Edge ratio and community structure in networks

Sonia Cafieri,* Pierre Hansen,† and Leo Liberti‡
LIX, École Polytechnique, F-91128 Palaiseau, France

(Received 28 August 2009; revised manuscript received 12 December 2009; published 17 February 2010)

A hierarchical divisive algorithm is proposed for identifying communities in complex networks. To that effect, the definition of *community in the weak sense* of Radicchi *et al.* [Proc. Natl. Acad. Sci. U.S.A. **101**, 2658 (2004)] is extended into a criterion for a bipartition to be optimal: one seeks to maximize the minimum for both classes of the bipartition of the ratio of inner edges to cut edges. A mathematical program is used within a dichotomous search to do this in an optimal way for each bipartition. This includes an exact solution of the problem of detecting *indivisible communities*. The resulting hierarchical divisive algorithm is compared with exact modularity maximization on both artificial and real world data sets. For two problems of the former kind optimal solutions are found; for five problems of the latter kind the edge ratio algorithm always appears to be competitive. Moreover, it provides additional information in several cases, notably through the use of the dendrogram summarizing the resolution. Finally, both algorithms are compared on reduced versions of the data sets of Girvan and Newman [Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002)] and of Lancichinetti *et al.* [Phys. Rev. E **78**, 046110 (2008)]. Results for these instances appear to be comparable.

DOI: 10.1103/PhysRevE.81.026105 PACS number(s): 89.75.Hc, 87.23.Ge, 89.20.Hh

I. INTRODUCTION

Networks, or graphs, are a basic and versatile tool for the study of complex systems in a variety of settings. This includes modeling of telecommunication networks, such as the World Wide Web [1], transportation networks [2], such as rail or road networks or electricity grids, social networks [3], such as board structures and situations of cooperation or conflict, citation and coauthorship networks [4], biological networks, such as food webs [5], and many more. Networks are composed of a set of vertices and a set of edges joining pairs of vertices. Vertices are associated with the entities of the system under study (people, companies, towns, natural species, etc.). Edges express that a relation defined on all pairs of vertices holds or not for each such pair. Often networks are weighted, i.e., a number is associated to each edge which expresses the strength of the corresponding relation. Networks have long been studied for their mathematical properties and as a tool for modeling and optimization [6–8]. In the past decade, extensive studies of complex networks have been made by the physicists' community. This led to several important discoveries, such as the power law distribution of degrees [9] and the small world property [10].

A topic of particular interest in the study of complex networks is the identification of *communities*, also called *modules* or sometimes *clusters*. Fortunato [11] recently made an extensive and thorough survey of that very active research domain. Speaking informally, a community is a subset of vertices such that there are more edges within the community than edges joining it to the outside (or, in other words, belonging to the cut separating that community from all others). Communities are akin to clusters which have been studied for a long time in data analysis and, more recently, in data mining. However, particular properties of networks lead to specialized heuristics or algorithms, many of which can identify communities in very large networks. To perform such a task, it is necessary to make precise the definition of a community. Newman and Girvan [12] proposed to compare the number of edges within a community to the expected number of edges within that community when they have been chosen at random with the same distribution of degrees. This definition was extended by formulating the concept of *modularity* for a partition of a network as the sum for all communities of the difference between the fraction of edges they contain and the expected fraction of edges under the *configuration model* [13,14]. Such a criterion can be used to evaluate partitions and its maximization leads to an optimal partition in a precise sense. Moreover, this optimal partition should itself have an optimal number of communities. A large number of heuristics were proposed to maximize modularity. They rely on simulated annealing [15], extremal optimization [16], mean field annealing [17], genetic search [18], dynamical clustering [19], multilevel partitioning [20], contraction dilation [21], multistep greedy [22], quantum mechanics [23], label propagation [24,25], and a variety of other approaches [26–31].

These heuristics provide, usually in moderate time, near optimal partitions for the modularity criterion or, possibly, optimal partitions but without the proof of their optimality. Brandes *et al.* [32] proved that modularity maximization is NP-hard. Recently, Xu *et al.* [33] proposed a mathematical programming model to maximize modularity exactly, and, using the CPLEX software [34], they were able to find optimal partitions for data sets with up to 104 vertices. While such a number of vertices is clearly moderate, problems of these sizes may be of interest in their own right. Moreover, such research may pave the way toward more efficient exact methods. Many data sets have however much more than 100 entities and can only be solved approximately by some heuristic. Clustering heuristics and algorithms can be divided, as traditional in cluster analysis [35–37], into partitioning algo-

*cafieri@lix.polytechnique.fr

†Also at GERAD, HEC Montréal, Canada. pierre.hansen@gerad.ca

‡liberti@lix.polytechnique.fr

rithms which aim at finding the best partition into a given number of clusters and hierarchical algorithms which lead to a set of nested partitions, i.e., partitions such that any two clusters in any of them are either disjoint or included one into the other. Hierarchical clustering schemes can be further divided into agglomerative and divisive ones. In agglomerative hierarchical clustering schemes one begins with a partition into as many clusters as entities, each containing a single entity, then one iteratively merges the two clusters such that the objective function increases the most in case of maximization (or decreases the most in case of minimization). In divisive hierarchical clustering schemes one begins with a single cluster containing all entities, which is then bipartitioned in such a way that the objective function increases most (or decreases most). While merging at each iteration in agglomerative algorithms is done in an optimal way, there is no guarantee that the partition obtained remains optimal after several iterations (there are a few exceptions as, e.g., the single linkage algorithm, which maximizes the *split* of partitions obtained at all levels [38]). In divisive hierarchical clustering algorithms the bipartitioning problem to be solved at each iteration is often NP-hard and requires a specific algorithm or heuristic. Again there is no guarantee that the partition obtained after several iterations will be optimal.

A very efficient agglomerative hierarchical clustering scheme was proposed by Clauset *et al.* in 2004 [39]. It exploits the fact that merging clusters is only profitable if there is at least one edge between them. For sparse networks this gives a heuristic with very low complexity, i.e., $O(n \ln^2 n)$, where n is the number of vertices. This contrasts with standard agglomerative hierarchical clustering schemes (e.g., single average, complete linkage, etc.) which require $O(n^2)$ time [40]. Several divisive algorithms were derived even before the definition of modularity was proposed [41,42]. They solve the bipartition problem arising at each iteration by removing edges of the network which appear to be likely to join different communities. One may then select iteratively edges with the largest *betweenness*, i.e., which belong to the largest number of shortest paths between pairs of vertices of the network. If removing edges increases the number of connecting components, a new partition has been obtained. Alternatively, one can use the clustering coefficient, i.e., the ratio of the number of triangles including an edge to the largest possible number of such triangles. Edges with small clustering coefficient are good candidates for removal. This approach can be extended by considering small cycles larger than triangles. A spectral method for divisive clustering with the modularity criterion was developed by Newman in 2006 [43]. Signs of the components of the first eigenvector of the so-called modularity matrix give a first approximate bipartitioning, which can be improved upon by some further heuristic such as the Kernighan-Lin method [44].

Clearly, maximizing modularity is the mainstream in community identification since about five years. However, several authors have criticized this concept, usually showing that counterintuitive results can be obtained for artificial constructed instances [32,45]. Moreover, it was shown [45] that using the modularity criterion has some limit of resolution. This means that in the presence of large communities, small communities may be undetectable even if they are very dense. Two such examples will be discussed later. To palliate this problem several modifications to the modularity function were proposed [31,46] and heuristics generalized accordingly.

An alternative approach to modularity maximization for finding communities is based on the satisfaction of reasonable *a priori* conditions to have a community. Radicchi *et al.* [42] proposed two such conditions defining communities in a strong and a weak sense. Recall that the degree k_i of a vertex i belonging to V is the number of its neighbors (or adjacent vertices). Let $S \subseteq V$ be a subset of vertices. Then the degree k_i can be separated into two components $k_i^{in}(S)$ and $k_i^{out}(S)$, i.e., the number of neighbors of i inside S and the number of neighbors of i outside S. A set of vertices S forms a community in the *strong sense* if and only if every one of its vertices has more neighbors within the community than outside,

$$k_i^{in}(S) > k_i^{out}(S), \quad \forall \quad i \in S.$$

Such a condition is hard to satisfy by a community and even more so by all communities of a partition. Therefore, it does not appear to be much used in practice. A set of vertices S forms a community in the *weak sense* if and only if the sum of all degrees within S is larger than the sum of all degrees joining S to the rest of the network,

$$\sum_{i \in S} k_i^{in}(S) > \sum_{i \in S} k_i^{out}(S).$$

This is equivalent to the condition that the number of edges within S is at least half the number of edges in the cut of S. From now on, we refer to this inequality as the *weak condition*. Note that it may be of interest to consider a similar definition but with a nonstrict inequality. Indeed, mathematical programming handles more easily nonstrict inequalities than strict ones. Moreover, as will be shown below, it may also be of interest to consider alternative optimal solutions for which the condition is satisfied as an equality. Divisive hierarchical algorithms work by successive bipartitions. It appears to be desirable that the weak condition be satisfied by *both* communities obtained when a bipartition takes place. Clearly, this is not always possible. This led Radicchi *et al.* [42] to propose such a condition as a local stopping criterion in a divisive hierarchical clustering algorithm. Wang *et al.* [47] called a community S *indivisible* if there is no bipartition, (S_1, S_2) of S, such that both S_1 and S_2 satisfy the weak condition. These authors give a mathematical programming formulation of the problem of determining whether a community is divisible or indivisible. Unfortunately, this formulation is a mixed 0-1 quadratic program with a nonconvex continuous relaxation, and consequently it is very difficult to solve.

In this paper, we give another, much simpler, program to detect indivisibility. We then observe that the weak condition is often satisfied by a very large number of bipartitions. To choose among them we consider the ratio of the number of edges within a community to the number of cut edges which have one end point only within that community; i.e., denoting this ratio by $r(S)$, we have

$$r(S) = \sum_{i \in S} k_i^{in}(S) \Bigg/ \sum_{i \in S} k_i^{out}(S).$$

When dividing S we consider this ratio for both communities S_1 and S_2 and maximize the smallest value; i.e., we address the problem

$$\max_{S_1,S_2 \subset V} \min(r(S_1), r(S_2)),$$

where $S_1 \cup S_2 = S$, $S_1 \cap S_2 = \emptyset$, and $S_1, S_2 \neq \emptyset$.

Solving this problem by a sequence of linear programs in 0-1 variables within a dichotomous search yields a divisive clustering algorithm, with a clear and well defined criterion. Moreover, it is *locally optimal* in the sense that each division is done in an optimal way.

The paper is organized as follows. In Sec. II some notation is given and conditions for a community to be divisible are presented. These conditions are used in an algorithm to maximize the edge ratio of a given community. Moreover, it is explained how this can be done for the communities obtained after several iterations. Computational results are presented in Sec. III, first on two artificial data sets and then on five well-known real world ones and on data sets from benchmarks. Results are compared to those obtained by maximizing modularity. Section IV presents conclusions and a few topics for future research.

II. MAXIMIZING THE EDGE RATIO

A. Indivisible communities

The first problem we address is to find whether a given network can be divided into two or more communities which all satisfy the weak condition. Note that if a network can be partitioned into more than two communities it can also be partitioned into two communities. Indeed, merging two communities can never decrease the number of inner edges nor increase the number of cut edges. Let $G=(V,E)$ denote the network under study, with vertex set V and edge set E. Then G is indivisible if and only if there is no bipartition (V_1, V_2) of V such that each class, V_1 and V_2, contains at least as many inner edges as one half the number of cut edges, i.e., edges joining vertices from one community to the other. The factor of one half implies that when both V_1 and V_2 satisfy the weak condition, the total number of inner edges is larger than or equal to the number of cut edges.

Both V_1 and V_2 must be nonempty and disjoint and their union must be equal to V. Binary variables x_i will be used to denote to which set V_1 or V_2 belongs vertex v_i for all $i \in V$. By convention, we assume $x_i=1$ if i belongs to V_1 and $x_i=0$ otherwise. We next introduce two sets of binary variables t_{ij} and s_{ij} associated to the edges (i,j) of E. Edge (i,j) will belong to the community induced by V_1 if $t_{ij}=1$ and $s_{ij}=0$ and to the community induced by V_2 if $t_{ij}=s_{ij}=0$ and will join vertices belonging to both communities if $t_{ij}=0$ and $s_{ij}=1$. All these conditions are imposed by the following constraints associated with each of the edges:

$$2t_{ij} + s_{ij} = x_i + x_j, \quad \forall \quad i,j \in E. \tag{1}$$

Indeed, if $x_i=x_j=1$, then $x_i+x_j=2$, which imposes $t_{ij}=1$ and $s_{ij}=0$; if $x_i=1$, $x_j=0$ or $x_i=0$, $x_j=1$, their sum is equal to 1, which imposes $t_{ij}=0$ and $s_{ij}=1$; finally, if $x_i=x_j=0$, their sum is equal to 0, which imposes $t_{ij}=0$ and $s_{ij}=0$.

We next express the weak condition. For the first community it amounts to

$$2\sum_{i,j \in E} t_{ij} \geq \sum_{i,j \in E} s_{ij}. \tag{2}$$

To find a similar expression for the second community, we note that its number of edges is equal to $|E| - \Sigma_{i,j \in E} t_{ij} - \Sigma_{i,j \in E} s_{ij}$. We can then write the condition as

$$2\sum_{i,j \in E} t_{ij} + 3\sum_{i,j \in E} s_{ij} \leq 2|E|. \tag{3}$$

In order for both communities to be nonempty, we need to add a further condition: at least one edge joins a vertex of one community to a vertex of the other,

$$\sum_{i,j \in E} s_{ij} \geq 1. \tag{4}$$

Moreover, all variable must be binary,

$$x_i, t_{ij}, s_{ij} \in \{0,1\}, \quad \forall \quad i,j \in E. \tag{5}$$

Observe that this mathematical expression of the weak condition does not imply any optimization and hence does not require an objective function. One could easily decide upon a reasonable one which would be used as a secondary criterion. For instance, one might wish to minimize the number of cut edges (which corresponds to min $\Sigma_{i,j \in E} s_{ij}$). Computational experiments show however that adding such an objective function may increase very substantially the resolution time of this mathematical program.

B. Finding two communities with largest edge ratio

The definition of a community in the weak sense given by Radicchi *et al.* [42] can often be satisfied by a very large number of communities, and it may be difficult to choose among them. This does not matter if one considers only those communities obtained with divisive hierarchical clustering schemes, such as those of Girvan and Newman [41] or of Radicchi *et al.* [42]. Indeed, in such cases, the identification of communities is done through exploiting betweenness of edges or clustering coefficients in order to choose edges to be removed one at a time until the network becomes disconnected. Following the proposal of Wang *et al.* [47], the weak community definition would then only be used as a stopping criterion. It would answer the indivisibility problem as a yes or no question.

The situation is different if one wishes to build a divisive hierarchical clustering scheme using only the weak condition or a variant thereof. One may then wonder if it is possible to strengthen this definition by quantifying how much the number of inner edges is larger than the number of cut edges. This is easily done by introducing a parameter α in the weak condition which then becomes equal to

$$\sum_{i \in S} k_i^{in}(S) \geq \alpha \sum_{i \in S} k_i^{out}(S). \tag{6}$$

So, in case of equality, the coefficient α is equal to the ratio of twice the number of edges within the community S divided by the number of edges within the cut of that community. We call it *edge ratio* for short. One can then seek the maximum value of α for which the network will be divisible. For this value α will be equal to twice the ratio of the number of edges within S divided by the number of edges within the cut of S.

Doing this, we obtain a more coherent divisive hierarchical clustering scheme than we would obtain following the proposal of Wang *et al.* [47] discussed above because the communities found will be selected using only the (extended) weak condition. Returning to the formulation of this condition given in Sec. II A, we observe that inequalities (2) and (3) become

$$2 \sum_{i,j \in E} t_{ij} \geq \alpha \sum_{i,j \in E} s_{ij} \tag{7}$$

and

$$2 \sum_{i,j \in E} t_{ij} + (2+\alpha) \sum_{i,j \in E} s_{ij} \leq 2|E|. \tag{8}$$

Then maximizing α subject to these last constraints as well as constraints (1), (4), and (5) gives us a mathematical programming formulation for identification of optimal communities according to the edge ratio criterion. This program has a linear objective function but, due to α, nonlinear and nonconvex constraints. As in the previous case, all the variables except α take the values 0 or 1. Moreover, if α is fixed, a linear program in 0-1 variables is obtained. Despite being NP hard, such programs may be solved efficiently in practice by a state-of-the-art software such as CPLEX [34]. This suggests to solve the optimal bipartition problem with a dichotomous search on the values of α. An initial value α equal to 1 can first be chosen. If there is no feasible solution for that value, the network is indivisible. Otherwise, the value of α may be doubled and feasibility checked until a value is attained for which the weak condition cannot be satisfied, i.e., the program is no more feasible. This gives an upper bound $\bar{\alpha}$ and the previous value of α gives a lower bound $\underline{\alpha}$. Then the dichotomous search proceeds by considering the mid value of the interval $[\underline{\alpha}, \bar{\alpha}]$. If the program is feasible for this value of α, the procedure is iterated on the upper half of the current interval, if not it is iterated on the lower half. The procedure stops when the length $\bar{\alpha} - \underline{\alpha}$ of the current interval is smaller than some given tolerance ϵ.

We note that an alternative approach can be based on the solution of a mixed-integer linear programming problem obtained considering α as a (continuous) variable and linearizing the products of α and the binary variables in constraints (7) and (8). However, this will lead to the introduction of many more variables and constraints. Also, in order to apply the linearization one needs a lower and an upper bound on α explicitly known. Thus, our approach, which dynamically computes bounds on α, appears to be more convenient.

This basic procedure can be accelerated in several ways. First, one can use an initial value of α corresponding to a solution obtained by some heuristic instead of the value $\alpha = 1$. Second, each time a feasible solution is obtained, one can check what is the corresponding maximum value for α, i.e., the minimum of the edge ratios for the two communities obtained. If this value is larger than the current value of α it may be taken as the lower bound of the next interval of values of α. Third, once the best value of α for the current solution is found, one may test whether the solution obtained for $\alpha + \epsilon$ is feasible or not. If not, the optimal solution (up to a tolerance ϵ) has been found. Fourth, symmetry of the solution set can be removed by fixing a variable, say x_1, at 1 from the outset.

Another possibility is to use an *alternating* algorithm, which explores increasing values of α by alternatively finding a feasible solution and the corresponding largest value for α. More precisely, it begins by considering the known feasible solution with the largest value of α. Then, it increases α by ϵ and attempts to solve the corresponding 0-1 program. If a feasible solution is found, the value of α for that solution is computed; i.e., α is set to the minimum of the edge ratios for both communities found and the procedure is iterated. If not, an optimal solution (up to a tolerance of ϵ) has been found.

Computational experiences show that there is no systematic dominance of the dichotomous search over the alternating algorithm or conversely.

C. Divisive algorithm

Once a partition into two communities has been found in the given network, one may wish to find further bipartitions of one or both of these or show that they are indivisible. In doing this, one must take into account not only the edges within each of these communities but also those of the cut between them. To this effect, one will introduce weights w_i associated to each vertex and equal to the number of cut edges between that vertex and those of the other community (or after several cuts have taken place of all other communities). Again, inequalities (2) and (3) are modified and become

$$2 \sum_{i,j \in E} t_{ij} \geq \alpha \left(\sum_{i,j \in E} s_{ij} + \sum_{i \in V} w_i x_i \right) \tag{9}$$

and

$$2 \sum_{i,j \in E} t_{ij} + (2+\alpha) \sum_{i,j \in E} s_{ij} + \alpha \sum_{i \in V} w_i (1 - x_i) \leq 2|E|. \tag{10}$$

All tools for building a divisive hierarchical clustering scheme based on the edge ratio criterion are now available. It proceeds by first finding the two communities with largest edge ratio in the given network using the algorithm described in Secs. II A and II B. Then the corresponding subproblems are updated by computing the weights of the vertices and stored together with the corresponding value of α. Iteratively, as long as some subproblems remain stored, one of them is selected (the order does not matter) and the bipartition of it with largest edge ratio is found using the algorithm

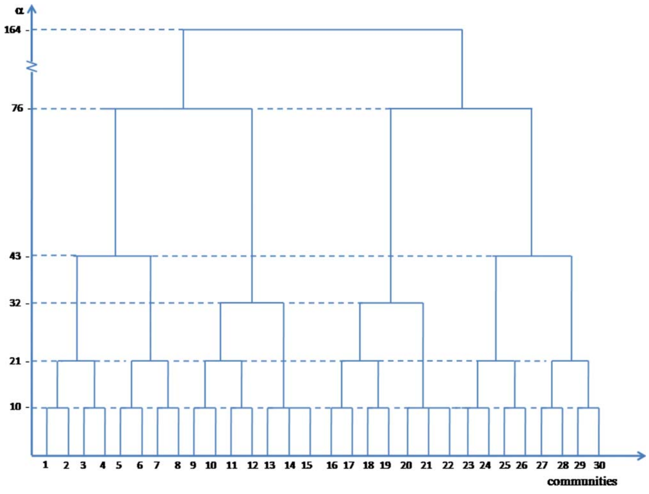

FIG. 1. (Color online) Dendrogram summarizing the resolution with the edge ratio algorithm for the first artificial data set.

of Sec. II B with formulas (9) and (10) instead of (7) and (8). When the best bipartition of the current subproblem has been found, the procedure is updated. If however it is indivisible, the subproblem is deleted and another one chosen. The algorithm stops when all remaining subproblems are indivisible.

Results can be represented on a dendrogram, which allows both tracking of the successive bipartitions and representation of the corresponding values of the edge ratios. This gives more information than simply noting successive divisions.

III. RESULTS AND COMPARISON

A. Two artificial examples

We first apply the edge ratio algorithm to two artificial examples of Fortunato and Barthelemy [45] mentioned in Sec. I.

The first example consists of a ring of cliques each joined to both of its neighbors by a single edge. As in [45], we consider the case of 30 cliques of five vertices. Maximizing modularity gives communities consisting each of two successive cliques joined by an edge instead of communities consisting of single cliques. The edge ratio algorithm does find, very quickly, communities corresponding to each of the cliques. The dendrogram summarizing the resolution is given in Fig. 1. The first bipartition, at $\alpha=164$, consists of two communities of 15 successive cliques. Each of these communities is bipartitioned at $\alpha=76$ into a community of eight successive cliques and a community of seven successive cliques. Bipartitions continue yielding communities corresponding to an equal or almost equal number of cliques. At $\alpha=10$ all communities correspond to single cliques and are shown to be indivisible.

The second example consists of two large cliques joined by a single edge and two small cliques joined by an edge and also each joined by an edge to the same large clique. Again as in [45], we consider the case where the large cliques have 20 vertices and the small ones 5. Maximizing modularity gives three communities corresponding to the two large cliques separately and to the union of the small ones. The edge ratio algorithm gives four communities which correspond to each of the cliques. The partition obtained with the edge ratio algorithm is presented in Fig. 2. The dendrogram summarizing the resolution is given in Fig. 3.

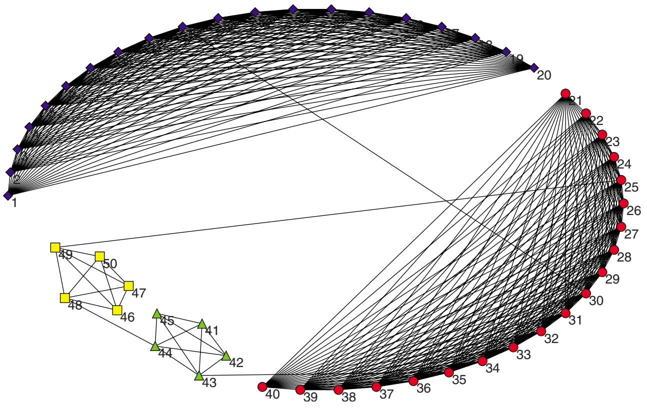

FIG. 2. (Color online) Partition obtained by the edge ratio algorithm for the second artificial data set.

B. Zachary's karate club

We now turn to data sets corresponding to various real world applications, often studied for purposes of evaluating community identification heuristics and algorithms. The first and probably the best known is Zachary's karate club data set. It describes friendship relations between 34 members of a karate club observed over two years by Zachary [48]. In that period the club splits into two groups after a dispute between the club owner and the karate instructor. The edge ratio algorithm obtains, after three bipartitions, a partition into four indivisible communities, which is quite close to those obtained by other researchers [12,32,33,41,49,50]. This partition is represented in Fig. 4. The corresponding dendrogram is depicted in Fig. 5. The first bipartition occurs at

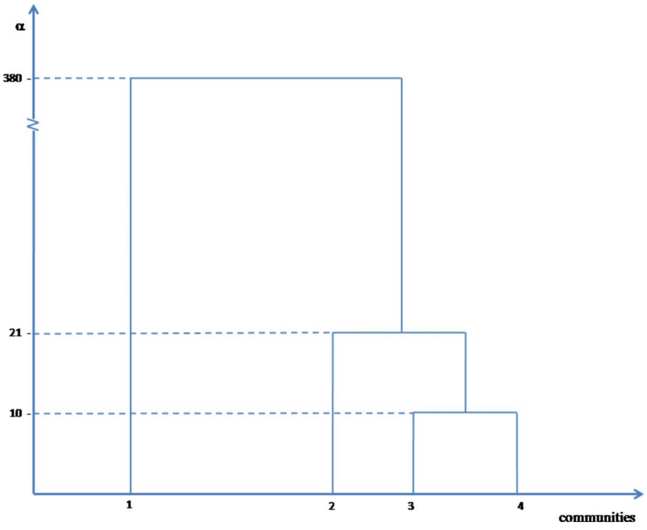

FIG. 3. (Color online) Dendrogram summarizing the resolution with the edge ratio algorithm for the second artificial data set.

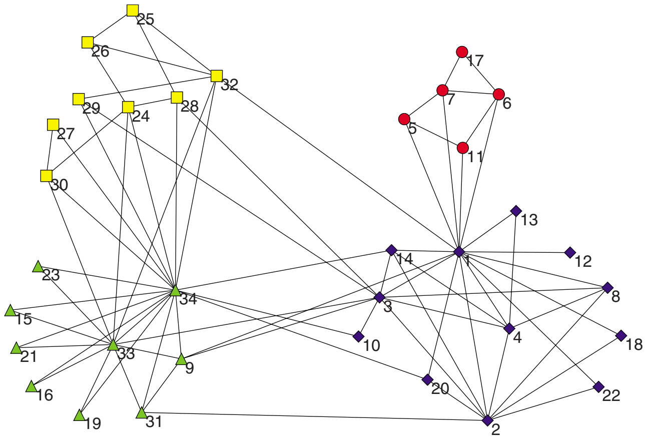

FIG. 4. (Color online) Partition obtained by the edge ratio algorithm for Zachary's karate club data set.

$\alpha=6.8$ and consists of the two following communities: $C_1=\{1,2,3,4,5,6,7,8,10,11,12,13,14,17,18,20,22\}$ and $C_2=\{9,15,16,19,21,23,24,25,26,27,28,29,30,31,32,33,34\}$. This bipartition corresponds exactly to the split of the karate club, as observed by Zachary, with the exception of member 10, which is included in the first community instead of the second one. Note that the vertex corresponding to this member is connected to two other vertices which correspond to member 3 from community 1 and member 34 from community 2. So the evidence that it should belong to one or the other community appears to be limited. It has several times been misclassified by former proposed methods, e.g., [49,50]. If vertex 10 be included in community 2 instead of community 1, the number of cut edges would remain unchanged at 10 and the edge ratio would be reduced by $\min(2\times 34/10, 2\times 34/10)-\min(2\times 35/10, 2\times 33/10)=6.8-6.6=0.2$ only. The next bipartition occurs at the lower level of $\alpha=3$ and splits the community C_1 into the two following communities: $C_3=\{5,6,7,11,17\}$ and $C_4=\{1,2,3,4,8,10,12,13,14,18,20,22\}$.

The small community C_3 is connected to one vertex of C_4 only and is fairly dense. To the best of our knowledge, it has been detected by all previous methods [12,32,33,41,49,50]. The last bipartition, of community C_2, arises at the very low level $\alpha=1.5$ and yields the two communities: $C_5=\{9,15,16,19,21,23,31,33,34\}$ and $C_6==\{24,25,26,27,28,29,30,32\}$.

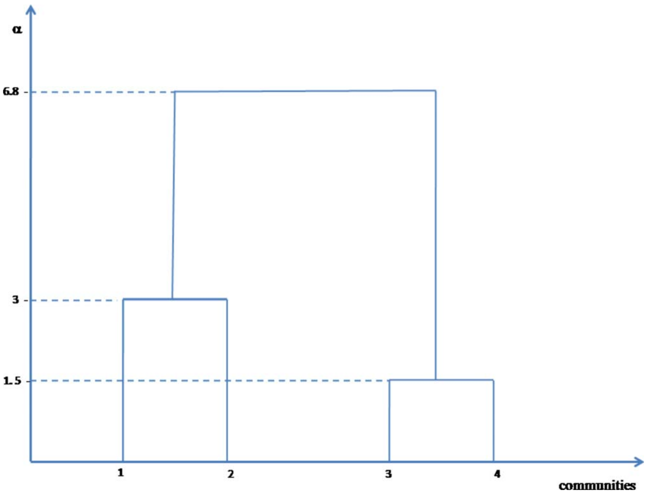

FIG. 5. (Color online) Dendrogram summarizing the resolution with the edge ratio algorithm for Zachary's karate club data set.

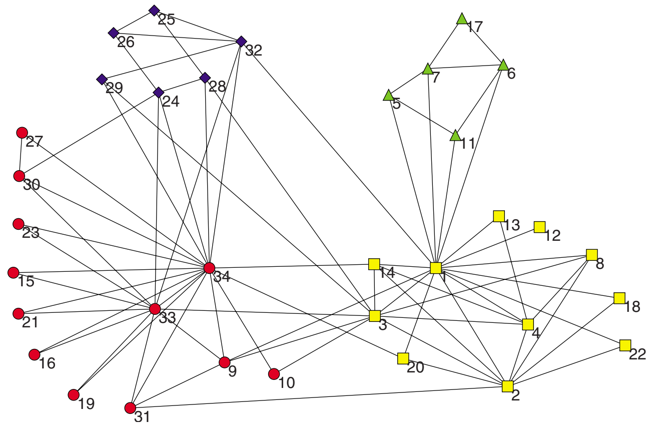

FIG. 6. (Color online) Partition obtained by the modularity-based algorithm for Zachary's karate club data set.

Comparing with results of modularity maximization, as reported for various previous methods and proved optimal by Xu *et al.* [33], we see that four communities are obtained and are close to those given by the edge ratio algorithm (Fig. 6). Indeed, community C_3 is the same, community C_4 differs only by vertex 10, and communities C_5 and C_6 differ by vertices 27 and 30 being included in C_5 instead of C_6 and vertex 10 being outside. The edge ratio for C_5 and C_6 is $\min(2\times 9/12, 2\times 16/16)=1.5$. If vertices 27 and 30 be included in community C_6 instead of C_5, the edge ratio for C_5 and C_6 would become $\min(2\times 8/10, 2\times 13/18)=1.44$, i.e., be reduced by 0.06 only.

Comparing briefly with results of betweenness-based divisive algorithm of Girvan and Newman [41] as reported in [33], we find a smaller degree of agreement. There are five communities, one of which is the isolated vertex 10 and other of which is exactly community C_3. Another community is very close to C_4 but does not include vertex 3. Vertices 25, 28, and 29 form a small community with vertex 3 and the remaining vertices form a large community including those of C_5 as well as vertices 22, 24, 26, 27, and 32.

To summarize, the edge ratio algorithm shows that there is one main bipartition at high level of α which corresponds (almost) to that one reported by Zachary, then two more bipartitions at medium and lower levels of α which thus appear to be less natural.

C. Lusseau's dolphins

A group of 62 bottlenose dolphins has been studied by Lusseau [51] for many years in Doubtful Sound, New Zealand. This led to a network with 62 vertices corresponding to the dolphins and 159 edges joining vertices associated with pairs of dolphins with frequent communications among them. This data set is also often studied, with various methods. An optimal partition into five communities for modularity maximization was obtained by Xu *et al.* [33] (these

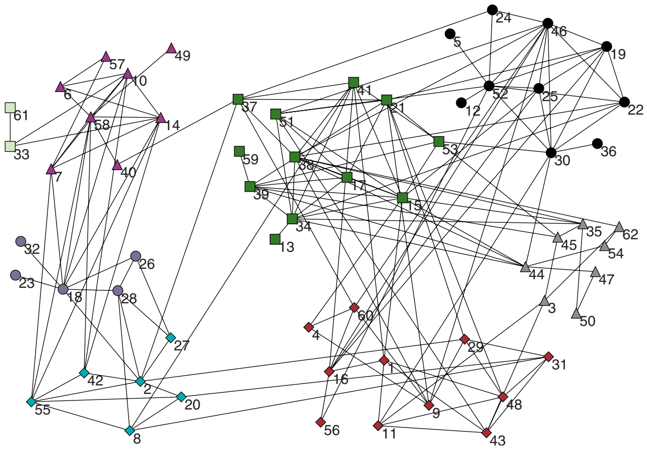

FIG. 7. (Color online) Partition obtained by the edge ratio algorithm for Lusseau's dolphin data set.

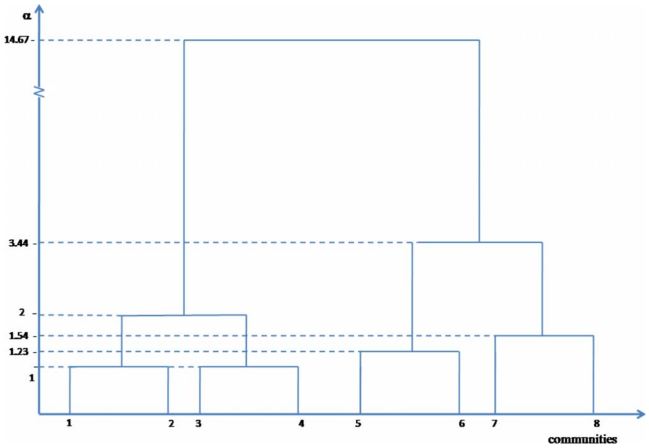

FIG. 9. (Color online) Dendrogram summarizing the resolution with the edge ratio algorithm for Lusseau's dolphin data set.

authors also obtained a rather different heuristic partition into five communities for the same criterion but using hierarchical clustering). The partition into five communities found by the former algorithm is the following: $C_1^m=\{1,3,11,21,29,31,43,45,48\}$, $C_2^m=\{2,6,7,8,10,14,18,20,23,26,27,28,32,33,42,49,55,57,58,61\}$, $C_3^m=\{4,9,37,40,60\}$, $C_4^m=\{5,12,16,19,22,24,25,30,36,46,52,56\}$, and $C_5^m=\{13,15,17,34,35,38,39,41,44,47,50,51,53,54,59,62\}$.

Applying the edge ratio algorithm yields an optimal partition into eight communities, which is represented in Fig. 7. These communities are $C_1=\{33,61\}$, $C_2=\{6,7,10,14,40,49,57,58\}$, $C_3=\{18,23,26,28,32\}$, $C_4=\{2,8,20,27,42,55\}$, $C_5=\{13,15,17,21,34,37,38,39,41,51,53,59\}$, $C_6=\{3,35,44,45,47,50,54,62\}$, $C_7=\{5,12,19,22,24,25,30,36,46,52\}$, and $C_8=\{1,4,9,11,16,29,31,43,48,56,60\}$. The partition obtained by the modularity-based algorithm is represented in Fig. 8. The corresponding dendrogram is given in Fig. 9.

Lusseau [51] noticed that two groups of dolphins, one predominantly male and one predominantly female, were separated during part of the observation period. The first bipartition, obtained at the edge ratio level of $\alpha=14.6667$, corresponds exactly to the bipartition described by Lusseau except for vertex 40 which is added to the first community instead of remaining in the second. As in the case of vertex 10 for the karate club example, vertex 40 is joined to two vertices only, one in each of the communities found. Then both communities obtained are bipartitioned at the α levels of 3.44 and 2.40. Furthermore, each of the four resulting communities is bipartitioned one more time at a level of α close or equal to 1.

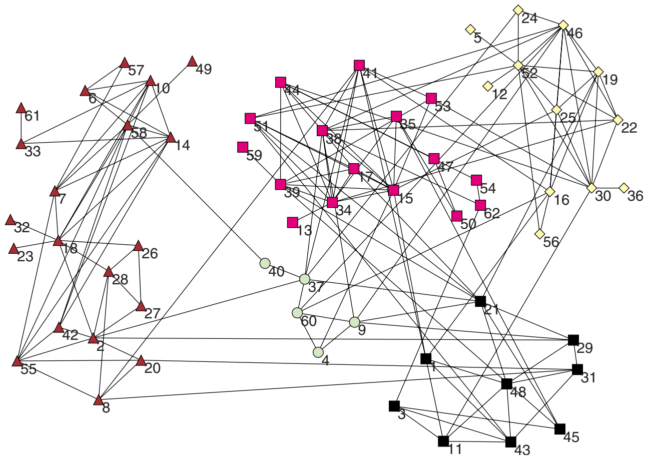

FIG. 8. (Color online) Partition obtained by the modularity-based algorithm for Lusseau's dolphin data set.

The modularity maximization partition does not separate the first left-hand side community, while the edge ratio algorithm separates it into four communities, i.e., C_1,C_2,C_3,C_4, which are thus included in the same community C_2^m. We leave the interpretation of these communities to the biologists. While the four right-hand side communities obtained by the edge ratio algorithm are sometimes fairly close to communities obtained with the modularity maximization algorithm they never coincide nor any community of one partition is included into a community of the other. Again, possible substantive interpretations of these communities are left to the biologists.

To summarize, the edge ratio algorithm finds one bipartition at high level of α which corresponds (almost) to that of Lusseau and several further partitions, one of which at $\alpha=3.44$ appears to be fairly natural.

D. Knuth on Hugo's *Les Misérables*

The next data set that we studied describes the relationships between characters in Hugo's masterpiece *Les Misérables*. Knuth [52] patiently noted the names and the interactions of all the 80 characters in this 1486 pages long novel [53]. A graph was then built with 77 vertices associated to characters which interact (not including, e.g., King Louis-Philippe, whose character is illustrated and discussed without interactions with other characters of the novel) and 257 edges associated with pairs of characters appearing jointly in at least one of the many and usually short chapters of the novel. The data are available at [52,54]. This network was studied by Newman and Girvan [12] with their betweenness-based divisive hierarchical algorithm, leading to a partition into 11 communities with a modularity $Q=0.54$. More recently, Xu *et al.* [33] obtained with their mathematical programming formulation an optimal solution with six communities and modularity $Q=0.56$. The communities of the optimal partition found are the following: $C_1^m=\{1,2,3,4,5,6,7,8,9,10\}$, $C_2^m=\{11,12,14,15,16,29,30,33,34,35,36,37,38,39,45,46\}$, $C_3^m=\{13,17,18,19,20,21,22,23,24,31,32\}$, $C_4^m=\{25,26,28,41,42,43,69,70,71,72,76\}$, $C_5^m=\{27,40,44,50,51,52,53,54,55,56,57,73\}$, and $C_6^m=\{47,48,49,58,59,60,61,62,63,64,65,66,67,68,74,75,77\}$.

We reproduced this result using a recent implementation of the Grötschel-Wakabayashi algorithm for clique partitioning [55]. Using the edge ratio algorithm we obtained a partition into ten communities, which is the following: $C_1=\{74,75\}$, $C_2=\{49,56,58,59,60,61,62,63,64,65,66,67,68,77\}$, $C_3=\{26,40,41,42,43,69,70,71,72,76\}$, $C_4=\{47,48\}$, $C_5=\{1,2,3,4,5,6,7,8,9,10\}$, $C_6=\{30,35,36,37,38,39\}$, $C_7=\{17,18,19,20,21,22,23\}$, $C_8=\{29,45,46\}$, $C_9=\{50,51,52,53,54,55,57\}$, $C_{10}=\{11,12,13,14,15,16,24,25,27,28,31,32,33,34,44,73\}$

The numbering of vertices corresponds to the order of first appearance of the associated characters in the novel. It is therefore to be expected that each community will contain several vertices with successive indices, all the more so if the communities correspond to subplots rather than involving characters in the central plot of the novel. One measure of this regularity is the number of *breaks* in the list of vertices of each community, i.e., the number of times that two vertices do not have successive indices, after ranking them in increasing order. The modularity partition has 17 breaks and the edge ratio partition has 13 breaks. The ten communities obtained by the edge ratio algorithm can be divided into three groups. (i) *Communities corresponding to subplots*, usually around some main character, i.e., C_5, C_6, C_7, and C_9, which have zero or one break. For instance, community C_5 consists of characters playing a role in the life of Bishop Myriel (vertex 1). Note that these characters do not interact between themselves with the exception of Myriel's sister and his servant. Consequently, there are ten inner edges only. As another example, community C_7 corresponds to the four students, Tholomyès, Listolier, Fameuil, and Blachevelle, and their *grisettes*. This community has maximum density or, in other words, it is a clique. The heroine Fantine (vertex 24) is not in this community despite being connected to all of its members as, due to other interactions, she belongs to the main plot community. (ii) *Community close to the central plot*, which have several breaks, i.e., C_2, C_3, and C_{10}. For instance, C_{10} contains vertices associated with the main hero redeemed convict Jean Valjean (vertex 12), his nemesis inspector Javert (vertex 28), as well as Fantine (vertex 24). (iii) *Small communities of unimportant characters*, i.e., C_1, C_4, and C_8. For instance, community C_1 consists of *child* 1, *child* 2, to which Hugo did not deem necessary to give names.

The modularity maximizing algorithm finds community C_{10} as does the edge ratio algorithm, but all other five communities that it finds have two breaks or more. Community C_3^m adds not only Fantine (vertex 24) to the group of students and their grisettes but also the old lady Marguerite (vertex 13) and the nuns Perpetue and Simplice (vertices 31 and 32), which have very few connections to the other members of that community.

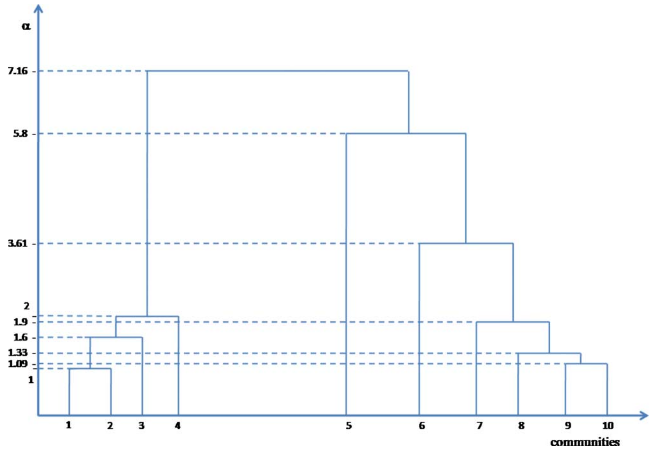

FIG. 10. (Color online) Dendrogram summarizing the resolution with the edge ratio algorithm for Hugo's *Les Misérables* data set.

The dendrogram summarizing the working of the edge ratio algorithm is given in Fig. 10 and also provides interesting information. First one can note that two groups of communities are separated at the very high α level of 7.16 and both of these groups present *chaining effects*, i.e., in all divisions one of the communities will not be separated anymore. Communities C_1-C_4 on the left-hand side are difficult to divide; i.e., the values of α go from 1 to 2 only. Communities C_5-C_{10} separate more easily: first community C_5 (bishop Myriel) at level 5.8, then community C_6 (affaire Champmathieu) at level 3.61, then community C_7 (students) at level 1.9, and finally community C_8 at level 1.33. The community C_9 (Gillesnormand family) only separates from C_{10} (main plot) at level 1.09.

To summarize, it appears that the edge ratio algorithm recognizes both dense and sparse communities and gives a quantitative measure of how close or how far they are, i.e., how difficult they are to separate. Moreover, it appears to be more selective in the inclusion of vertices into communities than modularity maximization, as well as less prone to the resolution limit. The partitions obtained by the edge ratio algorithm and by the modularity-based algorithm are represented in Figs. 11 and 12.

E. Krebs' political books

The third data set we studied deals with copurchasing of political books on Amazon.com. Krebs [56] listed 105 titles which are represented by vertices of a network with 441 edges. On the basis of titles and reviews, Newman [43] classified these 105 books as liberal (l), conservative (c), or neutral (n). This data set was studied with the modularity maximization criterion by Newman [43] using his hierarchical divisive spectral heuristic, by Agarwal and Kempe [50] using heuristically a mathematical programming model and randomized rounding, as well as by Brandes *et al.* [32] using an integer programming formulation and an algorithm close to those of Grötschel and Wakabayashi [55]. We reproduced these results with our version of the Grötschel-Wakabayashi algorithm. The optimal partition for modularity maximization contains the following five communities:

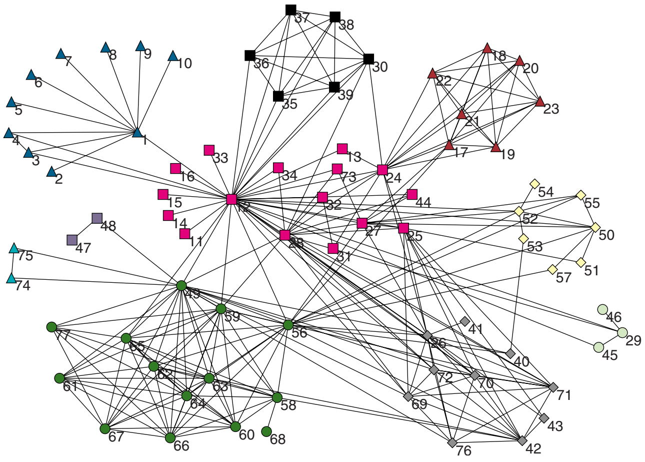

FIG. 11. (Color online) Partition obtained by the edge ratio algorithm for Hugo's *Les Misérables* data set.

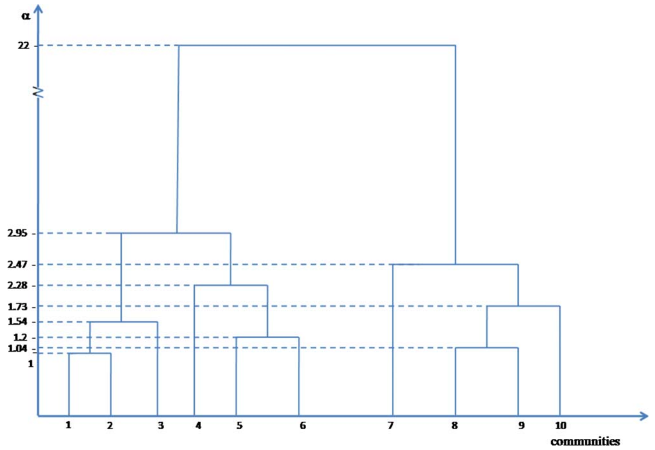

FIG. 13. (Color online) Dendrogram summarizing the resolution with the edge ratio algorithm for Krebs' political books data set.

C_1^m={1,2,3,5,6,7,8,19,29,30} with 6 n and 4 c, C_2^m={4,9,10,11,12,13,14,15,16,17,18,20,21,22,23,24,25,26,27,28,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,54,55,56,57} with 39 c, C_3^m={31,32,60,61,62,63,64,67,71,72,72,74,75,76,77,78,79,80,81,82,83,84,85,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103} with 38 l, 1 n, and 1 c, C_4^m={49,50,58} with 1 n and 2 c, and C_5^m={51,52,53,59,65,66,68,69,70,86,104,105} with 5 l, 4 n, and 3 c. These five communities consist of two large ones with no (for c) or very few (for l) misclassifications, two small communities with both n and c books, and one community with all three categories. We count misclassifications as follows: 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. The total number of misclassifications for the modularity maximization algorithm is 9.

The optimal partition obtained with the edge ratio algorithm is the following: C_1={67,74,82,85,87,89,90,94,979,98,101}, C_2={62,95,96,102,103}, C_3={60,61,63,64,100}, C_4={31,32,71,72,73,75,76,77,78,79,80,81,83,84,88,91,92,93,99}, C_5={68,104,105}, C_6={29,52,53,59,65,66,69,70,86}, C_7={9,10,12,14,18,21,23,25,27,278,41,42,43,44,45,46,47,48,49,50,51,54,55,57,58}, C_8={35,36,37,38,39,40}, C_9={4,11,13,15,16,17,19,20,22,24,26,33,34,56}, and C_{10}={1,2,3,5,6,7,8,30}. Again, the total number of misclassifications is 9.

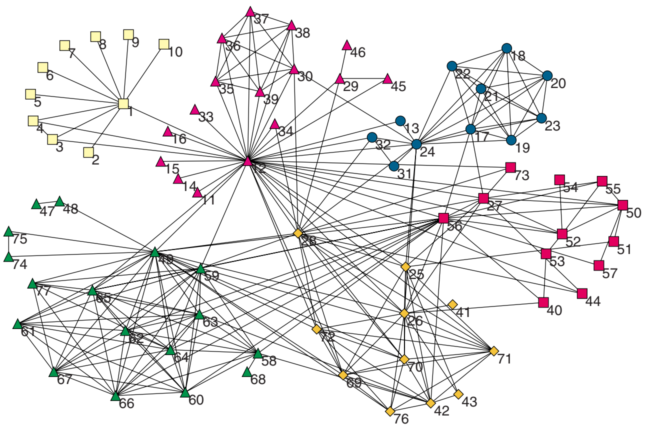

FIG. 12. (Color online) Partition obtained by the modularity-based algorithm for Hugo's *Les Misérables* data set.

The dendrogram summarizing the resolution with the edge ratio algorithm is presented in Fig. 13. At a very high level of α, i.e., 22, there is a division into two groups that clearly corresponds to liberal and to conservative books. Indeed, the left-hand side group, which eventually splits into six communities, contains vertices associated with 43 liberal books, six neutral and three conservative ones. The right-hand side group contains vertices associated with 46 conservative books, seven neutral and zero liberal ones. So in these sample purchasers of mostly conservative books never buy liberal ones, but occasionally buy a neutral one, while purchasers of mostly liberal books occasionally buy a conservative or a neutral book. A further division of the left-hand side group separates at level α=2.95 into a subgroup with communities C_1,C_2,C_3 which only contain liberal books and another subgroup which contains communities C_4,C_5,C_6 whose members sometimes buy neutral or conservative books. Whether it is to strive toward objectivity or to comfort prejudices, simultaneous purchasers of liberal and conservative books appear to be limited. There are several further partitions among homogeneous groups which might indicate some latent dimensions which cannot be explained only in terms of the l, n, and c categories. The partitions obtained by the edge ratio algorithm and by the modularity-based algorithm are represented in Figs. 14 and 15.

F. Girvan and Newman on American football games

As a final example of a real network, we consider the network in [41] representing the schedule of games between American college football teams in the Fall 2000. There are 115 teams, most of which belong to one or another of 11 conferences, with intraconference games more frequent than others. There are also five independent teams. This network has been analyzed by Girvan and Newman [41] with their betweenness-based divisive algorithm and by Radicchi *et al.* [42] using another divisive algorithm based on the frequency

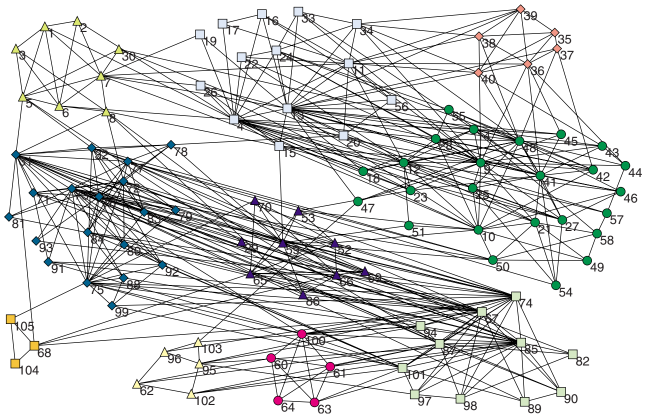

FIG. 14. (Color online) Partition obtained by the edge ratio algorithm for Krebs' political books data set.

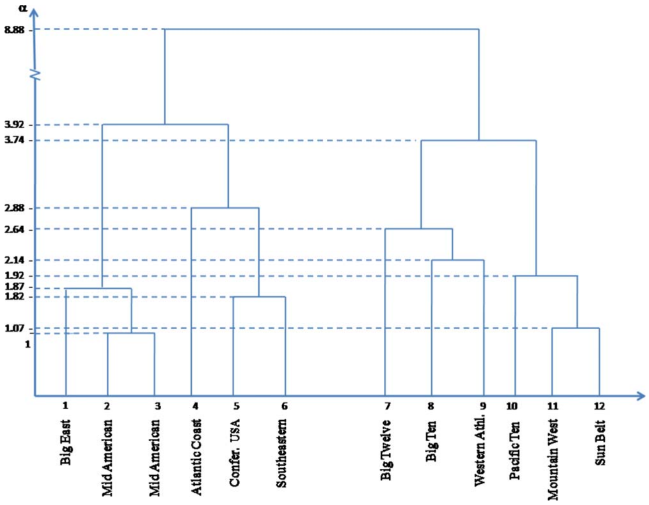

FIG. 16. (Color online) Dendrogram summarizing the resolution with the edge ratio algorithm for the Girvan-Newman football game data set.

of small cycles containing an edge. Newman [49] reported on the application of his agglomerative hierarchical clustering heuristic to maximize modularity. The same objective has been considered by Agarwal and Kempe [50], which use mathematical programming to find an initial, not necessary integer, solution followed by randomized rounding. Newman obtained a modularity of $Q=0.546$, but his algorithm found only six communities, often containing two or more conferences. Agarwal and Kempe obtained a modularity of $Q=0.6046$. Using again our implementation of the Grötschel-Wakabayashi [55] algorithm for clique partitioning led to the solution, for the first time, of the American college football team problem with a guarantee of optimality (a comparison of mathematical programming algorithms for modularity maximization is currently under way and will be reported in a future paper). This computation also showed that the heuristic solution of Agarwal and Kempe was indeed optimal. To compare results obtained with modularity maximization and edge ratio criteria for this example, one may consider two questions: (i) does the heuristic or algorithm find the structure of the problem, i.e., the number of communities, and (ii) how many misclassification errors are made. The Agarwal-Kempe heuristic found ten communities, thus missing one of the conferences. The edge ratio algorithm found 12 communities, two of which correspond to the same conference (Mid-American), but in one case also to two additional independent teams. Modularity maximization misclassifies ten teams, i.e., attributes them to a community of which they do not form the majority (the five independent teams not being counted). The edge ratio algorithm does better, as it misclassifies six teams only (again not considering independent teams). It is worth noting that the six misclassifications made by the latter algorithm are among the ten made by the former one. Results of the divisive heuristics of Girvan and Newman [41] and Radicchi *et al.* [42] are more difficult to interpret. In both cases the structure was recovered; i.e., 11 communities were found. While it is stated in [42] that "the observed communities perfectly correspond to the conferences, with the exception of the six members of the independent conference, which are misclassified," there are seven misclassifications in the case of Radicchi *et al.* (not counting the misclassifications of the *five* independent teams) and four teams (Nevada Las Vegas, Southern California, Louisiana Monroe, and Louisiana Lafayette) have inadvertently been omitted.

The dendrogram summarizing the resolution is given in Fig. 16 and conferences predominant in each of the communities are listed below. Observe that the only conference split among two communities is Mid-American and corresponds to a level of α equal to 1. So, taking strict inequality in the weak condition will give 11 communities, each corresponding to a single conference. Otherwise, not surprisingly, partitions follow geographic lines, as geographically close teams play more often together than far away ones. The first partition at level $\alpha=8.88$ corresponds to six communities located on the eastern half of USA and the other to the six communities located on the western half. Other bipartitions can be explained in a similar way.

To summarize, the edge ratio algorithm finds the structure of the data set with few misclassifications and through the dendrogram explains further the classification by geographi-

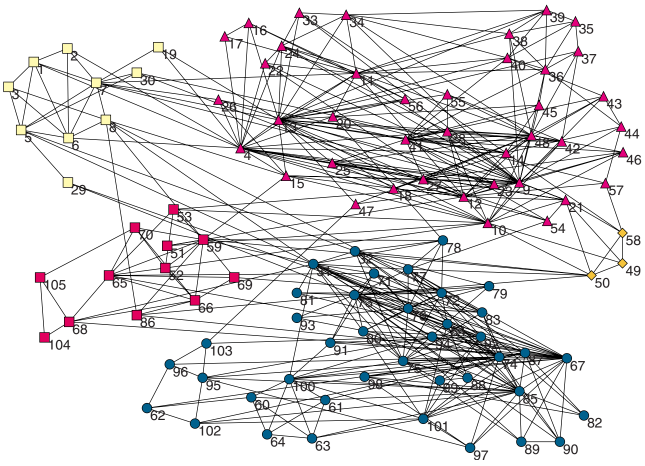

FIG. 15. (Color online) Partition obtained by the modularity-based algorithm for Krebs' political books data set.

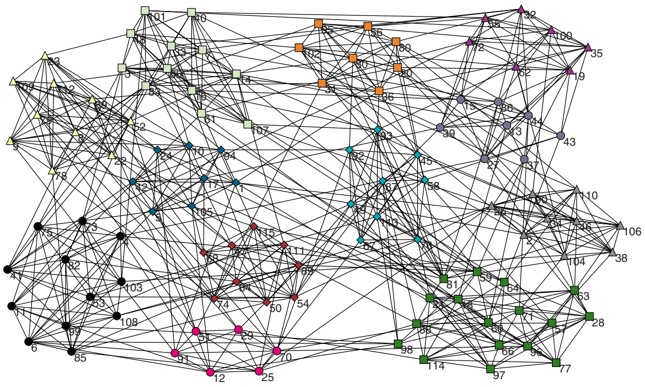

FIG. 17. (Color online) Partition obtained by the edge ratio algorithm for the Girvan-Newman football game data set.

cal considerations. In this case, modularity maximization does neither. The partitions obtained by the edge ratio algorithm and by the modularity-based algorithm are represented in Figs. 17 and 18.

G. Reduced version of benchmark of Girvan and Newman

Two anonymous referees suggested that the algorithm of this paper should be tested on standard benchmarks, i.e., those of Girvan and Newman (GN) [41] and of Lancichinetti *et al.* [57], which are often used in comparison of algorithms, e.g., [58,59]. These networks were generated using the code of Fortunato *et al.* [60].

The algorithm proposed in this paper is an exact one and requires a computing time rapidly increasing with the size of the data sets under study. To illustrate, computing times on the previous problems analyzed in this section are given in Table I. Moreover, randomly generated instances tend to be more time consuming than other ones that exhibit some structure. Consequently, we have kept the framework of the GN data set but reduced size. We consider networks with 32 entities, four equal communities of eight entities, vertices with degree 8, and a ratio of outer edges to inner edges controlled by a parameter μ. Again, modularity is computed exactly using the algorithm of Grötschel and Wakabayashi [55] and edge ratio using the proposed algorithm. Results are presented in Table II. The first four columns give characteristics of the networks under study. The two next columns give the number of communities found and the percentage of correctly classified vertices for the edge ratio algorithm. The next column gives the mutual information I_{er} between the partition found by the edge ratio algorithm and the one *a priori* known. It is computed as described in the paper of Danon *et al.* [58]. The last columns do the same for the modularity maximization algorithm. The percentage of correctly classified vertices is obtained by making a tableau with as many rows as communities obtained by the algorithm and as many columns as there are communities in the problem generated. Then the number of common elements is inserted in every cell. The number of correctly classified vertices is taken to be the value of an optimal solution of the corresponding assignment problem [61]. In other words, if p is the minimum number of rows or columns, then p cells are selected, one at most per line with a maximum sum. It appears

TABLE I. Computing time of edge ratio algorithm on data sets from literature. Solutions have been obtained on a 2.4 GHz Intel Xeon CPU of a computer with 8 Gbytes random access memory shared by three other similar CPU running LINUX.

Data set	n	m	Time (s)
Karate	34	78	1.51
Dolphin	62	159	259.91
Les Misérables	77	254	481.25
Political books	105	441	156395.86
Football	115	613	429266.09

TABLE II. Results obtained with edge ratio algorithm and modularity maximization algorithm on benchmark of Girvan and Newman.

				Edge ratio			Modularity maximization		
n	m	μ	Communities	Communities	Correctly classified vertices (%)	I_{er}	Communities	Correctly classified vertices (%)	I_m
32	128	0.15	4	4	100	1	4	100	1
32	128	0.2	4	4	100	1	4	100	1
32	128	0.25	4	4	100	1	4	100	1
32	128	0.3	4	4	100	1	4	100	1
32	128	0.35	4	4	100	1	4	100	1
32	128	0.4	4	4	100	1	4	100	1
32	128	0.45	4	4	90.6	0.8	4	100	1
32	128	0.5	4	4	100	1	4	100	1
32	128	0.55	4	2	46.9	0.11	4	56.3	0.48
32	128	0.6	4	2	37.5	0.12	4	53.1	0.28
32	128	0.65	4	2	40.6	0.11	4	43.8	0.18
32	128	0.7	4	3	37.5	0.10	4	43.8	0.12

that edge ratio and modularity maximization algorithms give close results when $\mu \leq 0.5$, then the edge ratio algorithm finds too few communities and classifies correctly less vertices than modularity maximization.

H. Version of benchmark of Lancichinetti *et al.*

Lancichinetti *et al.* [57] stressed that in many community identification problems the distribution of degrees, as well as the distribution of size communities, is not uniform but tends to follow power laws. Therefore, we considered problems of similar sizes as those of Sec. III G but with exponents different from 1 for these distributions. We consider a value of degree distribution $\gamma=2$ and a value of size community distribution $\beta=2$. Results are presented in Table III. Again it appears that results are fairly similar with the slight advantage for the edge ratio algorithm, with an average modularity of 73.16%, over the modularity maximization algorithm, with an average modularity of 72.13%. Results of the former algorithm are better in four cases, results of the latter one in three cases, and there are five ties. Once again the edge ratio algorithm finds slightly too few communities, with an average of 3 instead of 3.25, while the modularity maximization algorithm finds slightly too many, with an average of 3.42 instead of 3.25.

IV. CONCLUSIONS

Building upon the definition of community in the weak sense by Radicchi *et al.* [42], a criterion for a community in a network has been proposed, the edge ratio or ratio of twice the number of inner edges to the number of cut edges of that community. When bipartitioning a community, it is natural to consider the edge ratio values for both of the resulting communities. We propose therefore a locally optimal hierarchical divisive algorithm for identifying communities based on edge ratio. This algorithm was implemented and applied to both artificial and well-known real data sets with up to 115 entities.

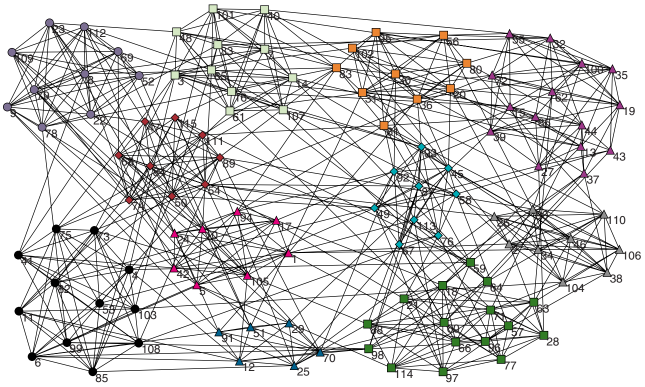

FIG. 18. (Color online) Partition obtained by the modularity-based algorithm for the Girvan-Newman football game data set.

Comparing the proposed algorithm with modularity maximization, it appears not to suffer from the resolution limit problem and usually identifies more communities, often with more precision. However, much work remains to be done and questions to be answered.

(i) *Divisive hierarchical versus partitioning algorithms*. As in [41,42,62,63], the algorithm we propose here proceeds by successive bipartitions of the network and subnetworks obtained until the indivisibility condition is satisfied for each of them. Clearly, this algorithm has the advantages and defects of many other divisive algorithms. For instance, on the one hand, as mentioned by a referee, "cutting a subgraph into more than two pieces could lead to more relevant but by the presented method undetectable structures." This would not be the case with a partitioning algorithm. On the other hand, the dendrogram associated with the divisive algorithm can give interesting information on the relationships between communities and their cohesion, as illustrated by the karate club and the football game cases discussed above. In our view, hierarchical and partitioning algorithms are comple-

TABLE III. Results obtained with edge ratio algorithm and modularity maximization algorithm on benchmark of Lancichinetti *et al.*

				Edge ratio			Modularity maximization		
n	m	μ	Communities	Communities	Correctly classified vertices (%)	I_{er}	Communities	Correctly classified vertices (%)	I_m
32	139	0.15	3	3	100	1	3	100	1
32	141	0.2	3	3	100	1	3	100	1
32	139	0.25	3	3	96.9	0.9	3	100	1
32	132	0.3	3	3	81.2	0.61	4	81.2	0.87
32	135	0.35	3	3	87.5	0.73	4	71.9	0.58
32	130	0.4	4	4	96.9	0.92	4	96.9	0.92
32	132	0.45	4	3	68.7	0.69	4	93.8	0.86
32	133	0.5	3	3	50	0.29	5	46.9	0.32
32	149	0.55	4	3	53.1	0.44	3	56.2	0.52
32	132	0.6	3	3	53.1	0.16	4	34.3	0.05
32	143	0.65	3	2	50	0.19	3	43.8	0.10
32	115	0.7	3	2	40.6	0.02	4	40.6	0.11

mentary (in previous work the second author and co-workers provided an $O(n^2 \ln n)$ exact algorithm for hierarchical divisive clustering with the diameter criterion [64] as well as a nonpolynomial exact algorithm for the NP-hard problem of partitioning with that criterion [65]). If possible, both types of algorithms should be applied and the information obtained compared. Developing a partitioning algorithm for the edge ratio criterion would complement the present work.

(ii) *Size of problems solved and heuristics*. Clearly, the proposed exact algorithm is very time consuming and this limits drastically the size of instances solved. The easiest way to palliate this defect is to replace some exact step(s) by one or several specialized heuristics. Here, the problem to be solved quickly but approximately is finding the bipartition of the network or subnetwork with maximum edge ratio. There are numerous ways to build a heuristic for that purpose and we plan to make a thorough investigation of them in future work. Indeed, a performing heuristic would help for several purposes: (a) accelerate the exact resolution by skipping a series of iterations in the dichotomous search; (b) solve rapidly instances of the size used in the benchmarks of GN and Lancichinetti *et al.* in order to make a standard comparison of algorithms; and (c) solve, if possible, fairly large instances with several thousands of vertices or more.

(iii) *Random networks and indivisible communities*. As mentioned by a referee, "in general, the configuration model is assumed to be a graph without community structure since there are no node correlations by construction; however, if the algorithm is able to divide a random graph into two subgraphs, both satisfying the weak community definition, this would be a *weakness* for the output of the algorithm." This remark points to an important and apparently little studied aspect of empirical analysis and comparison of clustering algorithms in complex networks. Indeed many clustering algorithms do provide nontrivial partitions regardless of the presence or absence of structure in at least some data sets. Finding when this happens, e.g., for which density of edges, is a theoretically difficult question. Castellano *et al.* [66] do provide estimates for a community to satisfy the weak or the strong conditions of Radicchi *et al.* [42] in the Erdős-Renyi model. Doing the same for the configuration model appears to be difficult. An empirical study of indivisibility of random networks when maximizing modularity or edge ratio and possibly other criteria would be easier and might lead to interesting insights.

(iv) *Weights*. Designing a weighted version of the edge ratio algorithm appears to be both straightforward and of interest.

[1] K. A. Eriksen, I. Simonsen, S. Maslov, and K. Sneppen, Phys. Rev. Lett. **90**, 148701 (2003).
[2] *Network Models: Handbooks in Operations Research and Management Science*, edited by M. Ball, T. Magnanti, C. Monma, and G. e. Nemhauser (Elsevier, New York, 1995), Vol. 7.
[3] S. Wasserman and K. Faust, *Social Network Analysis* (Cambridge University Press, Cambridge, 1994).
[4] R. De Castro and J. W. Grossman, Math. Intell. **21**, 51 (1999).
[5] J. Cohen, F. Briand, C. Newman, and Z. Palka, *Biomathematics* Vol. 20 (Springer-Verlag, New York, 1990).
[6] L. Ford and D. Fulkerson, *Flows in Networks* (Princeton University Press, Princeton, NJ, 1962).
[7] A. Schrijver, *Combinatorial Optimization* (Springer, New York, 2003).
[8] F. Chung and L. Lu, *Complex Graphs and Networks* (AMS, Providence, RI, 2006).
[9] A.-L. Barabási and R. Albert, Science **286**, 509 (1999).
[10] D. Watts and S. Strogatz, Nature (London) **393**, 440 (1998).
[11] S. Fortunato, Phys. Rep. (to be published).
[12] M. Newman and M. Girvan, Phys. Rev. E **69**, 026113 (2004).
[13] T. Luczak, Proceedings of the Symposium on Random Graphs, Poznań (Wiley, New York, 1989), pp. 165–182.
[14] M. Molloy and B. Reed, Random Struct. Algorithms **6**, 161 (1995).
[15] R. Guimerà and L. A. Nunes Amaral, Nature (London) **433**, 895 (2005).
[16] J. Duch and A. Arenas, Phys. Rev. E **72**, 027104 (2005).
[17] S. Lehmann and L. Hansen, Eur. Phys. J. B **60**, 83 (2007).
[18] M. Tasgin, A. Herdagdelen, and H. Bingol, e-print arXiv:0711.0491.
[19] S. Boccaletti, M. Ivanchenko, V. Latora, A. Pluchino, and A. Rapisarda, Phys. Rev. E **75**, 045102 (2007).
[20] H. Djidjev, Lect. Notes Comput. Sci. **4936**, 117 (2008).
[21] J. Mei, S. He, G. Shi, Z. Wang, and W. Li, New J. Phys. **11**, 043025 (2009).
[22] P. Schuetz and A. Caflisch, Phys. Rev. E **77**, 046112 (2008).
[23] Y. Niu, B. Hu, W. Zhang, and M. Wang, Physica A **387**, 6215 (2008).
[24] U. Raghavan, R. Albert, and S. Kumara, Phys. Rev. E **76**, 036106 (2007).
[25] M. Barber and J. Clark, Phys. Rev. E **80**, 026129 (2009).
[26] V. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, J. Stat. Mech.: Theory Exp. (2008), P10008.
[27] D. Chen, Y. Fu, and M. Shang, Physica A **388**, 2741 (2009).
[28] Y. Sun, B. Danila, K. Josic, and K. E. Bassler, EPL **86**, 28004 (2009).
[29] J. Ruan and W. Zhang, Phys. Rev. E **77**, 016104 (2008).
[30] Y. Fan, M. Li, P. Zhang, J. Wu, and Z. Di, Physica A **377**, 363 (2007).
[31] J. Kumpula, J. Saramaki, K. Kaski, and J. Kertesz, Fluct. Noise Lett. **7**, L209 (2007).
[32] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner, IEEE Trans. Knowl. Data Eng. **20**, 172 (2008).
[33] G. Xu, S. Tsoka, and L. Papageorgiou, Eur. Phys. J. B **60**, 231 (2007).
[34] ILOG, *ILOG CPLEX 11.0 User's Manual* (ILOG, Gentilly, France, 2008).
[35] J. Hartigan, *Clustering Algorithms* (Wiley, New York, 1975).
[36] L. Kaufman and P. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Series in Probability and Statistics (Wiley, New York, 2005).
[37] P. Hansen and B. Jaumard, Math. Program. **79**, 191 (1997).

[38] M. Delattre and P. Hansen, IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-2**, 277 (1980).

[39] A. Clauset, M. Newman, and C. Moore, Phys. Rev. E **70**, 066111 (2004).

[40] F. Murtagh, Comput. J. **26**, 354 (1983).

[41] M. Girvan and M. Newman, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002).

[42] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, Proc. Natl. Acad. Sci. U.S.A. **101**, 2658 (2004).

[43] M. Newman, Proceedings of the National Academy of Sciences (National Academy of Sciences, Boston, MA, 2006), pp. 8577–8582.

[44] B. Kernighan and S. Lin, Bell Syst. Tech. J. **49**, 291 (1970).

[45] S. Fortunato and M. Barthelemy, Proc. Natl. Acad. Sci. U.S.A. **104**, 36 (2007).

[46] A. Arenas, A. Fernandez, and S. Gomez, New J. Phys. **10**, 053039 (2008).

[47] J. Wang, Y. Qui, R. Wang, and X. Zhang, J. Syst. Sci. Complex. **21**, 637 (2008).

[48] W. Zachary, J. Anthropol. Res. **33**, 452 (1977).

[49] M. Newman, Phys. Rev. E **69**, 066133 (2004).

[50] G. Agarwal and D. Kempe, Eur. Phys. J. B **66**, 409 (2008).

[51] D. Lusseau, K. Schneider, O. Boisseau, P. Haase, E. Slooten, and S. Dawson, Behav. Ecol. Sociobiol. **54**, 396 (2003).

[52] D. Knuth, *The Stanford Graph Base: A Platform for Combinatorial Computing* (Addison-Wesley, Reading, MA, 1993).

[53] V. Hugo, *Les Misérables* (Gallimard, Bibliotheque de la Pleiade, Paris, 1951).

[54] M. Newman, http://www-personal.umich.edu/~mejn/

[55] M. Grötschel and Y. Wakabayashi, Math. Program. **45**, 59 (1989).

[56] V. Krebs, http://www.orgnet.com/

[57] A. Lancichinetti, S. Fortunato, and F. Radicchi, Phys. Rev. E **78**, 046110 (2008).

[58] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, J. Stat. Mech.: Theory Exp. (2005), P09008.

[59] A. Lancichinetti and S. Fortunato, e-print arXiv:0908.1062.

[60] S. Fortunato, http://santo.fortunato.googlepages.com/

[61] D. Bertsimas and J. Tsitsiklis, *Introduction to Linear Optimization* (Athena Scientific, Nashua, NH, 1997).

[62] J. Tyler, D. Wilkinson, and B. Huberman, *Communities and Technologies* (Kluwer, Deventer, 2003), pp. 81–96.

[63] J. Chen and B. Yuan, Bioinformatics **22**, 2283 (2006).

[64] A. Guénoche, P. Hansen, and B. Jaumard, J. Classif. **8**, 5 (1991).

[65] P. Hansen and M. Delattre, J. Am. Stat. Assoc. **73**, 397 (1978).

[66] C. Castellano, F. Cecconi, V. Loreto, D. Parisi, and F. Radicchi, Eur. Phys. J. B **38**, 311 (2004).