

Method to find community structures based on information centralitySanto Fortunato,¹ Vito Latora,² and Massimo Marchiori³¹*Fakultät für Physik, Universität Bielefeld, D-33501 Bielefeld, Germany*²*Dipartimento di Fisica e Astronomia, Università di Catania and INFN, Sezione di Catania, Via S. Sofia 64, 95123 Catania, Italy*³*WSC and Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*

(Received 20 February 2004; revised manuscript received 13 May 2004; published 15 November 2004)

Community structures are an important feature of many social, biological, and technological networks. Here we study a variation on the method for detecting such communities proposed by Girvan and Newman and based on the idea of using centrality measures to define the community boundaries [M. Girvan and M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002)]. We develop an algorithm of hierarchical clustering that consists in finding and removing iteratively the edge with the highest information centrality. We test the algorithm on computer generated and real-world networks whose community structure is already known or has been studied by means of other methods. We show that our algorithm, although it runs to completion in a time $O(n^4)$, is very effective especially when the communities are very mixed and hardly detectable by the other methods.

DOI: 10.1103/PhysRevE.70.056104

PACS number(s): 89.75.Hc

I. INTRODUCTION

Network analysis has revealed as a powerful approach to understand complex phenomena and organization in social, biological and technological systems [1–5]. In the framework of network analysis a given system is modeled as a graph in which the nodes are the elements of the system, for instance the individuals in a social system, the neurons in a brain and the routers in the Internet, and the edges represent the interactions, social links, synapses and electric wirings respectively, between couples of elements. A lot of interest has been focused on the characterization of various structural and locational properties of the network [1–5]. Among the others, an important property common to many networks is the presence of *subgroups* or *communities*.

For instance, in *social networks* some individuals can be part of a tightly connected group or of a closed social elite, others can be completely isolated, while some others may act as bridges between groups. The differences in the way that individuals are embedded in the structure of groups within the network can have important consequences on the behavior they are likely to practice. The division of the individuals of a social network into communities is a fundamental aspect of a social system. In fact, subgroups in social systems often have their own norms, orientations and subcultures, sometimes running counter to the official culture, and are the most important source of a person's identity [2]. For this reason one of the main concerns, since the very beginning of social network analysis, has been the definition and the identification of subgroups of individuals within a network. And the first algorithms to find community structures have been proposed in social network analysis.

Subgroups are also important to other networks. The presence of subgrouping in *biological* and *technological* networks may hinder important information on the functioning of the system, and can be relevant to understand the growth mechanisms of such networks. In fact, communities in the World-Wide-Web may represent pages on common topics, while community in cellular [6] and genetic networks [7]

might represent functional modules [8]. For this reason, the techniques to find the substructures within a network provide a powerful tool for understanding the structure and the functioning of the network.

In this paper we present a new method to discover community structures that uses the recently introduced *information centrality measure* [9,10], based on the concept of network global efficiency [11,12]. The information centrality is here used to quantify the relevance of each of the edges in the network. The method consists in finding and removing the edges with the highest centrality score until the network breaks up into components.

The paper is organized as follows. In Sec. II we review the definitions of cliques and cohesive subgroups and the standard methods for finding community structures in networks. In Sec. III we propose the new method and describe its implementation. In Sec. IV we discuss the application of the algorithm to computer-generated networks for which there is already a knowledge and control on the existing subgroups. We show that the algorithm, although slower than the best methods on the market, can be extremely effective at discovering community structures, especially when the communities are very mixed and hardly detectable. Finally in Sec. V we discuss a number of applications to real-world networks. In Sec. VI we present our conclusions.

II. DEFINITION OF COHESIVE SUBGROUPS

Social analysts were the first to formalize the idea of communities and to devise mathematical measures of the number and cohesion of communities. Here we review the most important definitions developed for social systems. For this reason the discussion of this section will be mainly in terms of social networks, although, as we will see in the following sections, the ideas of community structures applies as well to other networks. A *community*, or *cluster*, or *cohesive subgroup* is a subset of individuals among whom there are relatively strong, direct, intense ties. The starting point of all the

definitions and measures is the concept of subgraph. A *subgraph* is any collection of nodes selected from the nodes of the whole graph, together with a subset of the edges connecting those nodes. A random sample of points in a graph representing a social system is for example a subgraph but it is not likely to correspond to any meaningful social group. The notion of a meaningful social group is based on the property of cohesion among the various members of the subgraph. However the cohesion of a subgraph can be quantified by using various different properties of the ties among subsets of nodes. The choice of a particular property instead of another depends on the researcher's decision that a particular mathematical criterion can be given a meaningful and useful sociological interpretation. The general aim is to define a meaningful social category by investigating the structural properties of the whole graph and finding the naturally existing communities into which the social network can be divided.

The literature on cohesive subgroups contains various ways to conceptualize the idea of subgroups in social networks. In particular, there are four main ideas that take into account four different structural properties [1]. The resulting four categories of cohesive subgroups are sorted in such a way that going from the first to the last one we weaken the properties that the subgroups have to fulfill. We briefly present these ideas for one-mode, nondirected, nonvalued graphs.

(1) *The mutuality of ties.* Cohesive subgroups based on the mutuality of ties require that all pairs of subgroup members choose each other. This idea is formalized in the definition of cliques. A *clique* is a maximal complete subgraph of three or more nodes, i.e., a subset of nodes all of which are adjacent to each other and there are no nodes that are also adjacent to all the members of the clique.

(2) *The closeness or reachability* of the members of the subgroup. Since the definition of clique is rather strong and restrictive for real social networks, a number of extensions of the basic idea have been proposed. Cohesive subgroups based on reachability require that all the members are reachable from each other. The *n*-cliques extend the notion of cliques, weakening the requirement of adjacency among all the subgroup members. An *n*-clique is a maximal subgraph in which the largest geodesic distance between any two nodes is no greater than *n*. When *n*=1 we go back to the concept of clique. 2-cliques are subgraphs in which all nodes need not to be adjacent but are reachable through at most one intermediary. In 3-cliques all nodes are reachable through at most two intermediaries, and so on.

A definition that will be important in the following of the paper is that of component. A *component* is the maximal connected subgraph, i.e., a subgraph in which there is a path between all pairs of nodes, while there is no path between a node in the subgraph and any node not in the subgraph.

(3) *The frequency of ties* among members. This idea of cohesive subgroups is based on restrictions on the minimum number of actors adjacent to each other in a subgroup. Whereas the concept of *n*-clique involves increasing the permissible path lengths, an alternative way to relax the strong assumption of cliques involves reducing the number of other nodes to which each node must be connected. A *k*-plex is a

maximal subgraph containing *n* nodes in which each node is adjacent to no fewer than *n-k* nodes in the subgraph. Compared to *n*-clique analysis, *k*-plex analysis tends to find a relatively large number of smaller groups.

(4) *The relative frequency of ties* among subgroup members compared to nonmembers. This idea of cohesive subgroups is different from the previous three because it is based on the comparison of ties within the subgroup to ties outside the subgroup [14]. In this way cohesive subgroups are seen as areas of relatively high density in the graph, parts that are locally denser than the field as a whole. The *LS* set is the simplest formal definition of a subgroup in this class. An *LS set* is a set of nodes *S* such that any of its proper subsets (i.e., any possible subset of nodes that can be selected from the nodes in *S*) has more ties to its complement within *S* than to the outside of *S* [15]. The fact that *LS* sets are related by containment implies that there is a hierarchy of *LS* sets in a graph. The definition of lambda sets extends that of *LS* sets, and is based on the concept of edge connectivity. The edge connectivity of a pair of nodes *i* and *j* is equal to the minimum number of edges that must be removed from the graph in order to leave no path between the two nodes. A set of nodes *S* is a *lambda set* if any pair of nodes in *S* has larger edge connectivity than any pair of nodes consisting of one node within *S* and a node outside *S* [16]. Lambda sets are based on the idea that a cohesive subgroup is relatively robust, namely it is hard to disconnect by the removal of edges. An alternative approach based on the same idea is to consider if there are edges in the graph which, if removed, would result in a disconnected structure. This approach is easy to implement into an algorithmic procedure and allows to develop *hierarchical clustering methods*. Such methods rank and remove the edges of the network in terms of their importance, where the edge importance can be defined in different ways as will be clear in a moment. By doing this repeatedly the network breaks iteratively into smaller and smaller components until it breaks into a collection of single nonconnected nodes. The resulting hierarchical structure to clusters can be represented by *dendrograms*, or hierarchical trees, as the one reported in Fig. 1, showing the clusters produced at each step of the subdivision.

Recently, Girvan and Newman have considered two forms of edge betweenness to measure the edge importance: the shortest path betweenness and the random-walk betweenness [17–19]. The edge shortest path betweenness extends to the edges the node betweenness proposed by Freeman [20] as a centrality measure for the nodes, and is defined as the number of shortest paths between pairs of nodes that run through that edge [17]. The random-walk betweenness does consider random walks connecting all couples of nodes instead of the shortest paths (random walks have also been used to quantify the similarities-dissimilarities between nearest-neighboring nodes in other algorithms for finding communities [21]).

The algorithms by Girvan and Newman at each step identify and remove the edges that are the most between couples of nodes, in the sense that they are responsible for connecting many pairs of nodes. The method for finding community structures that we present in this paper is a modification of the method by Girvan and Newman. In our method we propose to identify directly the edges that when removed mostly

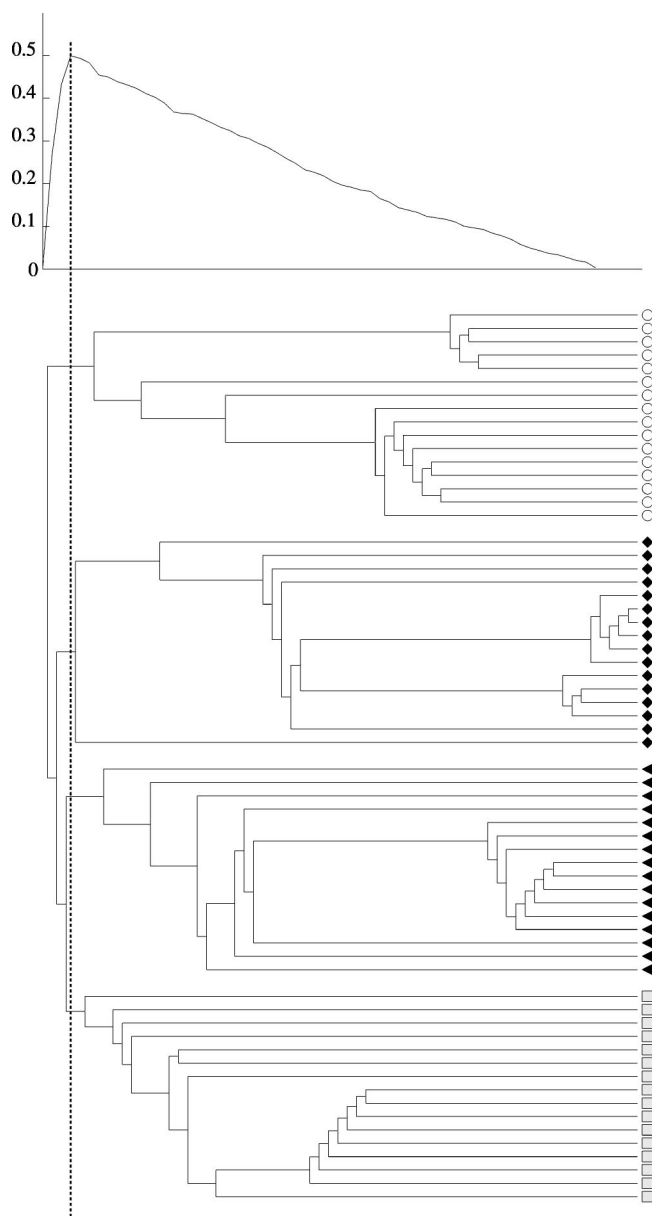


FIG. 1. Dendrogram of the communities found by applying our algorithm to a computer generated random graph with 64 vertices and 256 edges. The random graph has been obtained by dividing the nodes into 4 groups of 16 nodes each (respectively circles, diamonds, triangles, and squares) and considering $z_{in}=6, z_{out}=2$ (see text). In the top panel the value of Q corresponding to the various divisions of the dendrogram is reported.

disrupt the network's ability in exchanging information among the nodes. In fact, instead of the edge betweenness, we adopt a measure of centrality, the information centrality C^I [9,10], based on the concept of efficient propagation of information over the network [11,12]. The information centrality has revealed as an interesting quantity to characterize the centrality of the nodes of a network, and gives different results from the betweenness centrality [9]. For this reason we think that it might be useful to develop an algorithm of hierarchical clustering based on the edge information centrality.

After having described the formal definitions of cohesive subgroups based on the relative frequency of ties, we need to give some methods for assessing the cohesiveness of the subgroups. This is especially important in hierarchical clustering methods where one obtains a hierarchy of community structures, from the original graph to the extreme case in which all the nodes are disconnected: in this case the number of communities depends on the level at which the graph is partitioned, and we therefore need a criterium to say at which point to stop. One of the first measures of how cohesive a subgroup is, was proposed in Ref. [22] and is just the ratio of the number of ties (or the average strength of ties for a valued graph) within a subgroup divided by the number of ties from the subgroup to nodes outside the subgroup. This measure was recently extended in Ref. [18] by the measure of modularity that we will discuss in Sec. IV and which proves to be successful to express the degree of cohesiveness of the communities of many networks. This is why it was recently proposed in Ref. [23] to adopt the modularity itself as the quantity to maximize so to identify the best community structure. The numerical implementation of this maximization allows to analyze very large networks because it can be performed in a time which is by far shorter than the time required by all the previous algorithms.

III. OUR METHOD FOR FINDING COMMUNITIES

The algorithm for finding structures we propose here makes use of a recently introduced centrality measure [9,10], that is based on the concept of efficient propagation of information over the network [11,12]. We assume that the network we want to analyze can be represented as a connected, nondirected, nonvalued graph \mathbf{G} of N nodes and K edges. However, the extension to nonsymmetric and valued data does not present any special problem and will be considered in a forthcoming paper [13]. The graph \mathbf{G} is described by the adjacency matrix a , a $N \times N$ matrix whose entry a_{ij} is equal to 1 if i and j are adjacent and 0 otherwise. Two nodes in the graphs are said adjacent if they are connected by an edge. The entries on the main diagonal are undefined, and for convenience they are set to be equal to 0. We now give some definition that will be useful in the following. A *walk* is an alternating sequence of nodes and edges, where each edge is linked to both the preceding and the succeeding node. A *path* linking two nodes i and j is a walk from i to j in which all points and edges are distinct: the length of the path is the number of edges traversed to get from i to j . The shortest path, or *geodesic*, between i and j is any path from i to j containing the minimum number of edges.

In order to describe how efficiently the nodes of the network \mathbf{G} exchange information we use the *network efficiency* E , a measure introduced in Refs. [11,12]. Such a variable is based on the assumption that the information or communication in a network travels along the shortest paths (geodesics), and that the efficiency ϵ_{ij} in the communication between two nodes i and j is equal to the inverse of the shortest path length d_{ij} . The *efficiency* of \mathbf{G} is the average of ϵ_{ij} :

$$E[\mathbf{G}] = \frac{\sum_{i \neq j \in \mathbf{G}} \epsilon_{ij}}{N(N-1)} = \frac{1}{N(N-1)} \sum_{i \neq j \in \mathbf{G}} \frac{1}{d_{ij}} \quad (1)$$

and measures the mean flow-rate of information over \mathbf{G} . The quantity $E[\mathbf{G}]$ varies in the range $[0, 1]$, and is perfectly defined also in the case of disconnected graphs. In fact, when there is no path between i and j , we assume $d_{ij} = +\infty$ and consistently $\epsilon_{ij} = 0$. Such a property will be extremely important for our algorithm.

A measure of node centrality, the so called *information centrality*, based on the network efficiency, has been recently proposed [9]. The same measure can be used to quantify the importance of groups and classes [9,10].

Here we use such a measure to quantify the importance of an edge of the graph \mathbf{G} . The information centrality C_k^I of the edge k is defined as the relative drop in the network efficiency caused by the removal of the edge from \mathbf{G} :

$$C_k^I = \frac{\Delta E}{E} = \frac{E[\mathbf{G}] - E[\mathbf{G}'_k]}{E[\mathbf{G}]}, \quad k = 1, \dots, K. \quad (2)$$

Here by \mathbf{G}'_k we indicate a graph with N points and $K-1$ edges obtained by removing the edge k from \mathbf{G} . Notice that this measure is perfectly defined also when \mathbf{G}'_k is a disconnected graph.

The method for finding the hierarchy of cohesive subgroups in \mathbf{G} consists in the iterative removal of the edges with the highest information centrality, until the system breaks up into components. We expect that the edges that lie between communities are those with the highest information centrality, while those inside communities have a low information centrality. The general form of the algorithm is the following.

- (1) Calculate the information centrality score for each of the edges.
- (2) Remove the edge with the highest score.
- (3) Perform an analysis of the network's components.
- (4) Go back to point 1 until all the edges are removed and the system breaks up into N disconnected nodes.

As in the Girvan and Newman algorithms [17,18], the recalculation of the information centrality scores every time after an edge has been removed appears to be an important aspect of the algorithm. We will discuss this point in Sec. V. The calculation of all the shortest paths, necessary to compute the efficiency of the network, can be performed with a breadth-first search algorithm in time $O(KN)$ [24,25]. Then the calculation of the information centrality for all the edges takes a time $O(K^2N)$. This time is comparable to the time it takes to compute the random-walk betweenness for all the edges [18], but is longer than the time $O(KN)$ it takes to calculate the shortest paths betweenness for all the edges used in the method of Ref. [17]. The algorithm repeats the calculation of all the information centralities for each edge removed, i.e., K times. In conclusion, the entire community structure algorithm based on the information centrality can be completed in time $O(K^3N)$, or time $O(N^4)$ for a sparse graph. Although, as we will show in Sec. IV, the algorithm can be in some

cases better in finding community structures than the algorithm based on shortest path betweenness, for its poor performance it can be used only for graphs with up to a thousand of nodes. For extremely large networks the best algorithm to be used is the one proposed in Ref. [23] and based on the maximization of the modularity that runs in time $O(KN)$ or $O(N^2)$ on a sparse graph, or the one proposed in Ref. [26] based on the notion of voltage drops across the network and running in time $O(K+N)$.

IV. TESTING THE METHOD ON COMPUTER GENERATED NETWORKS

We first applied our algorithm to computer generated networks, i.e., random graphs constructed in such a way that they have a well defined community structure. All graphs have the same number of nodes, 128, and the same number of edges, 1024. The nodes are divided into four classes, which are the groups 1-32, 33-64, 65-96 and 97-128. We fixed to 16 the average number of edges per node, and we label the edges according to whether they connect members of the same group or not. The mixing between the classes is introduced by tuning the average number of edges connecting nodes belonging to different classes. From a generic vertex of the graph we have on average z_{in} edges which join it to other vertices of its group and z_{out} edges connecting it to vertices of the other groups. The two numbers are not independent, as we must of course have $z_{in} + z_{out} = 16$. We remark that this is the same set of graphs that Newman [23] and previously Girvan and Newman [17,18] have used to test their algorithms. In this way we are able to compare directly the role of edge betweenness and edge information centrality in determining the community structure. As a practical example we show in Fig. 1 the dendrogram corresponding to the analysis with our method of a graph of this type, where for illustration purposes we take a smaller network with 64 nodes and 8 edges per node. Here, $z_{in} = 6$ and $z_{out} = 8 - z_{in} = 2$, i.e. the network is strongly clustered. The algorithm produces a hierarchy of subdivisions of the network: from a single component to N isolated nodes, going from top to bottom in the dendrogram (left to right in the figure). To know which of the divisions is the best one for a given network, i.e., where we have to cut the hierarchical tree, we need to use a measure of the cohesiveness of the communities. The first measure of how cohesive a subgroup is, was proposed in Ref. [22]. If there are N nodes in the graph \mathbf{G} and N_S nodes in the subgroup \mathbf{S} , the cohesiveness of subgroup \mathbf{S} can be defined as the ratio of the number of ties within subgroup \mathbf{S} divided by the number of ties from \mathbf{S} to outsiders:

$$\frac{\sum_{i \in \mathbf{S}} \sum_{j \in \mathbf{S}} a_{ij}}{\sum_{i \in \mathbf{S}} \sum_{j \notin \mathbf{S}} a_{ij}}. \quad (3)$$

This measure was recently extended by Girvan and Newman in Ref. [18] into the measure of modularity, that allows us to consider more than a group at the same time and tell us how good a subdivision of \mathbf{G} in n subgroups is.

The modularity Q is defined in the following way. Let us suppose that we want to test the goodness of a subdivision of the network in n well defined communities. We expect that a good split is obtained if most of the edges fall inside the communities, with comparatively few edges joining the communities to each other. For this purpose one introduces a $n \times n$ symmetric matrix \mathbf{e} whose element e_{ij} is the fraction of all edges in the network that link vertices in community i to vertices in community j [28]. The trace of this matrix $\text{Tr } \mathbf{e} = \sum_i e_{ii}$ gives the fraction of edges in the network that connect vertices in the same community. To try just to maximize the value of the trace does not help because by considering the whole network as a single community we would get the maximal value 1 without doing any subdivision at all. Therefore we further define the row (or column) sums $a_i = \sum_j e_{ij}$, which represent the fraction of edges that connect to vertices in community i . If the network is such that the probability to have an edge between two sites is the same regardless of their eventual belonging to the same community (random network), we would have $e_{ij} = a_i a_j$. The modularity is defined as

$$Q = \sum_i (e_{ii} - a_i^2) = \text{Tr } \mathbf{e} - \|\mathbf{e}^2\|, \quad (4)$$

where $\|\mathbf{e}^2\|$ indicates the sum of the elements of the matrix \mathbf{e}^2 . This quantity then measures the degree of correlation between the probability of having an edge joining two sites and the fact that the sites belong to the same community. It now makes sense to look for high values of Q . In fact, if we take the whole network as a single community, we get $Q=0$ and we can easily get higher values by choosing subdivisions in more than just a single class. Values approaching $Q=1$, which is the maximum, indicate strong community structure; on the other hand, for a random network $Q=0$. The expression (4) is not normalized, so that Q will not reach a value of 1, even on a perfectly mixed network. For networks with an appreciable subdivision in classes, Q usually falls in the range from about 0.2 to 0.7.

In Fig. 1 we plot the Q corresponding to the classes we determined after each split. The x -coordinate represents the number of steps of the algorithm which end with a split of the network (or of one of its components, if the network is not connected). We remark that, since Q is always calculated by using the full network, Q can only vary if, after the removal of one edge, the number of components of the network changes, otherwise it keeps the value corresponding to the last subdivision. To take for the x -coordinate the number of removed edges would result in a plot with many intervals where Q stays constant and, even if that would not affect our description, we do not consider it appropriate for a presentation.

The plot presents a single peak, which exactly corresponds to the splitting of the network into the four groups. This means that the algorithm succeeds in identifying the four classes. The height of the peak is 0.499, which indicates that the network is indeed highly clustered. In Fig. 2 we show the details of the calculation. We plot the information centrality C^I of the edge removed, the global efficiency E , the number of components n of the resulting graph and the

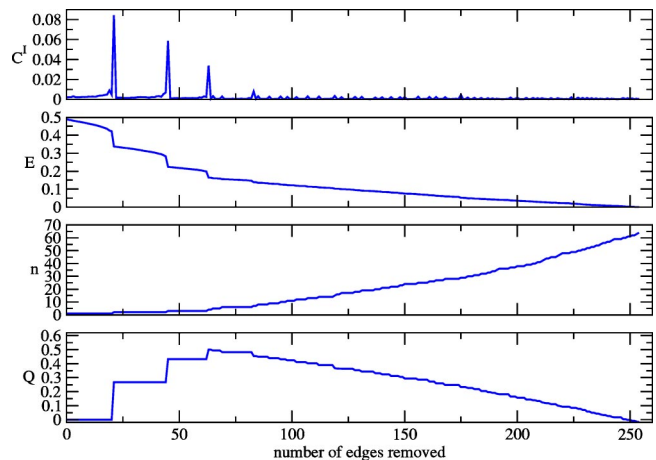


FIG. 2. Information centrality C^I of the edge removed, global efficiency E , number of components n and modularity Q for the resulting graph as a function of the number of edges removed.

value of Q as a function of the number of removed edges, i.e., as a function of the iterations of the algorithm. Each time we remove an edge with a high information centrality score, i.e. each time there is a sharp drop in the network efficiency, we also observe a sharp increase in the modularity. The height of the three main peaks in C^I is roughly proportional to the corresponding variations of Q . The correlation between C^I and Q is nontrivial, but we can give the following simple argument to explain it. Suppose that after the removal of an edge we get a split of the component A in two classes, say A_1 and A_2 . We indicate with I_{A_1}, I_{A_2}, I_A the number of edges joining pairs of vertices within A_1, A_2 and A , respectively. Furthermore, let us denote with m_{A_1}, m_{A_2}, m_A the sum of the vertex degrees of all the vertices of A_1, A_2 and A . According to Eq. (4), the modularity Q_b before the split is

$$Q_b \sim \frac{I_A}{K} - \left(\frac{m_A}{2K}\right)^2, \quad (5)$$

where K is the total number of edges of the network. Notice that I_A/K is exactly e_{AA} of Eq. (4) and $m_A/2K$ roughly a_A (with $i=A$). On the other hand, after the split, we get the modularity

$$Q_a \sim \frac{I_{A_1} + I_{A_2}}{K} - \left(\frac{m_{A_1}}{2K}\right)^2 - \left(\frac{m_{A_2}}{2K}\right)^2. \quad (6)$$

As just a few edges keep A_1 and A_2 together in A , m_A is approximately given by $m_{A_1} + m_{A_2}$. So, we come to the following expression for the modularity variation ΔQ after the split:

$$\Delta Q = Q_a - Q_b \sim \frac{I_{A_1} + I_{A_2} - I_A}{K} - \frac{m_{A_1} m_{A_2}}{2K^2}. \quad (7)$$

The first term on the right-hand side (RHS) of Eq. (7) is small, because $I_A \sim I_{A_1} + I_{A_2}$, so the dominant term is the second one, which is proportional to the product $m_{A_1} m_{A_2}$. On sparse graphs like those we are dealing with here, $m_{A_1} m_{A_2}$ is roughly proportional to the number of vertex pairs with a

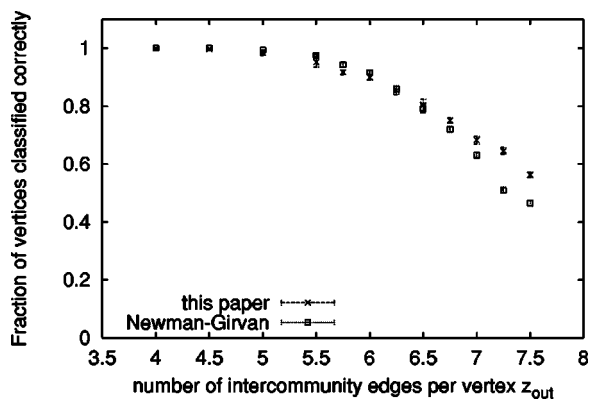


FIG. 3. Average fraction of correctly identified vertices as a function of z_{out} . Each point represents an average over 100 to 500 graphs. The comparison with the analogous results of Girvan and Newman shows that our algorithm performs better when the communities are very mixed and hardly detectable.

vertex in A_1 and the other in A_2 . This number of pairs equals the number of paths going from A_1 to A_2 , which after the split are of infinite length and give a vanishing contribution to the global efficiency of the network. The variation of the information centrality is then due to those paths, so it is proportional to ΔQ , as we find numerically.

Our aim is of course to test how the algorithm works for many different networks, and this is accomplished by considering many different realizations of the same graph and checking how many vertices are correctly classified in each case. We analyzed our artificial networks for various values of z_{out} , ranging from 4 to 7.5, with a step of 0.25. We did not do a quantitative analysis of the interval $0 < z_{out} < 4$ because there the algorithm always finds the right classes (more than 99% of successful attempts). For each value of z_{out} we produced from 100 to 500 samples, and calculated the average fraction of nodes which end up in their natural group. We plot such averages in Fig. 3 as a function of z_{out} . In the same plot we report the results obtained by using the algorithm of Girvan and Newman on the same network. We see that in the sector [4, 6] the two algorithms perform equally well; the algorithm of Girvan and Newman seems to lead in some cases to slightly better results but they are compatible with ours within errors except eventually for $z_{out}=5.75$. This is

also the region of values of z_{out} which corresponds to networks with a clear community structure. In the sector [6, 7.5], where the communities are very mixed and hardly detectable, both algorithms start inevitably to fail, but our algorithm clearly performs better. In [7, 7.5] our results are even better than the ones obtained through the modularity-based algorithm recently proposed by Newman [23]. These results may justify the extra price in terms of CPU time that we have to pay if we choose to adopt the algorithm based on the information centrality. As far as the modularity is concerned, we passed from peak values of about 0.65 for the lowest z_{out} we have taken (2) to about 0.25 for the most mixed cases ($z_{out}=7.5$).

As a further evidence of the similarities and differences between edge information and betweenness centrality we report in Fig. 4 a scatter plot of the two measures for each of the 1024 edges of the initial network, i.e., before we start the first iteration of the edge removal process. The figure shows that, as expected, the two measures are correlated, although there are some important differences. In particular we notice that the edges with the higher information are not always those with the higher betweenness. This is more evident when the communities are mixed and hardly detectable. For instance in the case $z_{out}=7$ the edge with the largest information, i.e., the one that will be removed by our algorithm is not the one with the largest betweenness.

V. APPLICATIONS TO REAL NETWORKS

After the first experiments on artificial networks, we can say that the algorithm seems promising. However, if our method is any good, it must work as well for real networks, which actually represent the systems we are mostly interested in. We present here the analysis of four networks, although we analyzed more. The first three of them, i.e., the Zachary’s karate club, the network of the American college football teams and the food web of the Chesapeake Bay, have also been studied by other authors, with other hierarchical clustering methods. In this way we can better understand what the differences between the various approaches are. The last network studied represents the interactions amongst a group of 20 monkeys.

A. Zachary’s karate club

The first example we considered is the famous karate club network analyzed by Zachary [27]. It consists of 34 persons

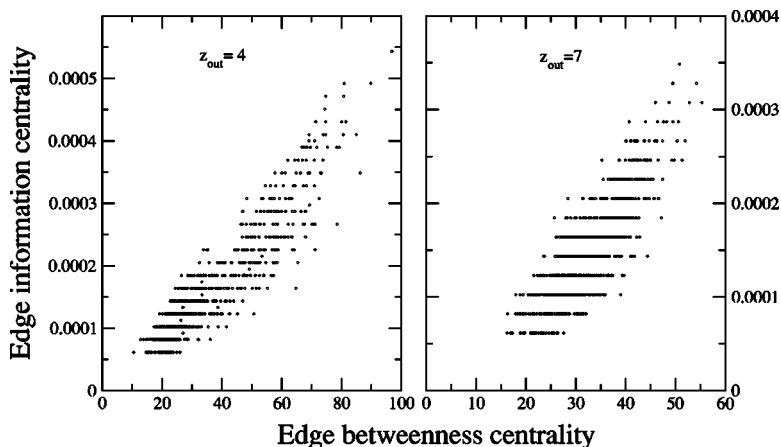


FIG. 4. Correlation between edge information centrality and betweenness centrality. Each point of the scatter plot refers to an edge of an artificially generated network with 128 nodes and 1024 edges. We consider the two values $z_{out}=4$ and $z_{out}=7$, respectively representing a case in which the communities are clearly separated and a case in which the communities are mixed and hardly detectable.

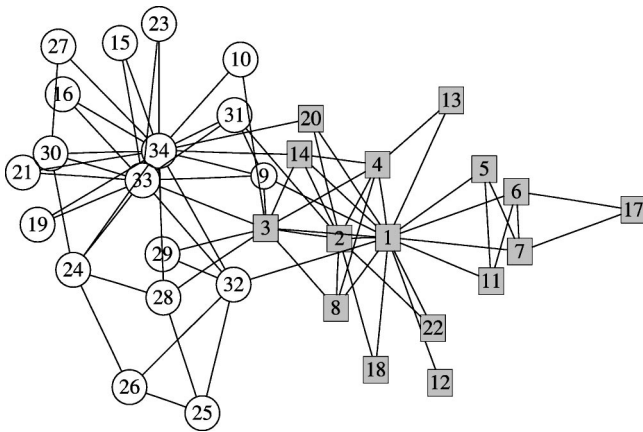


FIG. 5. The karate club network of Zachary (figure taken from Girvan and Newman [18]).

(78 edges) whose mutual friendship relationships have been carefully investigated over a period of two years. Due to contrasts between a teacher and the administrator of the club, the club split into two smaller ones. The questions we want to answer are the following: Is it possible, by studying the network community structures before the network splitting, to predict the behavior of the network and in particular to identify the two classes? Moreover, according to the network structure will a possible conflict most likely involve two factions or multiple groups? The network is presented in Fig. 5, where the squares and the circles label the members of the two groups. The results of our analysis are illustrated in the dendrogram of Fig. 6.

The first edge which gets removed is the one linking node 12 to the rest of the network. This edge corresponds to the edge between node 12 and node 1, an edge having the largest information centrality (0.024) and a medium value of betweenness (66) as shown in the scatter plot reported in Fig. 7. Notice also that the edge with the highest betweenness (142.79) is the edge connecting node 1 with node 32. The removal of the first edge then leads to the isolation of node 12. This is a feature that we encountered other times in our analyses. The early separation of a single node or of a small group is due to the fact that a system often loses more efficiency because of such splits than through the removal of intercommunities edges.

To see why this is so, let us consider the simple example of Fig. 8, describing a network G with N nodes composed by two cohesive subgroups, namely G_1 with N_1 nodes, and G_2 with N_2 nodes ($N_1 \sim N_2 \ll 1$), and by the two nodes k , which is joined to the network via a single edge (like node 12 in the karate club) and i , bridging G_1 to G_2 . In such a case the separation of the node k leads to a decrease of efficiency proportional to the number of remaining nodes, i.e., $\Delta E_{k\text{-split}} \propto O(N)$. In fact, because of the single edge, the shortest paths between pairs of nodes different from k are not affected by the removal of the edge, so the only contributions come from the paths from k to the rest of the network, which are $N-1$. On the other hand, the removal of the edge linking i to G_1 influences the lengths of $N_1 \times N_2$ shortest paths, so that $\Delta E_{\text{int-comm}} \propto O(N^2)$. In such a case, the edge standing between the two communities G_1 and G_2 will be the first one

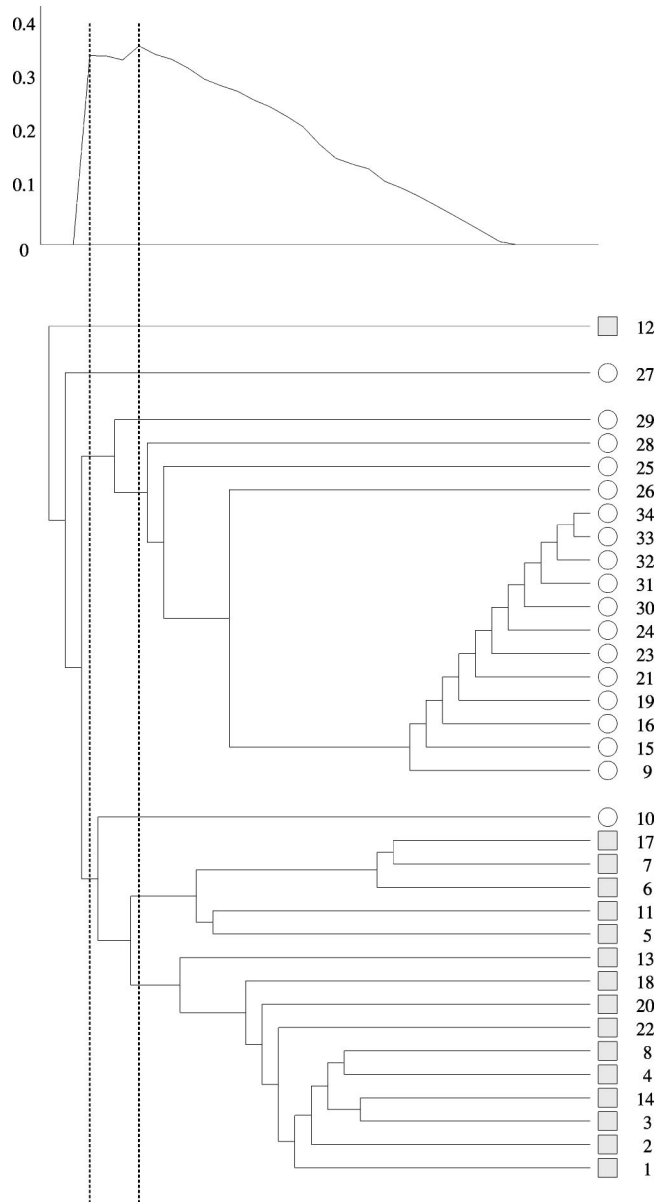


FIG. 6. Dendrogram of the communities of the karate club. Initially one has the split of two loosely bound nodes, 12 and 27, from the rest of the network. After that the two communities, with the exception of node 10 (and of the two above-mentioned nodes), are correctly identified. The separation of the two communities corresponds to a peak in the modularity Q .

to be removed. But this is not always the case, since a simple modification of the network considered in the figure would lead to a different result. In fact, if we now suppose that node i is connected to G_1 through two edges (as for the connection between node i and G_2) instead of a single one, then the algorithm will see the graph composed by G_1, G_2 and i as a more cohesive structure than before and the first edge to be removed will be the one connecting k to G_1 .

Going back to the dendrogram of Fig. 6, we see that after node 12 is removed from the network of the karate club, also the loosely bound node 27 (just two edges) isolates from the rest. The third split finally separates the two big groups. At this stage we have four components, two isolated nodes (12

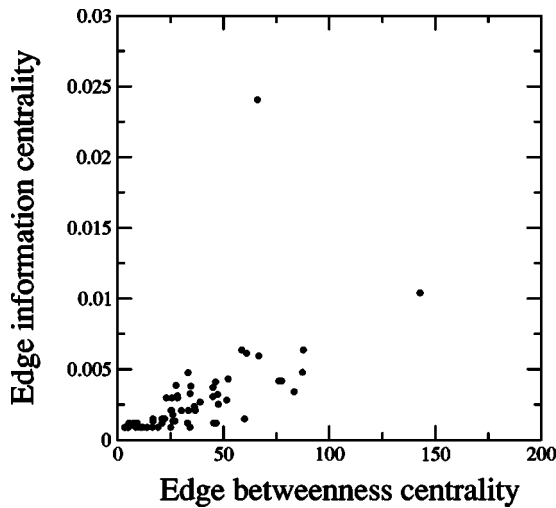


FIG. 7. Correlation between edge information centrality and betweenness centrality for the karate club network. Each point of the scatter plot refers to an edge of the network.

and 27) and two larger groups which are homogeneous except node 10 which is misclassified (curiously enough, this node is also misclassified by the fast algorithm of Newman [23]). The separation of the four above mentioned clusters corresponds to a peak in the plot of Q . However there is a second higher peak which is obtained for a split of the network into seven communities. This double peak structure is present as well in the Q -plot of the Girvan-Newman analysis [17,18].

As for the computer generated networks, we report in Fig. 9 the information centrality C^I of the edge removed, the global efficiency E , the number of components n of the resulting graph and the value of Q as a function of the number of edges removed from the network of the karate club. The figure is analogous to Fig. 2. We observe again a correlation between the peaks of C^I and the jumps of Q (here we have two). Moreover, like in the previous case, the absolute maximum of Q corresponds to the lower of the two peaks of C^I .

We remind that the variation of the efficiency corresponding to the remotion of one edge is calculated by taking into account the structure of the network at the current stage, i.e., without considering the edges which were eliminated in the previous steps. For the algorithm of Girvan and Newman this condition of recalculation turns out to be crucial, because removing the edges according to the (decreasing) values of

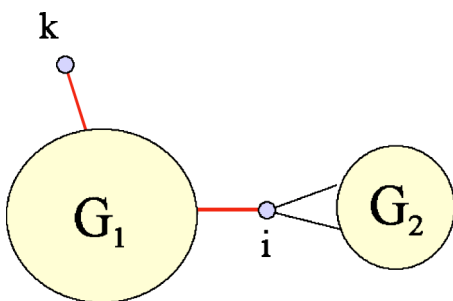


FIG. 8. A graph G composed by a node k and two cohesive subgroups, G_1 and G_2 , connected by node i .

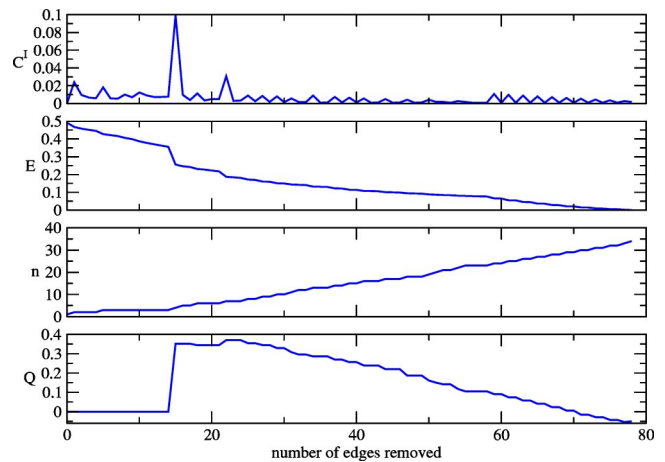


FIG. 9. Information centrality C^I of the edge removed, global efficiency E , number of components n and value of Q for the resulting graph as a function of the number of edges removed for the karate club network.

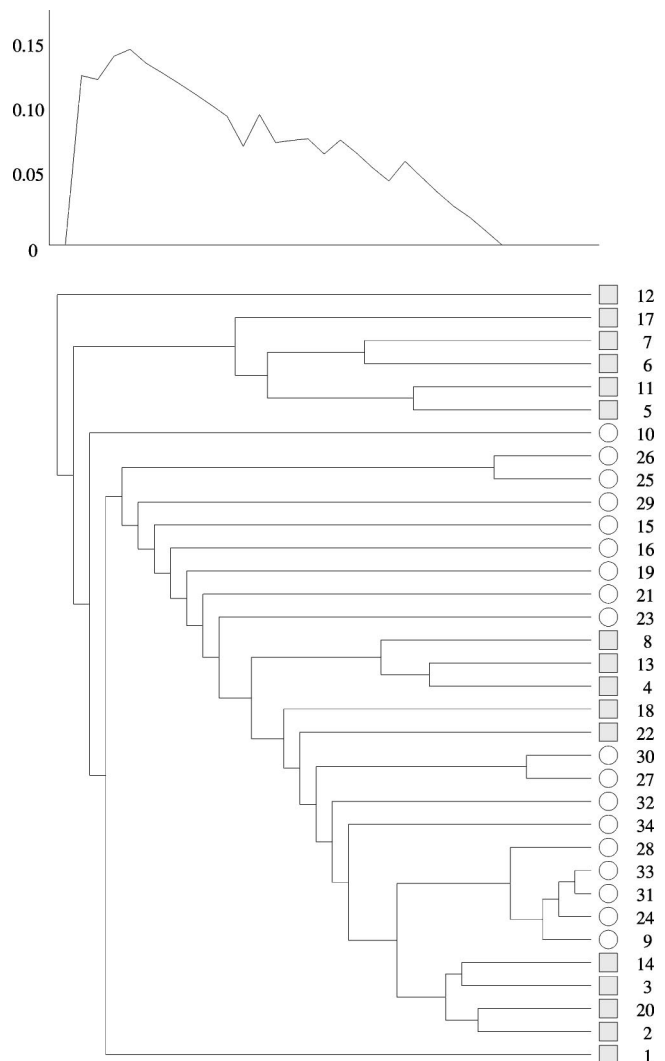


FIG. 10. Dendrogram of the communities of the karate club obtained by our method if we calculate the information centrality according to the initial structure of the network. This version of the algorithm fails to detect the communities.

- ◇ SEC
- ▽ Mid American
- Big 12
- Atlantic Coast
- ◇ Sunbelt
- ▼ Mountain West
- ▲ Conference USA
- Big East
- Western Athletic
- ◆ Pacific 10
- △ IA Independents
- Big 10

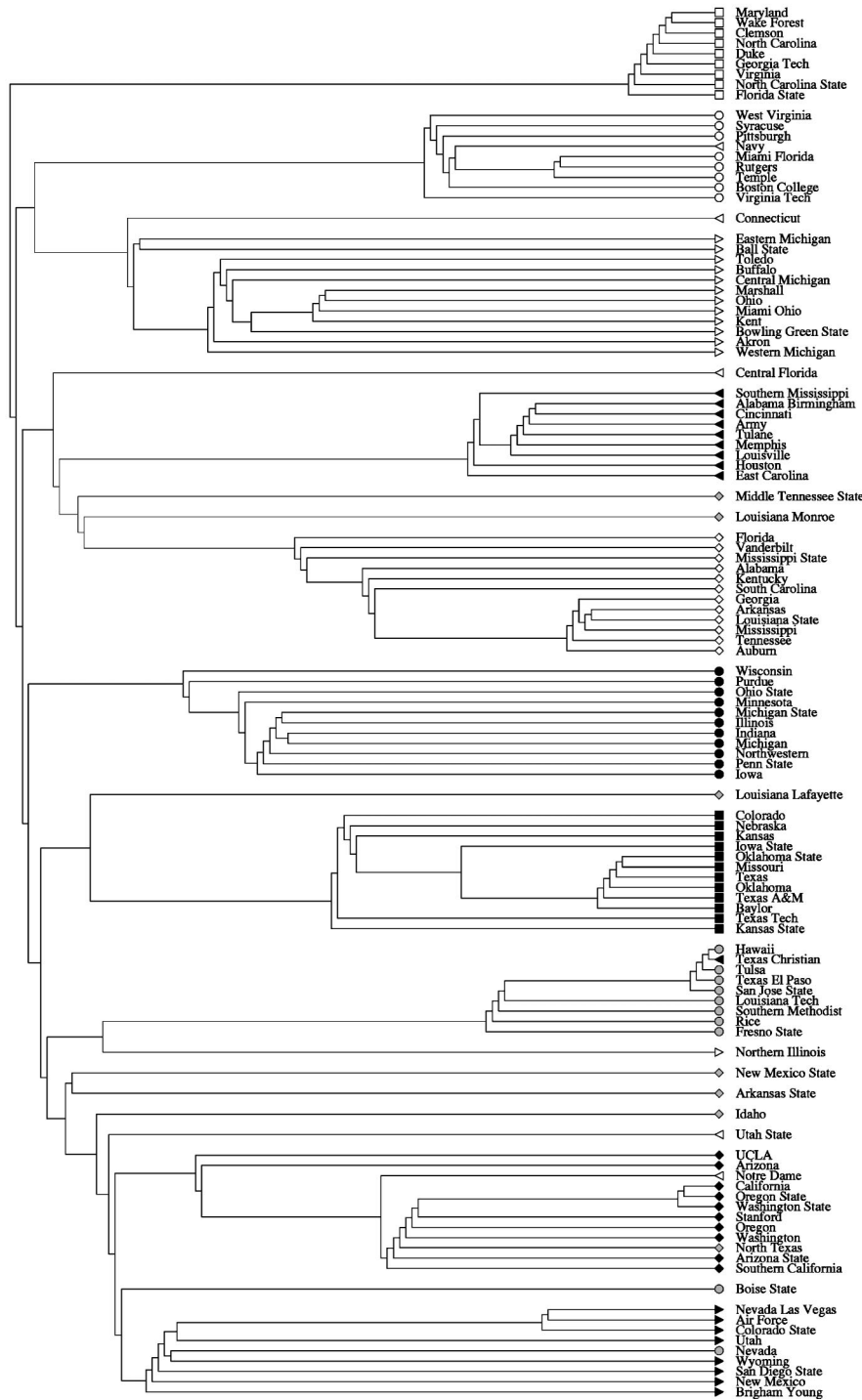


FIG. 11. Dendrogram of the communities found in the college football network. At some stage we get to the highlighted structure, which shows a split into ten main groups and few isolated nodes. The ten groups coincide, up to eventually a team, with ten of the conferences listed above the figure.

the betweenness as calculated from the original configuration of the network leads to very poor results. We wanted to check whether this is also true for our method. Indeed, Fig. 10 clearly shows that this is the case: the dendrogram does not reveal the real splitting of the network into the two classes, which instead look quite mixed up, and the modularity, whose values are quite low all over, presents a rather flat profile.

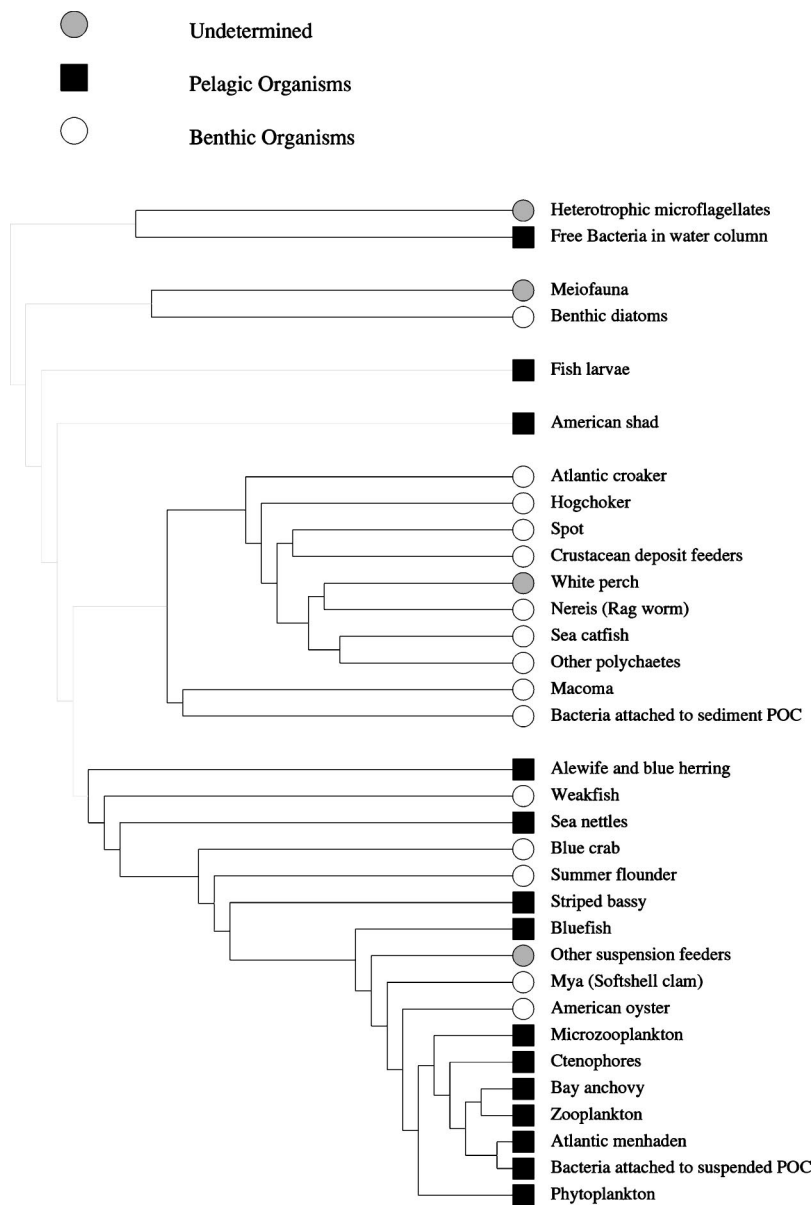


FIG. 12. Dendrogram of the communities of the Chesapeake Bay food web. The modularity peaks for the highlighted partition of the network. The two largest clusters are quite homogeneous, reflecting approximately the division between pelagic and benthic organisms.

B. Network of the American college football teams

The second network we have investigated is the college football network, representing the schedule of games between American college football teams in a season. The teams are divided into well known “conferences,” which are the communities, with a higher number of games between members of the same conference than between teams of different conferences. There are altogether eleven conferences plus few other teams which do not belong to any conference. Figure 11 shows the dendrogram we have derived with our method. The pattern of the modularity looks similar to the one we have shown for the karate club, and it again presents two peaks, the higher of which reaches the value $Q=0.485$. The corresponding subdivision of the network is the one we highlighted in the figure. We identify ten groups which coincide with ten conferences (either exactly or up to a team). The teams labeled as Sunbelt are not recognized as belonging to the same group. This group is misclassified as well in

the analysis of Girvan and Newman. There is however a reason for that, namely the fact that the Sunbelt teams played basically the same number of games against Western Athletic teams as they did among themselves. The independent teams (labeled as IA Independent) show indeed no relationship to each other nor to a particular conference and they appear as truly independent nodes.

C. Food webs

We have also applied our algorithm to several food webs. Here we mainly discuss the analysis of the food web of marine organisms living in the Chesapeake Bay, which is situated on the Atlantic coast of the United States. This special ecosystem was originally studied by Baird and Ulanowicz [29], who carefully investigated the trophic relationships (i.e., the predatory interactions) between the 33 most important taxa, which are the vertices of the network; a taxon is a species or a group of species. Baird and Ulanowicz studied

the exchanges of carbon among the taxa and in this way they compiled the matrix of their trophic relationships, specifying the percentage of carbon assimilated in each interaction. So, strictly speaking, the network is directed (A feeds on B but the opposite is not true), and valued (due to the different percentages of carbon exchanged in the interactions); nevertheless, we took it nondirected and nonvalued, following Girvan and Newman [17]. Figure 12 shows our analysis, which is quite similar to the analysis of Ref. [17]; the optimal split (peak of the modularity) is obtained for a separation in two large classes and four small ones. One of the big groups, with a few exceptions, contains pelagic organisms (which live near the surface or at middle depths), the other one mainly benthic organisms (which live near the bottom). Our classification of the taxa thus favours the habitat versus the trophic levels, in contrast to other methods used to study food webs. We must be careful, however. On the one hand, the mixed pattern of Fig. 12 suggests that one should probably take into account other criteria as well. On the other hand our analysis of a similar food web, relative to the sea-grass ecosystem of St. Marks National Wildlife Refuge [30], shows different results. This network is larger than the previous one (48 vertices versus 33) and has several species in common with the ecosystem of the Chesapeake Bay. The presence of terrestrial species and birds enlarges the variety of possible habitats and the spectrum of the trophic levels; the latter allowed us to identify in [30] five clusters of taxa. Nevertheless, our study did not reveal any particular subdivision of the species. Repeating the analysis with the algorithm of Girvan and Newman led essentially to the same results. We had similar problems by analyzing other food webs; the reason may be the fact that these networks often contain many edges, and our algorithm is probably not suitable for the analysis of dense graphs.

D. Primate network

In this section we consider a data set collected by Linda Wolfe [9,31], recording 3 months of interactions amongst a group of 20 monkeys, where interactions were defined as the joint presence at the river. The dataset also contains information on the sex and the age of each animal. Monkeys 1–5 are males; monkeys 6–20 are females. In increasing order of age: monkeys 7, 14, 18, 20 belong to the first age group (the youngest), monkeys 4, 5, 9, 10, 15, 17 to the second, monkeys 2, 3, 8, 12, 16 to the third and monkeys 1, 6, 11, 13, 19 to the fourth and oldest group. A detailed analysis of the individual and group centrality of this network can be found in Refs. [9,31]. The total number of links is 31, i.e., of the order of magnitude of the nodes. Indeed, six out of twenty monkeys did not actively participate in the social life of the group; the resulting nondirected nonvalued graph thus consists of 6 isolated points (labeled by the numbers 2, 6, 16, 18, 19, 20) and a connected component of 14 points. The results of our analysis are illustrated in Fig. 13, where we reported as well for each primate both sex (M =male, F =female) and age (in years). The modularity of the subsequent subdivisions of the network in components is very low, which shows

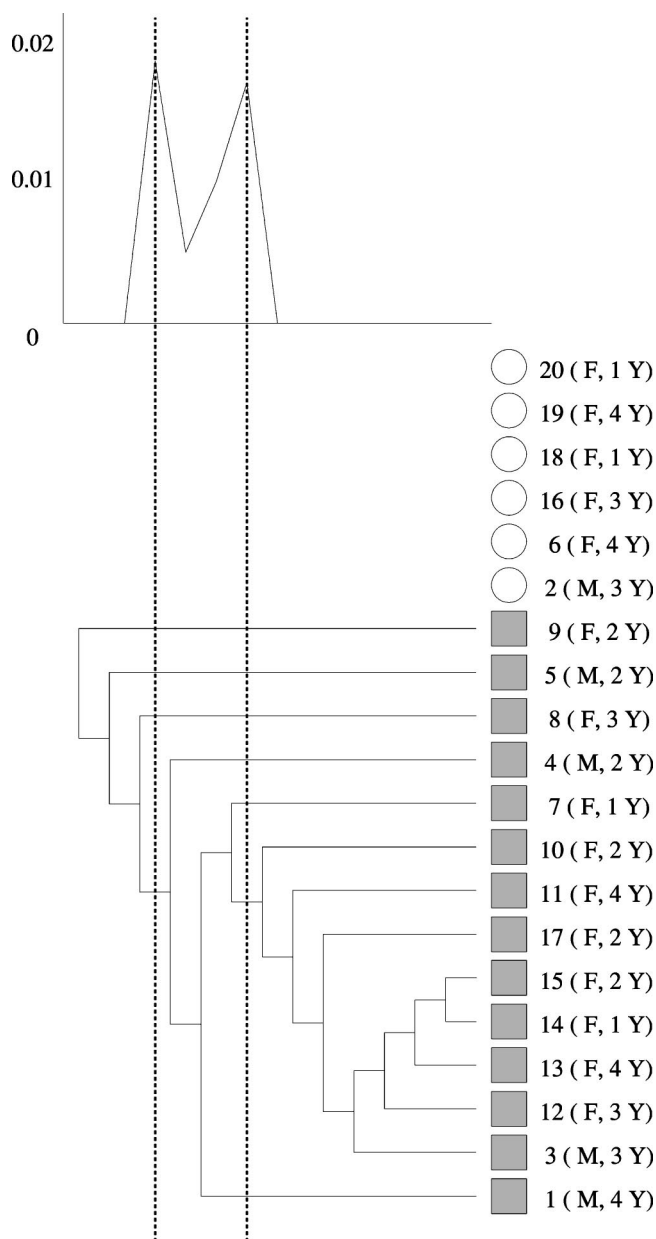


FIG. 13. Dendrogram of the primate network. The circles represent the asocial monkeys, the squares the social monkeys (see text). There is no separation in classes; our procedure leads to a progressive isolation of the nodes. The modularity Q is very low, the higher peak is relative to a partition in a large group and the isolated nodes 5, 8 and 9 (besides the asocial primates).

that there is no appreciable community structure; nevertheless, two peaks are clearly visible, the higher of which is obtained when the nodes 5, 8 and 9 separate one after the other from the network. One gets then a major community of eleven elements and nine isolated monkeys. We do not find any sensible relationships between our partition and the division of the primates in age groups. We analyzed the network as well with the method of Girvan and Newman and the results are essentially the same: one gets again two peaks for the modularity (whose values remain low) and the best partition of the network corresponds to a separation in the

same large community we found before without node 11, which is now isolated, plus isolated sites except the pair 5-8.

VI. CONCLUSIONS

We have presented a new algorithm to identify the subdivisions of complex networks in cohesive groups of vertices, or communities. The algorithm is based on a recently introduced centrality measure, the so-called information centrality, and consists in classifying all edges according to the value of this measure, so to determine which edge is most central: the latter edge is then removed from the network. One then recalculates the information centrality of the remaining edges and again removes the most central edge; the procedure is repeated until all edges are removed. The hope is that this sequential removal of edges loosens the bonds between tightly connected groups of vertices, so that, at some stage, they eventually separate from each other.

For the quantitative evaluation of the goodness of the successive splits, which is necessary in order to identify the best subdivision of the network, we adopted the modularity Q introduced in [18]. Our algorithm runs to completion in time $O(K^3N)$ (K and N are the number of edges and vertices of the graph, respectively) and therefore is not so fast as other methods; because of that, networks with thousands of vertices are unreachable. The aim of the paper, however, was to check whether the information centrality is relevant in the search of the communities.

The results of the application of our method both to computer generated networks and to real networks clearly show that the algorithm is indeed able to detect the real communi-

ties in most cases. This implies the existence of a correlation between the information centrality C_I^k of an edge k and the fact that the edge joins two different communities; the higher C_I^k , the more likely k is a tie between groups. This is confirmed by the correlation we observed between the peaks of C_I^k and the jumps in the modularity (see Figs. 2 and 9). We stressed the importance of the recalculation of the information centrality step by step; without it the algorithm is not able to distinguish the communities. Our method was especially devised for sparse graphs (i.e., when $K \sim N$), and it is probably doomed to fail for dense graphs ($K \sim N^2$).

The examples we have taken allowed us as well to see how efficient our algorithm is compared with others. In particular we made extensive comparisons with the algorithm of Girvan and Newman [17,18], which also uses a centrality measure, the edge betweenness. It turns out that our algorithm is generally as good as the one of Girvan and Newman. It seems to perform slightly better when there is a high degree of mixture between the classes; on the other hand, it sometimes has troubles with nodes which are too loosely bound to the rest of the network (like nodes with a single edge), which may separate too early and be misclassified, although they often happen to be truly independent communities.

ACKNOWLEDGMENTS

We thank B. A. Huberman for providing us with the network of the college football teams and G. Caldarelli for useful information on the food webs. S.F. acknowledges the support of the DFG Forschergruppe under grant FOR 339/2-1.

-
- [1] S. Wasserman and K. Faust, *Social Networks Analysis* (Cambridge University Press, Cambridge, England, 1994).
 - [2] J. Scott, *Social Network Analysis: A Handbook*, 2nd ed. (Sage, London, 2000).
 - [3] S. H. Strogatz, *Nature (London)* **410**, 268 (2001).
 - [4] S. N. Dorogovtsev and J. F. F. Mendes, *Evolution of Networks* (Oxford University Press, Oxford, 2003).
 - [5] R. Albert and A. L. Barabási, *Rev. Mod. Phys.* **74**, 47 (2002).
 - [6] P. Holme, M. Huss, and H. Jeong, *Bioinformatics* **19**, 532 (2003).
 - [7] D. Wilkinson and B. A. Huberman, *Proc. Natl. Acad. Sci. U.S.A.* **10**, 1073 (2004).
 - [8] A. Vespignani, *Nat. Genet.* **35**, 118 (2003).
 - [9] V. Latora and M. Marchiori, e-print cond-mat/0402050.
 - [10] V. Latora and M. Marchiori (unpublished).
 - [11] V. Latora and M. Marchiori, *Phys. Rev. Lett.* **87**, 198701 (2001).
 - [12] V. Latora and M. Marchiori, *Eur. Phys. J. B* **32**, 249 (2003).
 - [13] G. Bonanno, S. Fortunato, V. Latora, F. Lillo, and R. N. Mantegna (in preparation).
 - [14] R. D. Alba, *J. Math. Sociol.* **3**, 113 (1973).
 - [15] S. B. Seidman, *Soc. Networks* **5**, 97 (1983).
 - [16] S. P. Borgatti, M. G. Everett, and P. R. Shirey, *Soc. Networks* **12**, 337 (1990).
 - [17] M. Girvan and M. E. J. Newman, *Proc. Natl. Acad. Sci. U.S.A.* **99**, 7821 (2002).
 - [18] M. E. J. Newman and M. Girvan, e-print cond-mat/0308217.
 - [19] Newman and Girvan in Ref. [18] have also introduced a third measure of edge betweenness, the current-flow betweenness, defined by considering an electric circuit associated to the graph and by using Kirchoff's laws. Despite the different definition they have proven that the current-flow betweenness coincides with the random-walk betweenness.
 - [20] C. L. Freeman, *Sociometry* **40**, 35 (1977); *Soc. Networks* **1**, 215 (1979).
 - [21] H. Zhou, *Phys. Rev. E* **67**, 041908 (2003); **67**, 061901 (2003).
 - [22] R. D. Bock and S. Z. Husain, *Sociometry* **13**, 146 (1950).
 - [23] M. E. J. Newman, e-print cond-mat/0309508.
 - [24] M. E. J. Newman, *Phys. Rev. E* **64**, 016132 (2001).
 - [25] U. Brandes, *J. Math. Sociol.* **25**, 163 (2001).
 - [26] F. Wu and B. A. Huberman, e-print cond-mat/0310600.
 - [27] W. W. Zachary, *J. Anthropological Res.* **33**, 452 (1977).

- [28] Newman and Girvan consider all edges in the original network, even after edges have been removed by the community structure algorithm the modularity is calculated by using the full network.
- [29] D. Baird and R. E. Ulanowicz, *Ecol. Monogr.* **59**, 329 (1989).
- [30] R. R. Christian and J. Luczkovich, *Ecol. Modell.* **117**, 99 (1999).
- [31] M. G. Everett and S. P. Borgatti, *J. Math. Sociol.* **23**, 181 (1999).