# Loops and multiple edges in modularity maximization of networks

Sonia Cafieri,* Pierre Hansen,† and Leo Liberti‡

*LIX, École Polytechnique, F-91128 Palaiseau, France*

(Dated: January 25, 2010)

## Abstract

The modularity maximization model proposed by Newman and Girvan for the identification of communities in networks works for general graphs possibly with loops and multiple edges. However, the applications usually correspond to simple graphs. These graphs are compared to a null model where the degree distribution is maintained but edges are placed at random. Therefore, in this null model there will be loops and possibly multiple edges. Sharp bounds on the expected number of loops, and their impact on the modularity, are derived. Then, building upon the work of Massen and Doye, but using algebra rather than simulation, we propose modified null models associated with graphs without loops but with multiple edges, graphs with loops but without multiple edges and graphs without loops nor multiple edges. We validate our models by using the exact algorithm for clique partitioning of Grötschel and Wakabayashi.

PACS numbers: 89.75.Hc, 87.23.Ge, 89.20Hh

*Electronic address: cafieri@lix.polytechnique.fr

†Electronic address: pierre.hansen@gerad.ca; Also at GERAD, HEC Montréal, Canada

‡Electronic address: liberti@lix.polytechnique.fr

## I. INTRODUCTION

Community detection is a topic of particular interest in the analysis of complex networks, which are often used to represent systems arising in a variety of fields, such as telecommunications, sociology, biology, computer science. Roughly speaking, edges joining pairs of vertices in a community should be more dense than elsewhere. This idea was made precise by Newman and Girvan [32], who proposed to compare the fraction of edges within a community to the expected fraction of edges in that community for a *null model* where the edges would be distributed at random. To make this model realistic, the degree distribution should be kept constant. This led to precise definitions of the *modularity* of a community, or cluster, and, summing over all communities, of the modularity of a partition.

So, modularity can be used as a function to assess the quality of a network partition as well as a basic concept in models and methods for the identification of communities in networks. Indeed, modularity maximization has spawned in recent years numerous methods to identify such communities. Most of them are heuristics. They are based, for example, on simulated annealing [20], extremal optimization [11], genetic search [38], dynamical clustering [4], multilevel partitioning [10], contraction-dilation [29], multistep greedy search [36], quantum mechanics [34] and a variety of other approaches [3, 7, 12, 24, 35, 37]. Newman [33] developed an agglomerative hierarchical clustering heuristic to maximize modularity. An efficient agglomerative hierarchical algorithm was proposed by Clauset et al. [8]. Newman [31] also developed a spectral method for divisive clustering with the modularity criterion, based on signs of the components of the first eigenvector of the so-called modularity matrix. Other approaches based on modularity maximization are within the framework of mathematical programming. Agarwal and Kempe [1] used mathematical programming heuristically followed by randomized rounding. Brandes et al. [5] used an integer programming formulation and an algorithm close to those of Grötschel and Wakabayashi [19] for clique partitioning. Recently, Xu et al. [39] proposed a model to maximize modularity exactly, based on a mixed-integer quadratic program with a convex relaxation. These exact algorithms apply only to small instances with about a hundred entities.

Despite the great success of modularity, several criticisms of the original modularity concept can be found in recent literature. Some authors showed, for example, that counter-intuitive results can be obtained for artificially constructed instances [5, 15]. Moreover, some

limit of resolution has been pointed out for the modularity criterion in [15]. In this work, Fortunato and Barthelemy gave some examples of datasets where, in the presence of large communities, small communities are undetected even if they are very dense. These criticisms also led to modifications to the modularity function [2, 24]. Recently, Good et al. [18] also discussed the resolution problem as well as another difficulty in using the modularity function, i.e. the exponential number of high-modularity partitions.

Another criticism of the modularity function has been put forward by Massen and Doye in [28]. They remarked that in the original modularity model the graphs corresponding to the null model will have loops (or self-edges) and possibly multiple edges. However, graphs occurring in applications are usually simple graphs, i.e. graphs without loops nor multiple edges. So a better estimate of the density of random edges will be obtained by excluding loops and multiple edges from the model. Massen and Doye [28] proposed a simulation approach to do this. Starting from the given network $G$, they transform it into a random graph using rewiring [27, 30]. The basic rewiring operation consists in replacing two randomly chosen disjoint edges $(i, j)$ and $(k, l)$ by the two edges $(i, k)$ and $(j, l)$, or by the two edges $(i, l)$ and $(j, k)$, provided this does not create multiple edges. Clearly, it does not create loops. Note that this operation does not change the degree distribution. This step is repeated a sufficient number of times on each edge for the graph obtained to be considered as random, which can be estimated from the clustering coefficient reaching equilibrium, i.e. having only small fluctuations. After obtaining such a random graph the procedure is repeated to yield a sufficiently large family of them. Probabilities of presence of an edge are then estimated from those observed in this family. Modularity maximization heuristics can then be applied. Note that if precise probabilities are requested, the family of random graphs without loops or multiple edges should be large.

In this paper we consider again the problem treated by Massen and Doye, but using algebra instead of simulation. We first discuss further, in Section II, the basic modularity maximization model and derive sharp bounds on the number of loops and their impact on modularity. Then we propose modified null models associated with graphs without loops but with multiple edges, graphs with loops but without multiple edges and graphs without loops nor multiple edges. We validate our models in Section III by using the exact algorithm for clique partitioning of Grötschel and Wakabayashi [19]. Brief conclusions are given in Section IV.

## II. MODULARITY OF SIMPLE GRAPHS

### A. Expected contribution of loops to modularity

Let $G = (V, E)$ be a *network* (or *graph*) with vertex set $V$ and edge set $E$. Its *order*, or number of vertices, will be denoted by $n = |V|$ and its *size*, or number of edges, by $m = |E|$. An (undirected) *edge* is a pair of vertices represented by a line segment. $G$ can be described by its *adjacency matrix* $A = (A_{uv})$ where $A_{uv} = 1$ if an edge joins vertices $u$ and $v$ and $A_{uv} = 0$ otherwise. A *loop* is an edge for which both end vertices coincide. A *multigraph* is a graph such that several edges have the same pair of end vertices (equivalently, two vertices $u$ and $v$ can be joined by several edges in a multigraph). A graph is *simple* if it has neither loops nor multiple edges. The *degree* of a vertex $u$, denoted by $k_u$, is the number of edges of which it is an end vertex (loops, if any, being counted twice). A *path* joining vertices $u$ and $v$ is an alternating sequence of vertices and edges such that the first vertex of the first edge is $u$, the second vertex of this edge coincides with the first vertex of the next edge and so forth until vertex $v$ is reached. A graph is *connected* if there is a path between any pair of its vertices. All graphs considered in this paper will be assumed to be connected. A *partition* $V_1, V_2, \ldots, V_M$ into $M$ classes is such that $V_i \neq \emptyset$ for all $i \in \{1, \ldots, M\}$, $V_i \cap V_j = \emptyset$ for all $i < j \in \{1, 2, \ldots, M\}$ and $V_1 \cup V_2 \cup \cdots \cup V_M = V$. The *subgraph* $G_i$ of $G$ induced by a vertex set $V_i \subseteq V$ is the graph having $V_i$ as vertex set and as edges those of $E$ having both end vertices in $V_i$.

We are interested in finding partitions of $V$ the classes of which (or equivalently the subgraphs of $G$ induced by these classes of vertices) correspond to communities, i.e., roughly speaking, they contain more edges joining vertices of the same community than vertices belonging to different communities. A precise definition of the quality of a partition into communities has been given in a seminal paper of Newman and Girvan [32]. It is equal to the sum over all communities of the observed number of edges within them minus the expected number of edges within them when placed at random, the distribution of degrees remaining the same. It is called *modularity* and denoted by $Q$:

$$Q = \sum_s (a_s - e_s)\,, \tag{1}$$

where $a_s$ is the fraction of edges in community $s$ and $e_s$ is the expected fraction of randomly distributed edges in that community. Let $V_s$ be the vertex set of community $s$ and $m_s$ the

number of edges in that community (assuming $G$ to be loopless):

$$m_s = \frac{1}{2} \sum_{u,v \in V_s} A_{uv} = \sum_{\substack{u,v \in V_s \\ u<v}} A_{uv} \tag{2}$$

where in the first relation we consider the full adjacency matrix and in the second one its upper triangular part. Then $a_s = \frac{m_s}{m}$.

Let $d_s$ denote the sum of degrees of vertices in $V_s$:

$$d_s = \sum_{u \in V_s} k_u.$$

Given an edge, the probability that its first end vertex belongs to $V_s$ is equal to $\frac{d_s}{2m}$. Assuming independence, the probability that its second vertex belongs to $V_s$ is the same. This can be illustrated by an urn model containing $k_1$ balls of colour 1, $k_2$ of color 2,... and $k_n$ of color $n$, that is $2m$ balls in all. A ball is drawn at random and it is checked if its color is the one assigned to the vertices of $V_s$. Then the ball is replaced, a second random draw is made and the same condition checked. Note that this urn model will be modified and used again below in the context of non independent draws, excluding loops.

The expected fraction of edges in community $s$ is thus:

$$e_s = \frac{d_s^2}{4m^2}.$$

Hence, substituting in (1), we have:

$$\begin{aligned} Q &= \sum_s \left( \frac{m_s}{m} - \frac{d_s^2}{4m^2} \right) \\ &= \sum_s \left( \frac{m_s}{m} - \frac{\left( \sum_{u \in V_s} k_u \right)^2}{4m^2} \right) \\ &= \sum_s \left( \frac{m_s}{m} - \frac{\left( \sum_{u \in V_s} \sum_{v \in V_s : v \neq u} k_u k_v + \sum_{u \in V_s} k_u^2 \right)}{4m^2} \right) \\ &= \sum_s \left( \frac{m_s}{m} - \frac{\sum_{u \in V_s} \sum_{v \in V_s : v \neq u} k_u k_v}{4m^2} \right) - \frac{\sum_{u \in V} k_u^2}{4m^2}. \end{aligned}$$

Introducing the Kronecker symbol $\delta(c_u, c_v)$, equal to 1 if vertices $u$ and $v$ belong to the same community $c_u = c_v$ and to 0 otherwise, and observing that $\delta(c_u, c_u) = 1 \quad \forall u \in V$, we get:

$$Q = \sum_{\substack{u,v \in V \\ u \neq v}} \left( \frac{A_{uv}}{2m} - \frac{k_u k_v}{4m^2} \right) \delta(c_u, c_v) - \frac{\sum_{u \in V} k_u^2}{4m^2} \tag{3}$$

or, using the upper triangular matrix and the main diagonal:

$$Q = \sum_{\substack{u,v\in V \\ v>u}} \left( \frac{A_{uv}}{m} - \frac{k_u k_v}{2m^2} \right) \delta(c_u, c_v) - \frac{\sum_{u\in V} k_u^2}{4m^2}. \quad (4)$$

This formula will require half the variables of the previous one for its maximization, as shown in the next section. As $G$ contains $m$ edges, the expected number $P_{uv}$ of edges between vertices $u$ and $v$ is equal to $k_u k_v/(2m)$ and the expected number $P_{uu}$ of loops at vertex $u$ is $k_u^2/(4m)$. One can thus write, as in [14]:

$$Q = \frac{1}{2m} \sum_{u,v\in V} (A_{uv} - \frac{k_u k_v}{2m}) \delta(c_u, c_v), \quad (5)$$

$$= \frac{1}{2m} \sum_{u,v\in V} (A_{uv} - P_{uv}) \delta(c_u, c_v) \quad (6)$$

in which terms in $u$ and $v$ are repeated for symmetry and the constant is not made explicit.

So the contribution to $Q$ of the loops is equal in absolute value to

$$C = \frac{\sum_{u\in V} k_u^2}{4m^2}. \quad (7)$$

Let us now evaluate the importance of this constant.

The Cauchy-Schwartz inequality leads to the following inequality on the sum of squares of degrees of $G$:

$$k_1^2 + \cdots + k_n^2 \geq \frac{1}{n}(k_1 + \cdots + k_n)^2 = \frac{4m^2}{n}.$$

Substituting in (7) gives:

$$C \geq \frac{1}{n}.$$

The de Caen's inequality [9] gives:

$$k_1^2 + \cdots + k_n^2 \leq m \left( \frac{2m}{n-1} + n - 2 \right) = \frac{2m^2}{n-1} + m(n-2).$$

Substituting again in (7) gives:

$$C \leq \frac{1}{2n-2} + \frac{n-2}{4m}.$$

As the graph $G$ is connected by assumption, we have $m \geq n-1$. Substituting for $m$:

$$C \leq \frac{1}{2n-2} + \frac{n-2}{4(n-1)} = \frac{n}{4n-4}.$$

We next show that these lower and upper bounds are both sharp. Let us consider a *regular* graph with all degrees equal to $r$ and $n$ vertices. We have $2m = nr$ and:

$$\sum_{u \in V} \frac{k_u^2}{4m^2} = \frac{nr^2}{n^2 r^2} = \frac{1}{n}.$$

Hence the lower bound on $C$ is attained by a a very large class of graphs, which includes sparse ones, such as cycles, and dense ones, such as complete graphs.

Let us now consider a *star* graph (or, in other words, a tree with a dominant vertex connected to all others). Then $m = n - 1$ and:

$$\sum_{u \in V} \frac{k_u^2}{4m^2} = \frac{(n-1)^2 + (n-1)}{4(n-1)^2} = \frac{n(n-1)}{4(n-1)^2} = \frac{n}{4n-4}.$$

Hence the upper bound on $C$ is sharp for stars.

**Remark 1.** When the order $n$ of the graph increases, the bounds tend to different limits:

$$\lim_{n \to \infty} \frac{1}{n} = 0 \quad \text{and} \quad \lim_{n \to \infty} \frac{n}{4n-4} = \frac{1}{4}.$$

So, due to loops, at least a small constant will be subtracted from the community dependent part of $Q$ when $n$ increases, but quite a large one must be subtracted in the worst case, even for large $n$.

**Remark 2.** The previous bounds can be used to evaluate the expected number of loops. It suffices to multiply $C$ by the number $m$ of edges to obtain

$$\frac{m}{n} \leq mC \leq \frac{mn}{4n-4}.$$

So the expected number of loops is at least half the average degree of $G$ and at most slightly more than one quarter of its size $m$.

### B. A modified null model which avoids loops

As shown in the previous subsection, the effect of loops on the value of $Q$ may be substantial even if the graph $G$ is loopless. We next propose a modification to the configuration model which avoids loops completely. This can be done easily with conditional probabilities. Consider the probability that an edge joins vertices $u$ and $v \neq u$, assuming the distribution of degrees is the same as for $G$ and there are no loops. To compute this probability, one can draw at random a first vertex, say $u$, exactly as before, i.e. with a probability $k_u/2m$.

However, once this is done, to select the second vertex probabilities should be conditional on the fact that $u$ has been drawn. Vertex $u$ cannot be drawn again, as this would induce a loop. Considering once again the urn model discussed above, one sees that after a first draw of a ball of color $u$ among the $2m$ balls, it is necessary to remove all remaining balls of the same color before making the second draw. From the classical definition of probability (i.e. the ratio of number of favourable cases to the total number of cases), instead of having $\frac{k_v}{2m}$ one will have $\frac{k_v}{2m-k_u}$. Moreover, as the event considered occurs both when $u$ is first drawn and $v$ after and when $v$ is first drawn and $u$ after, a similar calculation must be done in the second case.

Then, substituting in (3), the modified modularity $Q'$ is expressed as:

$$Q' = \sum_{u,v \in V} \left( \frac{A_{uv}}{2m} - \frac{k_u}{2m} \frac{k_v}{2m - k_u} \right) \delta(c_u, c_v). \tag{8}$$

Let $P'_{uv}$ denote the expected number of edges joining $u$ and $v$ in the new model. Then:

$$P'_{uv} = \frac{k_u}{2m} \frac{k_v}{2m - k_u} + \frac{k_v}{2m} \frac{k_u}{2m - k_v} \tag{9}$$

and, using an upper triangular matrix, $Q'$ is equal to:

$$Q' = \sum_{u,v \in V : v > u} \left( \frac{A_{uv}}{m} - P'_{uv} \right) \delta(c_u, c_v). \tag{10}$$

Substituting for $P'_{uv}$ gives the final expression:

$$Q' = \sum_{u,v \in V : v > u} \left( \frac{A_{uv}}{m} - \frac{k_u k_v}{2m} \left( \frac{1}{2m - k_u} + \frac{1}{2m - k_v} \right) \right) \delta(c_u, c_v). \tag{11}$$

### C. A modified null model which avoids multiple edges

We now consider the problem of avoiding multiple edges in the null model. This will be done by transferring probabilities from vertex pairs for which the expected number of edges is greater than 1 to the other vertex pairs. Increases in probabilities smaller than 1 (after the probability transfer) will be chosen to be proportional to the values of these probabilities. So the ratio of any two of these probabilities which have increased will remain the same. The key observation is that pairs of vertices will belong to three categories:

1. the set $I_1$ for which the expected number of edges $\frac{k_u k_v}{2m}$ is larger than 1; the excess probability for each of these edges will be $\frac{k_u k_v}{2m} - 1$. The excess probability for all edges will be $\sum_{u,v} \max(\frac{k_u k_v}{2m} - 1, 0)$;

2. the set $I_2$ for which the expected number of edges $\frac{k_u k_v}{2m}$ is smaller than 1, but would become greater than 1 after excess probability is redistributed proportionally;

3. the set $I_3$ for which the expected number of edges $\frac{k_u k_v}{2m}$ is smaller than 1 and would remain smaller than 1 after excess probability is redistributed proportionally.

We proceed in the following way in order to obtain modified probabilities. Excess probability will be redistributed among index pairs of category 2 in order to raise their probability up to 1 and among index pairs of category 3 proportionally to $\frac{k_u k_v}{2m}$. In other words, for all index pairs of category 3 the redistribution of probability corresponds to a change in scale.

Probability redistribution satisfies the following equation after ranking of index pairs by decreasing expected number of edges:

$$\sum_{(u,v)\in I_1} (\frac{k_u k_v}{2m} - 1) = \sum_{(u,v)\in I_2} (1 - \frac{k_u k_v}{2m}) + \delta \sum_{(u,v)\in I_3} \frac{k_u k_v}{2m}, \tag{12}$$

where $\delta$ is such that $\delta > 1 - \frac{k_u k_v}{2m} \quad \forall (u,v) \in I_2$ and $\delta \leq 1 - \frac{k_u k_v}{2m} \quad \forall (u,v) \in I_3$. The index pair set $I_2$ is not known a priori and may be empty. We first check if this is the case. This is done by computing $\delta$ and checking if the second inequality is satisfied. If so, $I_2$ is empty and the redistribution of expected number of edges satisfied the desired conditions. If not, $I_2$ is not empty and the first index pair follows immediately the last one of $I_1$. We then have to determine the last index pair of $I_2$. To check if the first index pair of $I_2$ is also the last one, we compute $\delta$ by using (12) and check if the two inequalities on $\delta$ are satisfied. If not, we increase the index by 1 and iterate. Note that, for large networks, a quicker procedure would be to use a dichotomous search for the last index pair in $I_2$, but this step is not very time consuming. Probabilities of pairs of vertices with indices in $I_2$ are then increased to 1 and probabilities of pairs of vertices with indices in $I_3$ multiplied by $1 + \delta$. The algorithm just described can be applied either to the initial model or to the model obtained after removing loops.

## III. COMPUTATIONAL EXPERIMENTS

In this Section, we recall a mathematical programming formulation due to Grötschel and Wakabayashi [19] for the clique partitioning problem. This model can be used to maximize modularity with any of the models described in the previous section. Binary variables

$x_{uv}$ are associated with all pairs of vertices $u$ and $v$ and equal to 1 if $u$ and $v$ are in the same community and 0 otherwise. Cliques correspond to communities. They satisfy the conditions of reflexivity, commutativity and transitivity. As we use the upper triangular matrix, commutativity is automatically satisfied. It is also the case for reflexivity as both end vertices of any loop always belong to the same community. The remaining constraints express transitivity, i.e. if vertices $u$ and $v$ are in the same community and $v$ and $w$ are in the same community, then also $u$ and $w$ are in the same community. The resulting problem is a linear 0-1 program with $n(n-1)/2$ variables and $n(n-1)(n-2)/2$ constraints:

$$\max_{u,v} \quad \sum_{u,v \in V : v >= u} Q_{uv} x_{uv} \tag{13}$$

$$s.t. \ \forall u < v < w \in V \quad x_{uv} + x_{vw} - x_{uw} \leq 1 \tag{14}$$

$$\forall u < v < w \in V \quad x_{uv} - x_{vw} + x_{uw} \leq 1 \tag{15}$$

$$\forall u < v < w \in V \quad -x_{uv} + x_{vw} + x_{uw} \leq 1. \tag{16}$$

This problem is NP-hard, even for the particular case of modularity maximization, as shown by Brandes et al. [5]. To accelerate the solution process, instead of considering the full model from the beginning, unsatisfied constraints may be added by batches until there are no more. When neither loops nor multiple edges are excluded, the coefficients $Q_{uv}$ in the objective function (13) are those of formula (4) and the constant $C$ of (7) must be subtracted from the objective. When loops are excluded from the null model but not multiple edges these coefficients are replaced by the coefficients of formula (11) (recall we denote this modified modularity by $Q'$). When multiple edges are excluded from the null model but not loops, these coefficients are obtained by applying the algorithm in subsection II C to the coefficients of (4) (in which case we denote the modified modularity by $Q''$). The constant $C$ must be subtracted from the objective function and the decrease in weights of the loops (if any) added. Finally, when loops and multiple edges are excluded from the null model, these coefficients are obtained by applying the same algorithm to the coefficients of (11) (in which case we denote the modified modularity by $Q'''$).

We implemented the Grötschel and Wakabayashi [19] algorithm using AMPL[16] and the "lazy constraints" feature of CPLEX [21], which automatically sets aside constraints which are strictly satisfied.

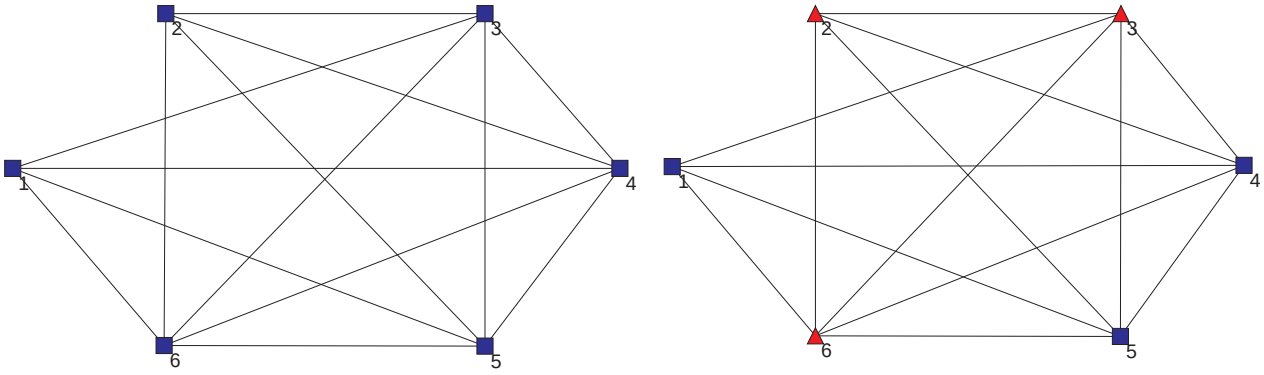


FIG. 1: (Color online) Partitions obtained on the artificial network with 6 vertices and 14 edges using the original modularity maximization model (left) and the modified model avoiding loops (right).

### A. Artificial examples

We first consider results for two small artificial instances. The complete graph with $n = 6$ vertices minus an edge given in Figure 1 is the first of them. In the standard model a single community $C_1 = \{1, 2, 3, 4, 5, 6\}$ is obtained with $Q = 0$. The constant $C$ is equal to 0.168367, close to the lower bound of the interval of feasible values $[0.166667, 0.300000]$, which would be attained if one more edge was added. When forbidding loops in the null model a partition into the two communities $C_1 = \{1, 4, 5\}$ and $C_2 = \{2, 3, 6\}$ is obtained, with a modularity $Q' = 0.0300207$. As there are no multiple edges in the null model, $Q'' = Q$ and the single community partition is obtained once more. When forbidding loops and multiple edges in the null model, once again two communities are obtained, i.e. $C_1 = \{1, 5, 6\}$ and $C_2 = \{2, 3, 4\}$, with a modularity $Q''' = 0.0254706$. Note that this partition is equivalent to the partition found for the case of excluded loops. Its value $Q'''$ differs from $Q'$ because removing loops entailed some expected numbers of edges becoming greater than 1 and the edge probabilities were then modified by the algorithm of subsection II C.

We next consider a graph with 20 vertices and 28 edges with a skewed distribution of degrees (see Figure 2). When using the standard model we obtain a partition into 4 communities: $C_1 = \{1, 2, 3, 4\}$, $C_2 = \{5, 6, 7, 8, 9\}$, $C_3 = \{10, 11, 12, 13, 14, 15, 16\}$, $C_4 = \{17, 18, 19, 20\}$, with a modularity value $Q = 0.475128$ and constant $C = 0.063776$ belonging to the interval $[0.050000, 0.263158]$. When forbidding loops in the null model a different

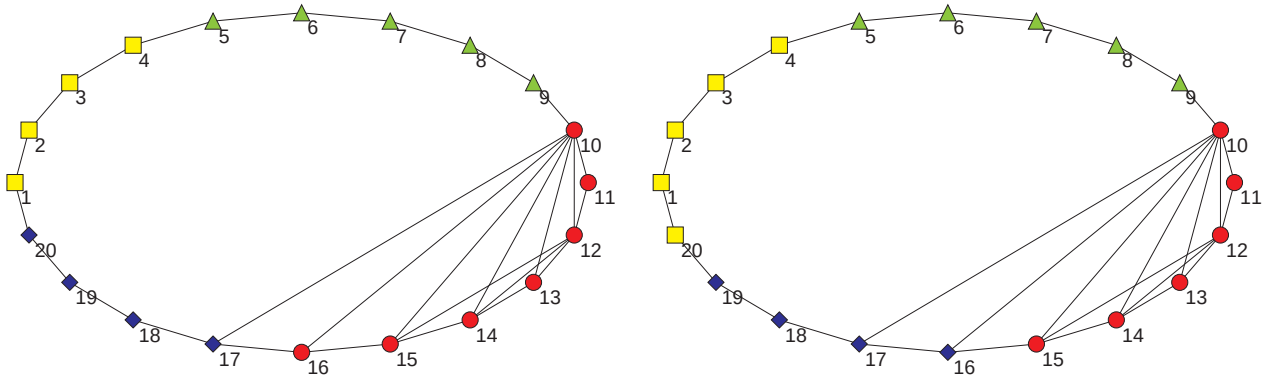


FIG. 2: (Color online) Partitions obtained on the artificial network with 20 vertices and 28 edges using the original modularity maximization model (left) and the modified model avoiding loops (right).

partition into four communities is obtained: $C_1 = \{1, 2, 3, 4, 20\}$, $C_2 = \{5, 6, 7, 8, 9\}$, $C_3 = \{10, 11, 12, 13, 14, 15\}$, $C_4 = \{16, 17, 18, 19\}$, with modularity $Q' = 0.518175$, so that vertex 20 is moved, in the new partition, from $C_4$ to the first community, and vertex 16 is moved from community $C_3$ to the fourth one. There are no multiple edges with or without loops. So, $Q'' = Q$ and $Q''' = Q'$.

### B. Examples from the literature

We now describe the results obtained on some datasets corresponding to various real world applications, which are often used to test algorithms and heuristics for community identification. We consider Zachary's karate club dataset [40], Lusseau's dolphins dataset [26], Hugo's *Les Misérables* dataset compiled by Knuth [22], Girvan and Newman dataset on American football games [17] and Krebs' political books dataset [23]. The first dataset describes friendship relations between 34 members of a karate club, studied by Zachary [40]. During the period of observation, a dispute between the club administrator and the karate instructor led to a split into two groups. The second dataset describes the communications among a group of 62 bottlenose dolphins of Doubtful Sound, New Zealand, studied by Lusseau [26], leading to a graph with 159 edges. Hugo's *Les Misérables* dataset describes the relationships between characters in Victor Hugo's masterpiece. Knuth [22] built a graph with 77 vertices associated to characters which interact and 257 edges associated with pairs

of characters appearing jointly in at least one chapter. The dataset on American football games [17] represents the schedule of games between American college football teams in the Fall 2000. The network is made of 115 vertices and 613 edges. Vertices represent 115 teams, most of which belong to one or another of 11 conferences, with intra-conference games more frequent than others. There are also 5 independent teams. Finally, the last dataset deals with co-purchasing of political books on `Amazon.com`.

Applying the exact algorithm by Grötschel and Wakabayashi [19] we found optimal partitions with values of modularity reported in Table I. This led to the following conclusions:

- the optimal partitions for the first four out of the five examples were the same for all models;
- when forbidding loops alone, the modularity always increased, sometimes substantially;
- when forbidding multiple edges alone, the modularity did not change much but decreased slightly or remained the same;
- when forbidding both loops and multiple edges, the modularity increased slightly over the value obtained when forbidding loops alone or remained the same.

| *dataset* | $n$ | $m$ | $\underline{C}$ | $C$ | $\overline{C}$ | $Q$ | $Q'$ | $Q''$ | $Q'''$ |
|---|---|---|---|---|---|---|---|---|---|
| `karate` | 34 | 78 | 0.029412 | 0.049803 | 0.257576 | 0.41979 | 0.45517 | 0.418578 | 0.455302 |
| `dolphin` | 62 | 159 | 0.016129 | 0.021399 | 0.254098 | 0.52852 | 0.54528 | 0.52852 | 0.54528 |
| `les miserables` | 77 | 254 | 0.012987 | 0.023731 | 0.253289 | 0.56001 | 0.57943 | 0.55976 | 0.578091 |
| `football` | 115 | 613 | 0.008696 | 0.008755 | 0.252193 | 0.60457 | 0.61249 | 0.60457 | 0.61249 |
| `political books` | 105 | 441 | 0.009524 | 0.013531 | 0.252404 | 0.52724 | 0.53587 | 0.52724 | 0.53587 |

TABLE I: Number of vertices and number of edges of real-world datasets, values of the constant $C$ defined in (7), together with its lower bound ($\underline{C}$) and upper bound ($\overline{C}$), and modularity values found applying the Grötschel and Wakabayashi algorithm on the original modularity maximization model ($Q$), the model forbidding loops ($Q'$), the model forbidding multiple edges ($Q''$) and the model forbidding loops and multiple edges ($Q'''$).

We discuss in more details the results obtained for Krebs' political books dataset [23], for which we find a different partition when forbidding loops in the null model. This dataset is a network with 105 vertices corresponding to titles of books and with 441 edges corresponding to co-purchases. Newman [31] provided a classification of these 105 books as liberal ($l$), conservative ($c$) or neutral ($n$). The optimal partition we found for the standard model consists in the 5 communities shown in Table II.

| $C_1$ | $C_2$ | $C_3$ | $C_4$ | $C_5$ |
|---|---|---|---|---|
| 1, 2, 3, 5<br>6, 7, 8, 19<br>29, 30 | 4, 9, 10, 11<br>12, 13, 14, 15<br>16, 17, 18, 20<br>21, 22, 23, 24<br>25, 26, 27, 28<br>33, 34, 35, 36<br>37, 38, 39, 40<br>41, 42, 43, 44<br>45, 46, 47, 48<br>54, 55, 56, 57 | 31, 32, 60, 61<br>62, 63, 64, 67<br>71, 72, 73, 74<br>75, 76, 77, 78<br>79, 80, 81, 82<br>83, 84, 85, 87<br>88, 89, 90, 91<br>92, 93, 94, 95<br>96, 97, 98, 99<br>100, 101, 102, 103 | 49, 50, 58 | 51, 52, 53, 59<br>65, 66, 68, 69<br>70, 86, 104, 105 |
| $c = 4$, $l = 0$, $n = 6$ | $c = 39$, $l = 0$, $n = 0$ | $c = 1$, $l = 38$, $n = 1$ | $c = 2$, $l = 0$, $n = 1$ | $c = 3$, $l = 5$, $n = 4$ |

TABLE II: Partition obtained with the standard modularity maximization model on Krebs' political book dataset

The partition obtained when forbidding loops consists again of 5 communities, which differ from the previous ones and are those shown in Table III.

So in both cases we found a partition containing two large communities with very few misclassifications and such that one of these communities, namely $C_3$, is the same for both models. We also found three smaller communities, again in both cases. Using the standard model we have two small communities with both $n$ and $c$ books and one community with all three categories. Using the first new model we found 3 communities more balanced in size, such that the first one and the last one differ from those obtained with the original model by one vertex only, and the second one contains only $c$ books. This new model splits

| $C'_1$ | $C'_2$ | $C'_3$ | $C'_4$ | $C'_5$ |
|---|---|---|---|---|
| 1, 2, 3, 5<br>6, 7, 8, 29<br>30 | 4, 9, 10, 12<br>13, 14, 15, 18<br>19, 21, 22, 23<br>24, 25, 26, 27<br>28, 33, 41, 42<br>43, 44, 45, 46<br>47, 48, 49, 50<br>51, 54, 55, 57<br>58 | 31, 32, 60, 61<br>62, 63, 64, 67<br>71, 72, 73, 74<br>75, 76, 77, 78<br>79, 80, 81, 82<br>83, 84, 85, 87<br>88, 89, 90, 91<br>92, 93, 94, 95<br>96, 97, 98, 99<br>100, 101, 102, 103 | 11, 16, 17, 20<br>34, 35, 36, 37<br>38, 39, 40, 56 | 52, 53, 59, 65<br>66, 68, 69, 70<br>86, 104, 105 |
| $c = 4$, $l = 0$, $n = 5$ | $c = 31$, $l = 0$, $n = 2$ | $c = 1$, $l = 38$, $n = 1$ | $c = 12$, $l = 0$, $n = 0$ | $c = 2$, $l = 5$, $n = 4$ |

TABLE III: Partition obtained with the modularity maximization model forbidding loops on Krebs' political book dataset

the original community $C_2$ into the two communities $C'_2$ and $C'_4$, while the original $C_4$ is included in the new community $C'_2$.

In [6] we introduced the following criterion to count misclassifications: any $l$ in a community with a majority of $c$'s or $n$'s or conversely counts for 1; any $n$ in a community with a majority of $c$'s or a majority of $l$'s or conversely counts for 1/2 misclassification. Using this criterion, we have that the total number of misclassifications for the original model is 9, while for the model avoiding loops it amounts to 8.5. Figures 3 and 4 show the partitions obtained using the standard model and the model forbidding loops. As there are no multiple edges, either in the standard model or in the model forbidding loops, optimal partitions for the last two models coincide with those of the first and the second model respectively.

### C. Examples from Lancichinetti et al.'s benchmark

As final examples, we describe results obtained for five networks with the same format as those of the Lancichinetti et al.'s [25] benchmark. These networks are caractherized

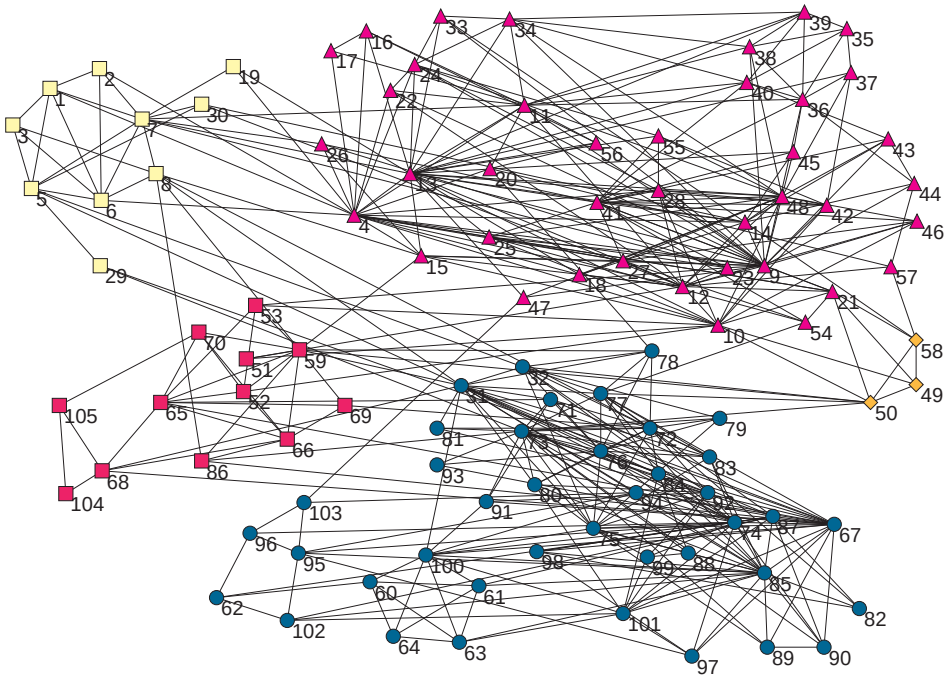

FIG. 3: (Color online) Partition obtained using the standard modularity maximization model for Krebs' political books dataset.

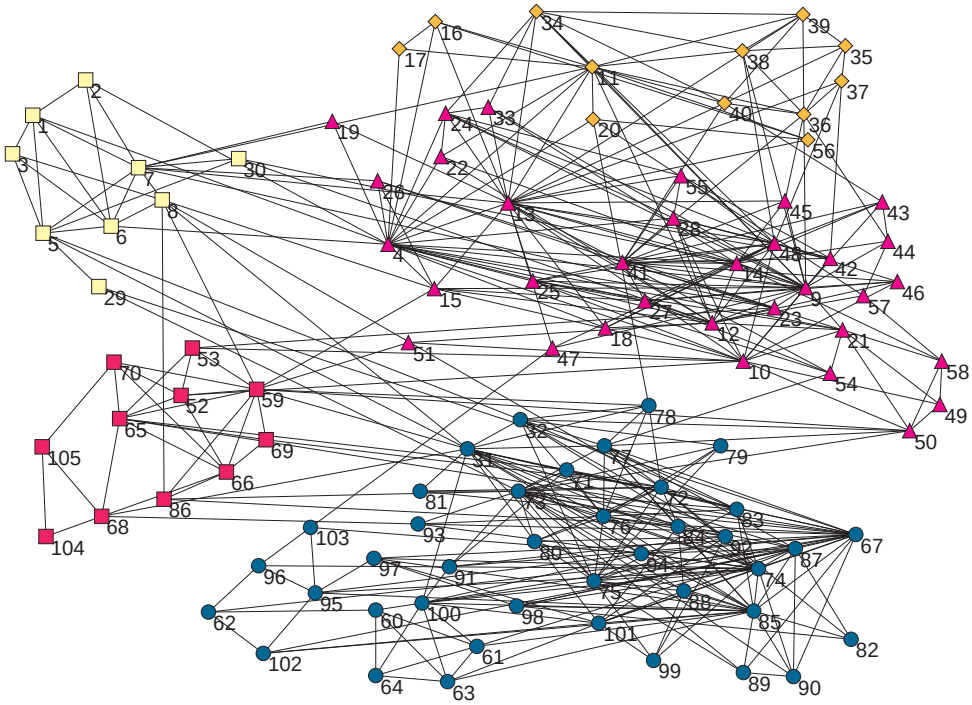

FIG. 4: (Color online) Partition obtained using the modularity maximization model forbidding loops for Krebs' political books dataset.

by a distribution of degrees and a distribution of size communities that follow power laws. Furthermore, the ratio of outer edges to inner edges is controlled by a parameter $\mu$. We used the code by Fortunato et al. [13] in order to generate these networks, considering a value of degree distribution $\gamma = 2$, a value of size community distribution $\beta = 2$ and $\mu = 0.3$. The size of the networks is increasing up to $n = 256$, $m = 8192$.

The optimal partitions for all these networks were the same for all models. In Table IV we show the values of modularity obtained on the considered networks using the original modularity maximization model and its proposed variants . The values of the constant $C$ defined in (7), together with its lower bound and upper bound, are also reported. The results show that when forbidding loops alone, the modularity always increased, as already observed for examples from the literature. There are no multiple edges in the considered examples, hence the values of modularity for the model forbidding multiple edges alone are the same values obtained with the original model and the values of modularity for the model forbidding loops and multiple edges are the same values obtained with the model forbidding loops only. For four networks the value of the constant $C$ is on its lower bound.

| $n$ | $m$ | $\underline{C}$ | $C$ | $\overline{C}$ | $Q$ | $Q'$ | $Q''$ | $Q'''$ |
|---|---|---|---|---|---|---|---|---|
| 32 | 128 | 0.031250 | 0.031250 | 0.258065 | 0.453125 | 0.477319 | 0.453125 | 0.477319 |
| 64 | 517 | 0.015625 | 0.017760 | 0.253968 | 0.389303 | 0.401042 | 0.389303 | 0.401042 |
| 100 | 1250 | 0.010000 | 0.010000 | 0.252525 | 0.4508 | 0.458376 | 0.4508 | 0.458376 |
| 128 | 2048 | 0.007812 | 0.007812 | 0.251969 | 0.450684 | 0.456589 | 0.450684 | 0.456589 |
| 256 | 8192 | 0.003906 | 0.003906 | 0.250980 | 0.449829 | 0.45277 | 0.449829 | 0.45277 |

TABLE IV: Number of vertices and number of edges of datasets from Lancichinetti et a.'s benchmark, values of the constant $C$ defined in (7), together with its lower bound ($\underline{C}$) and upper bound ($\overline{C}$), and modularity values found applying the Grötschel and Wakabayashi algorithm on the original modularity maximization model ($Q$), the model forbidding loops ($Q'$), the model forbidding multiple edges ($Q''$) and the model forbidding loops and multiple edges ($Q'''$).

## IV. CONCLUSIONS

In the standard modularity maximization model of Newman and Girvan the null model is associated with a graph containing loops and possibly multiple edges, while the graph under study usually has neither. We have given sharp bounds on the expected number of loops and on their impact on the modularity value. While the absolute value of the lower bound on this last quantity is only $1/n$, which is attained for regular graphs, and tends to be small, the absolute value of the upper bound is $n/(4n-4)$, which is attained for stars, and is thus large. So, the effect of loops in the null model can be considerable in the worst case.

Using conditional probabilities, we have provided a modified formula for modularity in the case where loops are excluded from the null model. We have also given an algorithm for redistribution of the excess over 1 of the expected number of edges between two vertices to the other edges for which it is not the case. This redistribution is proportional to the edge probabilities. The algorithm can be applied either to the initial null model or to the modified null model in which loops have been eliminated.

The effect of these modifications are studied on a couple of small artificial graphs as well as on five graphs from the literature. For four of the latter, the optimal partition remains the same but the modularity value is increased. For the fifth one, both modularity value and the corresponding partition are different from those of the standard model. Similar results were obtained for five increasingly large networks of Lancichinetti et al. [25] format.

The theoretical results of this paper show that the influence of loops on the value of modularity is substantial in worst case. Experimental results show that the partitions obtained with the standard modularity function and with the modified modularity function in which loops are excluded are often the same. However, values of the latter function are larger than those of the former.

When loops and multiple edges do not appear in the model under study, the proposed variants of modularity can be useful, in two ways: if the data set is small enough for exact optimization in reasonable computing time, that can be done. Otherwise, as any heuristic for standard modularity maximization can be viewed as one for the modified modularity, any performing heuristic can be used to find a good partition, the value of which will then

be computed from the main formula of this paper.

---


[1] G. Agarwal and D. Kempe. Modularity-maximizing graph communities via mathematical programming. *The European Physical Journal B*, 66(3):409–418, 2008.

[2] A. Arenas, A. Fernandez, and S. Gomez. Analysis of the structure of complex networks at different resolution levels. *New Journal of Physics*, 10:053039, 2008.

[3] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal Statistical Mechanics*, (P10008), 2008.

[4] S. Boccaletti, M. Ivanchenko, V. Latora, A. Pluchino, and A. Rapisarda. Detecting complex network modularity by dynamical clustering. *Physical Review E*, 75:045102, 2007.

[5] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner. On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering*, 20(2):172–188, 2008.

[6] S. Cafieri, P. Hansen, and L. Liberti. Edge ratio and community structure in networks. *Physical Review E*, to appear.

[7] D. Chen, Y. Fu, and M. Shang. A fast and efficient heuristic algorithm for detecting community structures in complex networks. *Physica A*, 388(13):2741–2749, 2009.

[8] A. Clauset, M.E.J. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E*, 70:066111, 2004.

[9] D. de Caen. An upper bound on the sum of squares of degrees in a graph. *Discrete Mathematics*, 185:245–248, 1998.

[10] H.N. Djidjev. A scalable multilevel algorithm for graph clustering and community structure detection. *Lecture Notes in Computer Science*, 4936, 2008.

[11] J. Duch and A. Arenas. Community identification using extremal optimization. *Physical Review E*, 72 -027104(2):027104, 2005.

[12] Y. Fan, M. Li, P. Zhang, J. Wu, and Z. Di. Accuracy and precision of methods for community identification in weighted networks. *Physica A*, 377(1):363–372, 2007.

[13] S. Fortunato. `http://santo.fortunato.googlepages.com/`.

[14] S. Fortunato. Community detection in graphs. *Physics Reports*, 486:75–174, 2010.

[15] S. Fortunato and M. Barthelemy. Resolution limit in community detection. *Proceedings of*

*the National Academy of Sciences, USA*, 104(1):36–41, 2007.

[16] R. Fourer and D. Gay. *The AMPL Book*. Duxbury Press, Pacific Grove, 2002.

[17] M. Girvan and M. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences, USA*, 99(12):7821–7826, 2002.

[18] Benjamin H. Good, Yves-Alexandre de Montjoye, and Clauset Aaron. The performance of modularity maximization in practical contexts. Technical Report 0910.0165, arXiv, 2009.

[19] M. Grötschel and Y. Wakabayashi. A cutting plane algorithm for a clustering problem. *Mathematical Programming*, 45:59–96, 1989.

[20] R. Guimerà and A.N. Amaral. Functional cartography of complex metabolic networks. *Nature*, 433:895–900, 2005.

[21] ILOG. *ILOG CPLEX 11.0 User's Manual*. ILOG S.A., Gentilly, France, 2008.

[22] D.E. Knuth. *The Stanford GraphBase: A Platform for Combinatorial Computing*. Addison-Wesley, Reading, MA, 1993.

[23] V. Krebs. `http://www.orgnet.com/` (unpublished).

[24] J.M. Kumpula, J. Saramaki, K. Kaski, and J. Kertesz. Limited resolution and multiresolution methods in complex network community detection. *Fluctuation and Noise Letters*, 7(3):L209–L214, 2007.

[25] A. Lancichinetti, S. Fortunato, and F. Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, page 046110, 2008.

[26] D. Lusseau, K. Schneider, O.J. Boisseau, P. Haase, E. Slooten, and S.M. Dawson. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. can geographic isolation explain this unique trait? *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003.

[27] S. Maslova, K. Sneppenb, and A. Zaliznyak. Detection of topological patterns in complex networks: correlation profile of the internet. *Physica A*, 333:529–540, 2004.

[28] C.P. Massen and J.P.K. Doye. Identifying communities within energy landscapes. *Physical Review E*, 71:046101, 2005.

[29] J. Mei, S. He, G. Shi, Z. Wang, and W. Li. Revealing network communities through modularity maximization by a contraction-dilation method. *New Journal of Physics*, 11:043025, 2009.

[30] R. Milo, N. Kashtan, S. Itzkovitz, M. E. J. Newman, and U. Alon. On the uniform generation of random graphs with prescribed degree sequences. Technical Report 0312028, arXiv, 2004.

[31] M. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences, USA*, pages 8577–8582, 2006.

[32] M. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69:026133, 2004.

[33] M.E.J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69:066133, 2004.

[34] Y.Q. Niu, B.Q. Hu, W. Zhang, and M. Wang. Detecting the community structure in complex networks based on quantum mechanics. *Physica A*, 387(24):6215–6224, 2008.

[35] J. Ruan and W. Zhang. Identifying network communities with a high resolution. *Physical Review E*, 77:016104, 2008.

[36] P. Schuetz and A. Caflisch. Efficient modularity optimization by multistep greedy algorithm and vertex mover refinement. *Physical Review E*, 77:046112, 2008.

[37] Y. Sun, B. Danila, K. Josic, and K. E. Bassler. Improved community structure detection using a modified fine-tuning strategy. *Europhysics Letters*, 86:28004, 2009.

[38] M. Tasgin, A. Herdagdelen, and H. Bingol. Community detection in complex networks using genetic algorithms. *arXiv:0711.0491*, 2007.

[39] G. Xu, S. Tsoka, and L.G. Papageorgiou. Finding community structures in complex networks using mixed integer optimization. *Eur. Physical Journal B*, 60:231–239, 2007.

[40] W.W. Zachary. An information flow model for conflict and fission in small group. *Journal of Anthropological Research*, 33:452–473, 1977.