are more similar and more likely to lie in the same community when they have many neighbours in common. This is exactly one of the basic vertex similarity measures called common neighbour index [14]. There are also some other variations of vertex similarity measures based on number of common neighbours and paths of longer lengths [15, 16, 14, 7].

We propose a new measure of community detection called generalized modularity (GM) which extends modularity's assumption about similar vertices. In addition to common edges, GM takes into account common neighbours and longer paths between vertices and compares the number of these paths to a randomly distributed network to achieve a more comprehensive interpretation of vertex similarity.

Although in the literature, some research has tried to detect communities based on a vertex similarity concept [17, 1], these approaches have mostly failed to take advantage of modularity's strength in noticing common edges between vertices. The vertex similarity probability (VSP) model of Li and Pang [17] is one such approach which is just based on common neighbours of vertices but doesn't notice common edges or relations of longer lengths.

In other work, Alfalahi *et al* [1] proposed the concept of vertex similarity for modularity. They construct a virtual network which is initially the same as the original network. Then, vertices with higher Jaccard [15] similarity index (which is based on common edge and common neighbour concepts) than a pre-defined

threshold, would have an extra edge in the virtual network. Finally, modularity maximization is applied to the virtual network in order to find communities. Although this approach aims to add vertex similarity concepts into modularity's common edge criterion, paths of longer length than two are neglected. Also, the consideration of similarity between vertices strongly depends on the choice of threshold value which divides similarity status of vertices into "similar" or "not-similar". In generalized modularity the interpretation of vertex similarity is not limited to 0 and 1. In addition, as opposed to Alfalahi's approach, GM benefits from the comparison to random graphs for measuring vertex similarity. In this sense, GM's interpretation of vertex similarity is close to Leicht *et al* [16] who proposed a vertex similarity index based on comparison to a randomized network, though there are basic differences in context and approach of comparison.

The second limitation of modularity, the resolution limit, arises from its null model. It causes the systematic merging of small communities into larger modules, even when the communities are well defined and loosely connected to each other [13]. Fortunato and Barthelemy in [13] and Fortunato in [7] discussed this issue in more detail. According to [7], in the modularity definition, the weak point of the null model is the implicit assumption that each vertex can communicate with every other vertex of the network. This is however questionable, and certainly wrong for large networks like the Web graph. To address the resolution limit problem, multiresolution versions of modularity have been introduced [18] which allow users to specify their target scale of communities. The choice of correct value for this scale parameter is still an issue with these approaches.

However, by considering longer paths, GM moderates the questionable assumption of modularity's null model. Because expecting network members to be able to share a neighbour with others is a more reasonable assumption. Even more realistic is the possibility of existence of paths with short lengths between members of a network, in particular, networks with the small-world property. According to Watts *et al* [19], small-world networks are those in which the typical distance L between two randomly chosen vertices grows proportionally to the logarithm of the size of the network. This means the transition from one vertex to any other vertex of the network requires just a few hops. It has been shown that a wide range of real-world complex networks like social networks, the connectivity of the Internet, wikis, collaboration networks and gene networks exhibit small-world network characteristics. In addition to small-world networks, Watts and Strogatz showed that in fact many real-world networks have a small average shortest path length between vertices [19]. Thus, although GM is a global criterion and considers the whole network for defining communities, the small-world property of real networks supports the assumption behind its null model.

Modularity maximization and most community detection criteria are designed for unipartite networks in which edges inside communities are more dense. In near-bipartite networks, however, connections between communities are denser than inside them and modularity maximization cannot find correct communities because it aims to minimize the number of edges between communities. Although there are community detection methods for bipartite networks

like modularity minimization and some others [4], they require knowing the type of the data in advance. This problem is more important when it comes to near-bipartite networks, since the identification of such networks is more difficult. In this paper, we also propose a unified version of generalized modularity called UGM which can detect communities in unipartite, bipartite and near-bipartite networks without knowing the type of the network structure.

Briefly, the main contributions of this paper are:

- Extending the "modularity" community detection quality function and proposing a new criterion named Generalized Modularity (GM) which takes advantage of vertex similarity and longer paths between vertices.
- Proposing a more realistic null model in comparison to modularity, which enables generalized modularity to perform better than modularity in small-world data sets with communities of different scales.
- Introducing a unified version of the generalized modularity measure (UGM) which is able to detect communities in unipartite, bipartite and near-bipartite networks without any pre-knowledge about the structure of the data.
- Experimental comparison of the GM and UGM methods with some state of the art approaches and statistically demonstrating their high performance.

2 Generalized modularity (GM)

The core concept of our proposed generalized modularity measure is to extend modularity to take advantage of indirect communications between vertices.

According to the definition of modularity, a pair of vertices is likely to be in the same community if they share more edges than what is expected from a randomly distributed network. Pairs of vertices can be also similar to each other based on the number of their shared neighbours [14, 7]. In generalized modularity we believe that sharing more neighbours than what is expected (in a randomized network) also expresses how likely it is for the pair to lie in the same community. Likewise, two vertices are more likely to be in same community if they have more paths of length three or more, than the corresponding expected number in a randomize network. Hence, generalized modularity is inspired by the concept of vertex similarity while preserving the basic idea of modularity.

The general form of the proposed GM measure is presented in equation (2) which given a partition, adds up the elements of the W^{GM} matrix for pairs of the same communities. The generalized modularity matrix W^{GM} is the weighted summation of $W^{(\ell)}_{norm}$s (equation (3)) which are normalized generalized modularity matrices of level ℓ which means just relations with paths of length ℓ are considered. α_ℓ represents the weight of contribution of $W^{(\ell)}_{norm}$ in W^{GM}.

$$Q_{GM} = \sum_{i,j \in V} W^{GM}_{i,j} \delta(C_i, C_j) \tag{2}$$

$$W^{GM} = \sum_{\ell=1}^{\infty} \alpha_\ell W^{(\ell)}_{norm} = \sum_{\ell=1}^{\infty} \alpha_\ell \frac{W^{(\ell)}}{||N^{(\ell)}||} = \sum_{\ell=1}^{\infty} \alpha_\ell \frac{[N^{(\ell)} - E^{(\ell)}]}{||N^{(\ell)}||} \tag{3}$$

$N^{(\ell)}$ is the matrix representing the number of simple paths (paths containing no loops) of length ℓ between vertices. The matrix of $N^{(\ell)}$ is equal to the adjacency matrix power to ℓ, (A^ℓ), for $\ell = 1, 2$. $||N^{(\ell)}||$ is the entry-wise 1-norm of matrix $N^{(\ell)}$ which is summation of absolute values of the matrix elements. The matrix $E^{(\ell)}$ represents the expected number of paths of length ℓ between vertices in a randomized network. We can normalize each term by dividing it by the total number of paths of corresponding length which is denoted by $||N^{(\ell)}||$. According to equation (3), $W^{(1)}$ is exactly the same as the modularity matrix of Newman [6] while the matrix $W^{(2)}$ is the existing number of common neighbours (relations with paths of length 2) between vertices minus the expected number of such common neighbours in a corresponding randomized network. Other terms are also defined likewise.

The expected number of paths of length one between i and j is calculated by multiplying the number of edges connected to i (degree of vertex i) by the probability that an edge ends in j which is $d_j/2m$. By applying a similar approach, we calculate the expected terms in $W^{(2)}$ and $W^{(3)}$ for a pair of vertices in an un-weighted network. Note that the direct edges between two vertices cannot participate in any path of length 2 and 3 between them. So, in equation (4), apart from $d_i d_j/2m$ expected connections between i and j, we expect $(d_i - \frac{d_i d_j}{2m})$ remaining edges of i to contribute in simple paths of longer lengths. For these edges, the probability to be linked to the intermediate vertex k is $d_k/2m$ and then an edge from the set of $d_k - 1$ remaining edges of k must be linked to j with probability of $(d_j - \frac{d_i d_j}{2m})/2m$. Since the probability of existence of edges between vertices are independent to each other, the probability of existence of a path of length ℓ simply equals the multiplication of probabilities of each of its ℓ edges. Finally, as intermediate vertex k can be any vertex of the network except i and j, we have a summation over all possible ks.

$$E^{(2)}_{i,j} = \sum_{k \in V \setminus \{i,j\}} \left[\frac{(d_i - \frac{d_i d_j}{2m}) d_k}{2m} \right] \left[\frac{(d_k - 1)(d_j - \frac{d_i d_j}{2m})}{2m} \right] \tag{4}$$

$$W^{(2)}_{ij} = N^{(2)}_{ij} - E^{(2)}_{i,j} = (A^2)_{ij} - \frac{(d_i - \frac{d_i d_j}{2m})(d_j - \frac{d_i d_j}{2m})}{(2m)^2} \sum_{k \in V \setminus \{i,j\}} d_k(d_k - 1) \tag{5}$$

Similarly, we can calculate the expected value for paths of length 3 which vertices i and j are connected through two intermediate vertices k and k'.

$$E^{(3)}_{i,j} = \sum_{k,k' \in V \setminus \{i,j\}} \left[\frac{(d_i - \frac{d_i d_j}{2m}) d_k}{2m} \right] \left[\frac{(d_k - 1) d_{k'}}{2m} \right] \left[\frac{(d_{k'} - 1)(d_j - \frac{d_i d_j}{2m})}{2m} \right] \tag{6}$$

In the calculation of paths of length 3 as opposed to the two previous cases, there is a possibility for loops which are illustrated in Figure 2. Among these four topologies, just Figure 2-a is considered in the calculation of term $W^{(3)}_{ij}$, because

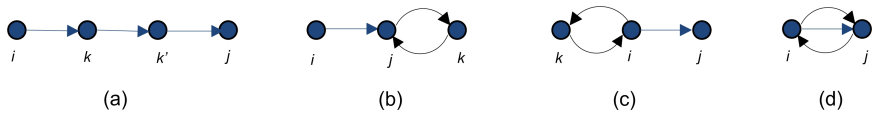

Fig. 2. Four different possible topologies for paths of length 3 between i and j

the existence of the other three paths is dependent on the existence of a common edge between i and j which we already considered in the calculation of $W_{ij}^{(1)}$. As matrix (A^3) counts all four topologies, we use matrix $(A^3)'$ of equation (7) which just represents the number of simple paths of length 3 between vertices. Hence, the third term of the generalized modularity is equal to the equation (8).

$$(A^3)'_{ij} = (A^3)_{ij} - A_{ij}(d_i + d_j - 1) \tag{7}$$

$$W_{ij}^{(3)} = N_{ij}^{(3)} - E_{i,j}^{(3)} = (A^3)'_{ij} - \left[\frac{(d_i - \frac{d_i d_j}{2m})(d_j - \frac{d_i d_j}{2m})}{(2m)^3}\Big(\sum_{k \in V \setminus \{i,j\}} d_k(d_k - 1)\Big)^2\right] \tag{8}$$

The number of terms in generalized modularity increases according to the path lengths considered, however, paths of length more than 3 are more complicated as the number of possible topologies and non simple paths rapidly increases. In addition, intuitively, it seems they would have smaller importance weight (α_l) than the first couple of terms. Therefore, in this paper, we limit generalized modularity to its first three terms which are related to paths of length $(\ell = 1, 2, 3)$.

2.1 Comparison to Modularity

Although our GM quality function was initially inspired by modularity, extending the measure to consider neighborhoods with longer paths leads to improvements in several aspects.

First, GM is more comprehensive in its interpretation of similarity as it considers vertex similarity as well. Therefore, when edge related properties are still the same (as in Figure 1), GM can detect communities better than modularity since it uses common neighbours and the neighborhood of longer paths as well.

To illustrate how well generalized modularity can reveal the underlying community structures, we use visualization. The visual assessment of tendency (VAT) [20], is a tool for revealing the number of clusters. It uses the logic of Prim's algorithm and reorders the objects of symmetric square dissimilarity matrix R to show the number of clusters by squared shaped dark blocks along the diagonal in the VAT image. We scaled each element of modularity and GM matrices to $(-W_{ij} + 1)/2$ to ensure elements are in interval [0,1] and then we used them as dissimilarity matrix for VAT. Figure 3 presents VAT images of modularity

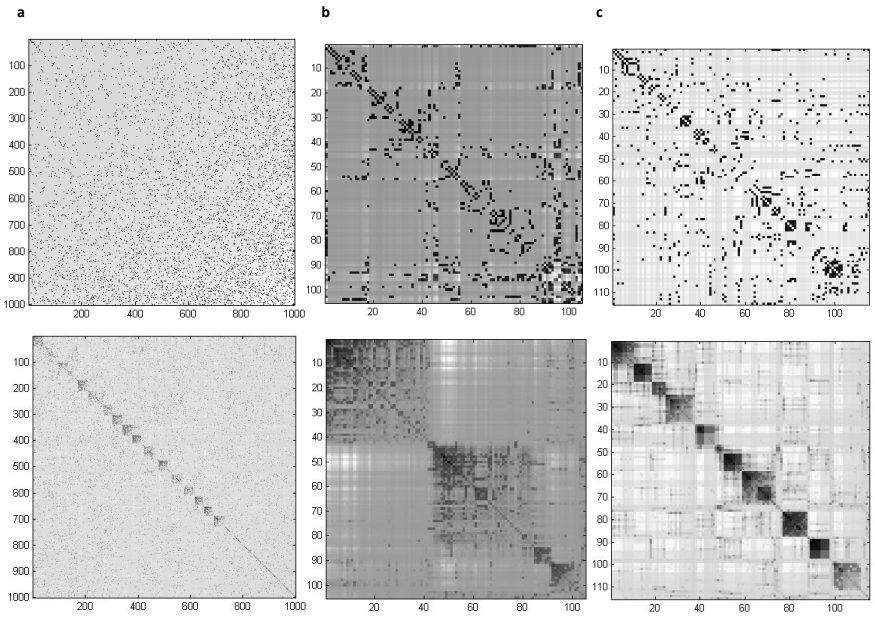

Fig. 3. VAT image of modularity matrix (top images) and generalized modularity matrix (bottom images) for three data sets (a) LFR, (b) Political Books, (c) American Football. Dark blocks in VAT images of GM correspond to communities in the data.

and GM ($\alpha_1, \alpha_2, \alpha_3 = (0.25, 0.5, 0.25)$) for an LFR data set (which is a community detection synthetic benchmark proposed by Lancichinetti [21]) and also two real-world data sets. In this Figure, modularity's VAT image does not reveal the community structure of the data sets while dark blocks in GM's VAT image effectively distinguish community structures. A similar trend was also observed for the other real-world and artificial data sets used in our experiments.

The second advantage of GM is related to community detection in data sets of multi-scale communities. As explained about the resolution limit of modularity, it is related to the assumption/interpretation of vertex similarity in modularity. Modularity expects two similar vertices to share an edge while this is not reasonable, in the sense that, in large networks each vertex cannot know about all other vertices of the network. Although one cannot expect vertices to be able to directly communicate with all other members of the network, it is more sensible to expect them to be able to share a neighbour, or even more realistic, to expect them to have a longer path to other members of the network. This idea is powerful when it comes to small-world networks which are discussed in introduction and proved to have a small diameter [19]. Even in large networks with this property, although each vertex cannot communicate directly to all others, it is related to all other vertices with comparatively very short paths. This fact supports the

more realistic underlying assumption in the definition of generalized modularity measure. So that in data sets with different community sizes in particular those of small-world networks, GM can achieve higher performance than modularity. However, GM is not completely free of resolution limit problems. Because GM is a global optimization criterion which considers the whole network for defining communities and resolution limit seems to be a general problem for all methods with a global optimization goal [7].

The third advantage of our generalized modularity is discussed in Section 2.2 which introduces a unified version of the generalized modularity measure.

2.2 Unified generalized modularity (UGM)

As explained in the introduction, modularity maximization cannot detect communities of bipartite networks. However, a specialised version of GM (in equation (9)) using the difference between the number of common neighbours and the expected such numbers in a randomized network, can detect communities in uni-partite, near-bipartite and bipartite networks without pre-knowledge of the network type. In equation (9), $W_{ij}^{(2)}$ is same as the term defined in equation (5).

$$Q_{UGM} = \frac{1}{||A^2||} \sum_{i,j \in V} W_{i,j}^{(2)} \delta(C_i, C_j) \tag{9}$$

In unipartite models, the basic community detection principle is "edges inside a community are dense and outside are sparse." Consider a partition of a unipartite network (Figure 4-a) detected by maximizing Q_{UGM}. As explained before, the elements of $W^{(2)}$ are higher for pairs of vertices who have more common neighbours than what is expected in a randomized network. As a general property of unipartite networks, vertices have neighbours of the same community. Hence, cluster members detected by Q_{UGM} have common neighbours which lie within the same community. This means density of connections inside communities is much more than the edge density between communities. Therefore, the Q_{UGM} criteria is completely aligned with properties of communities in unipartite networks.

However, in bipartite (near-bipartite) networks, all (most) common neighbours of members of the same cluster are definitely (probably) located in the opposite community. In these networks, the basic community detection principle is "edges inside communities are sparse and outside are dense."

Consider a partition of a bipartite or near-bipartite network (Figure 4-b) which is achieved by maximizing Q_{UGM} without any pre-knowledge about the type of the network. Q_{UGM} maximization, assigns vertices with more common neighbours— than the expected number in a randomized network— to the same community. This leads to high density of between-community edges because members of a community share neighbours which belong to the other community. Hence, Q_{UGM} maximization in bipartite and near-bipartite networks

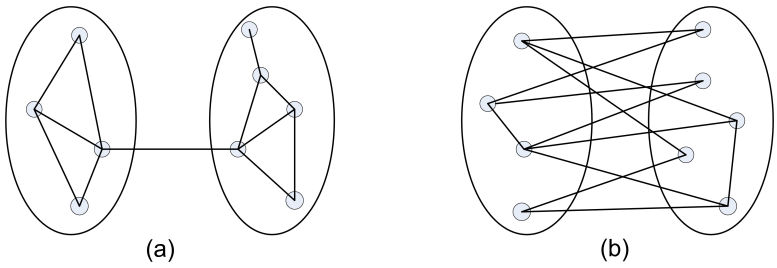

Fig. 4. Example of a unipartite network (left side) and a near-bipartite network (right side)

finds communities with sparse inter-community connections and dense between-community links. Therefore, the unified generalized modularity (UGM) measure is able to detect communities in unipartite, near-bipartite and bipartite networks without pre-knowledge of the network's structure.

2.3 Finding communities based on the GM quality function

Similar to modularity maximisation, finding a partition with maximum generalized modularity is also an NP-hard problem. However, as GM can be represented as a matrix (similar to the modularity matrix), heuristic and exact algorithms of modularity maximization can be reused for partitioning data based on GM.

In this paper, we use an agglomerative community detection algorithm similar to one of Blondel *et al* in [8] which is also discussed in the introduction section. This algorithm considers each vertex as a community initially and then merges these small communities in a way that increases the GM value of the partition. It then updates the network information based on new communities and starts the next iteration and continues until no further improvement is possible.

3 Experiments

In this section, we present empirical analysis of generalized modularity and compare it with some state of the art approaches in the literature. All experiments are done on a PC with core i7 CPU 3.40 GHz and 16GB RAM.

Data sets: In order to present a comprehensive comparison, we used four different categories of data sets which are common in the literature.

- We used LFR data sets proposed in [21]. In LFR data sets, degrees follow a power-law distribution $p(d) = d^{-\alpha}$ with parameter α and the community size a power-law distribution with parameter β. A mixing parameter, μ is the proportion of external degree for each vertex. Based on the original LFR data set in [21] we fixed α and β to be 2 and 1 respectively.

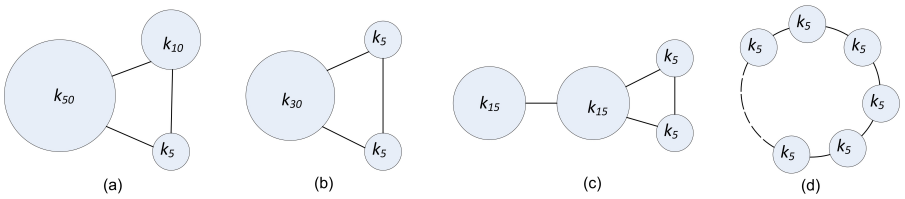

Fig. 5. Synthetic data sets for testing the resolution limit

- To address the resolution limit of modularity, there are some structures of networks proposed in [13] where modularity fails to detect the underlying communities correctly. Similarly, we used four synthetic data sets with structures of Figure 5. In this figure, each circle represents a clique or complete graph which is denoted by k. For instance, k_{50} is a complete graph or clique of 50 vertices. Figure 5-d shows a circle of 30 cliques of size 5 [13].
- Four real-world data sets including Zachary Karate Club [22], Books about US Politics, American College Football [2] and Sampson's monastery data set [23] were selected. These data sets were chosen because their ground truth tags are known and we can measure performance by comparing the results to the ground truth.
- We also used the South Women data set [5] as a real bipartite network. We also generated random near-bipartite networks for further experiments.

Comparison measure: Since we have the real ground truth of the data sets, for evaluating quality of partitioning, we use the Normalized Mutual Information of equation (10) which is proposed by Danon *et al* [24].

$$I_{norm}(A,B) = \frac{-2\sum_{i=1}^{CA}\sum_{j=1}^{CB} N_{ij}\log(N_{ij}N/N_{i.}N_{.j})}{\sum_{i=1}^{CA} N_{i.}\log(N_{i.}/N) + \sum_{j=1}^{CB} N_{.j}\log(N_{.j}/N)} \tag{10}$$

In equation (10), A represents the real communities and B represents the detected communities while CA and CB are the number of communities in A and B respectively. In this formula, N is the confusion matrix with rows representing the original communities and columns representing the detected communities. The value of N_{ij} is the number of common vertices that are in the original community i but found in community j. The sum over the ith row is denoted by $N_{i.}$ and the sum over the jth column is denoted by $N_{.j}$

In the rest of this section, first, we examine the unified version of generalized modularity. Then we discuss the choice of model parameters α_l based on a set of training experiments. Finally, we report the comparison with other approaches.

3.1 Testing unified generalized modularity

We tested our unified generalized modularity of equation (9) on several real and artificial unipartite, bipartite and near-bipartite networks to evaluate its performance. We compared the proposed UGM model with modularity and VSP

Table 1. Comparison of UGM to other algorithms on unipartite networks

Data sets	#vertices	#cluster	UGM	Modularity	VSP
LFR10K-0.3	10000	24	**1.00**	**1.00**	**1.00**
LFR10K-0.4	10000	23	**1.00**	0.98	**1.00**
LFR10K-0.5	10000	22	**1.00**	0.97	0.99
LFR15K-0.3	15000	19	**1.00**	0.99	**1.00**
LFR15K-0.4	15000	20	**1.00**	0.99	**1.00**
LFR15K-0.5	15000	19	**1.00**	0.92	0.99
Karate Club	34	2	**0.83**	0.64	0.12
PolBooks	105	3	**0.54**	**0.54**	**0.54**
Football	115	12	0.17	**0.20**	0.16
Samson T4	18	4	**0.64**	0.59	**0.64**
Samson T1-T5	25	2	0.60	0.57	**0.62**
Figure 5-a	65	3	**1.00**	0.88	0.79
Figure 5-b	40	3	**1.00**	0.87	0.63
Figure 5-c	40	4	**0.93**	**0.93**	**0.93**
Figure 5-d	150	30	0.86	**0.89**	0.86
p-Value			baseline	0.0209	0.0588

model of Li and Pang [17]. We chose the VSP model since it is one of the few unified community detection algorithms and is expected to detect communities without knowing the network structure type. For the sake of consistency, we used the same algorithm (greedy algorithm of Blondel *et al* [8]) to maximize the three examined measures. Information of the data sets are presented in three first columns of Table 1. The normalized mutual information index achieved by UGM, modularity and VSP are shown in the remaining columns respectively. The real-world and artificial data sets were introduced earlier. We used mixing parameter 0.3–0.5 and generated large LFR data sets with average degree of 30 and maximum degree of 70. In Table 1, the LFR data sets are named based on their size and mixing parameter. The Samson data set represents affect relations among the novices in a New England monastery which were measured at five moments in time. The first Samson data set in Table 1 is just based on measurements on the fourth moment and the second data set is based on all measurements at five moments. Based on results of Table 1, UGM outperforms modularity and the VSP method in most data sets. Friedman statistical test results are also reported with null hypothesis of no difference in performance.

We also tested UGM on the bipartite network of Southern Women [5] who participated in social events. The proposed UGM measure 100% correctly detects the two groups in this bipartite network without knowing the structural type in advance.

For further comparison, we also generated near-bipartite networks. In randomly generated near-bipartite networks, each vertex shares an edge with a (randomly chosen) member of the same community with the probability P_{in} and the minimum degree of vertices is chosen uniformly from range of 1 and corresponding community size. In Figure 6, performance of GM, modularity and

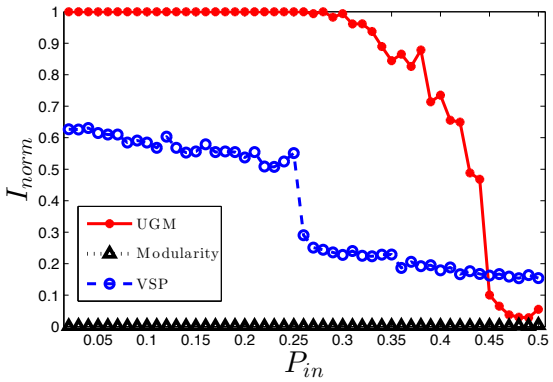

Fig. 6. Sensitivity analysis of methods on randomly generated near-bipartite network which has communities of size 500 and 300.

VSP are analysed over the change in P_{in} parameter. By increasing P_{in}, the data set becomes less and less bipartite as the percentage of inter-community edges increases. Figure 6 illustrates that the high performance of GM is maintained on near-bipartite networks up until they become close to unipartite.

As explained earlier, when not knowing the type of data in advance, modularity maximization fails to detect communities and performs poorly on bipartite or near-bipartite data sets.

3.2 Training parameters of generalized modularity

According to the definition of GM in equation (3), parameters α_1, α_2 and α_3 determine the importance of each term to the generalized modularity. We can tune these parameters based on use of training data.

For training the parameters, we chose the LFR benchmark data [21] because we can generate it in different sizes and features and it properly simulates real world [7]. Similar to Lancichinetti *et al* [21], we used LFR data sets of size 1000 while the size of communities is between 20 and 100 and the average degree is 20 and the maximum degree is set to be 50. The mixing parameter ranges from 0.1 to 0.5 and we also used a LFR with mixing parameter 0.7 in which communities are not well defined and community detection seems to be more challenging.

In the generalized modularity matrix of equation (3), without loss of generality, we assumed α_1, α_2 and α_3 to be between 0 and 1 and α_3 to be equal to $1-\alpha_1-\alpha_2$. We considered 5 levels for each parameter α_1 and α_2 and examined all 15 unique combinations of three parameters of our GM measure. Tabel 2 presents the average I_{norm} over all training data sets for each combination. Based on results reported in Table 2, all combinations of GM which include $W^{(1)}_{norm}$ ($\alpha_1 > 0$), on average, perform better than modularity. It shows that presence of $W^{(1)}_{norm}$ is essential but also using $W^{(2)}_{norm}$ and $W^{(3)}_{norm}$ improves the results. In Table 2

Table 2. Empirical training for parameter configuration for generalized modularity

		α_2				
		0	0.25	0.5	0.75	1
α_1	0	0.831	0.844	0.850	0.852	0.861
	0.25	0.866	0.877	**0.881**	0.877	
	0.5	0.873	0.877	0.878		
	0.75	0.869	0.874			
	1	0.866				

it is shown that the combination of $0.25W^{(1)}_{norm} + 0.5W^{(2)}_{norm} + 0.25W^{(3)}_{norm}$ has the best performance on average over our train data sets. Therefore, we use this combination in subsequent experiments for comparison with other methods.

3.3 Comparison with other methods

In this section, we compare our trained GM community detection model with some state of the art models in the literature. We compare GM with the modularity based algorithm of Blondel *et al* [8][4] and the vertex similarity probability (VSP) model of Li and Pang [17]. In order to be consistent in the experiments, we used the same algorithm of Blondel *et al* [8] for optimizing VSP and GM models. Table 3 reports average I_{norm} value of 10 independent runs.

Table 3 demonstrates that GM performs much better than the other methods over different data sets. It performs very strongly in large data sets. Besides, it detects communities of real world networks more precisely. Note that our results for the VSP model do not exactly match with experiments reported in [17], possibly due to differences in optimization procedure (which wasn't described in that paper). GM also detects small communities in data sets of Figures 5-a and 5-b. The reason that GM couldn't outperform modularity in the data set of Figure 5-d is because this data set has a large diameter in comparison to size of the network which shows its structure is very different from small-world networks. The result of a pairwise Friedman statistical test is also reported at the bottom of Table 3. The null hypothesis of this test is two algorithms have no significant difference in their performance. This hypothesis is rejected based on the very small p-values, indicating statistically significant differences in performance.

4 Conclusion

We have proposed a generalized modularity criterion (named GM) for community detection in complex networks. Generalized modularity extends the interpretation of modularity by taking into account paths between vertices rather than

[4] We also compared our method with modularity-based algorithms of Danon [24] and Newman [11] but as method of Blondel *et al* [8] outperforms the other two, we just report Blondel *et al* [8] here.

Table 3. Comparison of GM to other state of the art models of community detection

Data sets	#vertices	#clusters	GM	Modularity	VSP
LFR10K-0.3	10000	24	**1.00**	**1.00**	**1.00**
LFR10K-0.4	10000	23	**1.00**	0.98	**1.00**
LFR10K-0.5	10000	22	**1.00**	0.97	0.99
LFR15K-0.3	15000	19	**1.00**	0.99	**1.00**
LFR15K-0.4	15000	20	**1.00**	0.99	**1.00**
LFR15K-0.5	15000	19	**1.00**	0.92	0.99
Karate Club	34	2	**1.00**	0.64	0.12
PolBooks	105	3	**0.56**	0.54	0.54
Football	115	12	**0.20**	**0.20**	0.16
Samson T4	18	4	**0.69**	0.59	0.64
Samson T1-T5	25	2	0.60	0.57	**0.62**
Figure 5-a	65	3	**1.00**	0.88	0.79
Figure 5-b	40	3	**1.00**	0.87	0.63
Figure 5-c	40	4	**0.93**	**0.93**	**0.93**
Figure 5-d	150	30	0.86	**0.89**	0.86
p-Value			baseline	0.0039	0.0196

just common edges. The modelling of existence of paths between vertices enables GM to deliver better performance especially in small-world networks with different community sizes and it can also work on bipartite and near-bipartite networks. Although GM improves the resolution limit of modularity especially in small-world networks, still it can have this problem and approaches to solve it are a clear direction for future work.

Acknowledgements

NICTA is funded by the Australian Government through the Department of Communications and the Australian research Council through the ICT center of excellence program. James Baileys work is supported by an ARC Future Fellowship (FT110100112).

References

1. K. Alfalahi, Y. Atif, and S. Harous, "Community detection in social networks through similarity virtual networks," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining.* ACM, pp. 1116–1123, 2013.
2. M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.
3. G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 17, no. 6, pp. 734–749, 2005.

4. M. J. Barber and J. W. Clark, "Detecting network communities by propagating labels under constraints," *Physical Review E*, vol. 80, no. 2, p. 026129, 2009.
5. A. Davis, B. B. Gardner, and M. R. Gardner, *Deep south.* University of Chicago Press, 1969.
6. M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.
7. S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010.
8. V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.
9. M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical review E*, vol. 74, no. 3, p. 036104, 2006.
10. D. Aloise, S. Cafieri, G. Caporossi, P. Hansen, S. Perron, and L. Liberti, "Column generation algorithms for exact modularity maximization in networks," *Physical Review E*, vol. 82, no. 4, p. 046112, 2010.
11. M. E. Newman, "Fast algorithm for detecting community structure in networks," *Physical review E*, vol. 69, no. 6, p. 066133, 2004.
12. U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner, "On modularity clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 2, pp. 172–188, 2008.
13. S. Fortunato and M. Barthelemy, "Resolution limit in community detection," *Proceedings of the National Academy of Sciences*, vol. 104, no. 1, pp. 36–41, 2007.
14. D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American society for information science and technology*, vol. 58, no. 7, pp. 1019–1031, 2007.
15. C. Blundo, E. De Cristofaro, and P. Gasti, "Espresso: Efficient privacy-preserving evaluation of sample set similarity," in *Data Privacy Management and Autonomous Spontaneous Security.* Springer, 2013, pp. 89–103.
16. E. Leicht, P. Holme, and M. E. Newman, "Vertex similarity in networks," *Physical Review E*, vol. 73, no. 2, p. 026120, 2006.
17. K. Li and Y. Pang, "A unified community detection algorithm in complex network," *Neurocomputing*, vol. 130, pp. 36–43, 2014.
18. A. Arenas, A. Fernandez, and S. Gomez, "Analysis of the structure of complex networks at different resolution levels," *New Journal of Physics*, vol. 10, no. 5, p. 053039, 2008.
19. D. J. Watts and S. H. Strogatz, "Collective dynamics of small-worldnetworks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
20. J. C. Bezdek and R. J. Hathaway, "VAT: A tool for visual assessment of (cluster) tendency," in *Proc. IJCNN*, pp. 2225–2230, 2002.
21. A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Physical Review E*, vol. 78, no. 4, 2008.
22. W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, pp. 452–473, 1977.
23. S. F. Sampson, "A novitiate in a period of change: An experimental and case study of social relationships," Ph.D. dissertation, Cornell University, 1968.
24. A. D.-G. Leon Danon and A. Arenas, "The effect of size heterogeneity on community identification in complex networks," *Journal of Statistical Mechanics: Theory and Experiment*, p. P11010, 2006.