Community detection by signaling on complex networks

Yanqing Hu, Menghui Li, Peng Zhang, Ying Fan, and Zengru Di[*]
Department of Systems Science, School of Management, Center for Complexity Research, Beijing Normal University, Beijing 100875, China

(Received 24 November 2007; revised manuscript received 23 April 2008; published 30 July 2008)

Based on a signaling process of complex networks, a method for identification of community structure is proposed. For a network with n nodes, every node is assumed to be a system which can send, receive, and record signals. Each node is taken as the initial signal source to excite the whole network one time. Then the source node is associated with an n-dimensional vector which records the effects of the signaling process. By this process, the topological relationship of nodes on the network could be transferred into a geometrical structure of vectors in n-dimensional Euclidean space. Then the best partition of groups is determined by F statistics and the final community structure is given by the K-means clustering method. This method can detect community structure both in unweighted and weighted networks. It has been applied to *ad hoc* networks and some real networks such as the Zachary karate club network and football team network. The results indicate that the algorithm based on the signaling process works well.

DOI: 10.1103/PhysRevE.78.016115 PACS number(s): 89.75.Hc, 89.75.Fb, 89.65.−s

I. INTRODUCTION

The study of complex networks has been paid an enormous amount of attention [1–3] by the scientific community in recent years. The topologies of a wide variety of systems are studied, such as the World Wide Web [4], social and communication networks [5,6], biochemical networks, and so on [7]. One interesting problem is detecting the community structure of networks. Communities or modules within networks can loosely be defined as subsets of nodes which are more densely linked with each other, while compared to the rest of the network [8,9]. Such communities have been observed in many different contexts, including metabolic networks, banking networks, and most notably social networks. As a result, the identification of communities has been the focus of many recent efforts. Community detection in large networks is potentially very useful, because nodes belonging to a tight-knit community are more likely to have some properties in common. What is more, these communities may probably be functional groups, which provide us valuable reference to our study in many other fields. In recent studies, many different algorithms have been proposed [8–24] (see [11] for a review) to detect the community structures. These algorithms can be divided into three categories. Some algorithms are designed according to maximal modularity Q. Some are designed based on topology structures (betweenness, degree, or clustering coefficient). And the others are designed according to the dynamical properties of the network.

Modularity Q is an index advanced by Newman and Grivan [25] as a measurement for the community structure. It gives a clear and precise definition of the characteristics of the acknowledged community and has had very successful application in practice. So it leads us to many other algorithms brought forward to divide a community by maximizing the modularity Q. Unfortunately, maximizing the modularity Q has been proven to be a nondeterministic polynomial time (NP)-complete problem [26], which makes it unable to work out the partition corresponding to maximal Q in a large network. Actually many algorithms for maximizing Q are usually heuristic algorithm. Besides, with respect to the modularity Q, it has been strictly proven that, as an index to measure the community structure, it tends to combine the small communities rather than identify them successfully in networks with definite communities [27]. Though the modularity Q has been proven to have the above-mentioned inherent defects, it is still a successful index to measure a network for the moment. Therefore, lots of works for detecting communities are dependent on the index Q.

The Grivan-Newman (GN) algorithm [9] and spectral analysis method are two algorithms based on network topology. The GN algorithm was proposed by Grivan and Newman. It first gets the dendrogram concerning the network structure by removing links with the largest betweenness. Then with the help of the modularity Q or other indices, the best partition of the network can be obtained. The principle of the spectral analysis method [29] is based on the theory of the eigenvector of the matrix. Relatively speaking, the spectral analysis method is the most mathematical-theory-based approach. It needs also some methods to determine the best partition, by using Q, ascertaining the sizes of the two subnetworks by the sign of the elements of Fielder's eigenvector and so on.

There are still other algorithms based on the dynamics of networks, among which the random walk [14] method and circuit approach method [17] will be briefly discussed here. For the random walk method, each node contains a walker initially. Then each walker will randomly choose a neighbor of the node it currently stands on to localize. It is a Markov process. After a proper period of time, the walker has a probability to reach any other nodes. Based on this possibility, a dendrogram of the network can be obtained. Then partition can be made by the aid of the modularity Q. When using the random walk method to detect communities, it is difficult to specify the optimum random-walking time. And the best partition is usually dependent on the index Q. The principle of the circuit approach method is to regard the edges of the

[*] Author to whom correspondence should be addressed. zdi@bnu.edu.cn

network as the resistances and add voltage to adequate nodes of the network, then work out the voltage of each node by Kirchhoff's law. Nodes with similar voltages are regarded to exist in the same community more probably. At the same time, it defines the index such as tolerance to realize the partition of the network.

In this paper, we propose an algorithm for identification communities based on the signaling process of a network. In this approach, every node is viewed as an excitable system. It can send, receive, and record signals. Initially, a node is selected as the source of signal. An initial signal is set on the source node, while other nodes are without any signals. Then the source node sends a signal to its neighbors and itself first. Afterwards, the nodes with signals can also send signals to their neighbors and themselves. In this signaling process, we require that the node record the amount of signals it has received, and at every time step, each node sends all its present-owning signals to its neighbors and itself. After a certain T time steps, the amount distribution of signals over the nodes could be viewed as the influence of the source node on the whole network. For a network with n nodes, the signal distribution can be characterized by an n-dimensional vector.

If a network has n nodes, we can obtain the influence of every node by the same operation. The results are given by n n-dimensional vectors. Generally speaking, the source node should influence its own community first and then influence the whole network by spreading among different communities. So naturally, compared with nodes in other communities, the nodes of the same community have more similar influence on the whole network. And the difference of influence could be given by the n-dimensional vectors.

Thus, by the above signaling process on networks, the topological structure of nodes is converted into the geometrical relationships of vectors in n-dimensional Euclidean space. We can obtain the community structures of nodes by clustering these n-dimensional vectors. Actually, there are already many methods to cluster vectors in Euclidean space. Here we chose the K-means clustering (KMC) method assisted by F statistics [31] to get the best partition of the communities. F statistics describes the best partition as having a shorter average distance between vectors inside the same community and a larger distance between vectors of different communities. After getting the best number of groups by F statistics, we can detect the communities by the KMC method.

Some problems related to the above method are also discussed, including the optimum time steps T of inspiration and the generalization of the method to weighted networks. Then we applied the method to detect the communities in *ad hoc* and some real networks. Its precision and accuracy are obtained and compared with some other algorithms. The results indicate that the method based on the signaling process performs well.

II. METHOD BASED ON THE SIGNALING PROCESS

A. Basic algorithm

1. Signaling process

For a network with n nodes, every node is assumed to be a system which can send, receive, and record signals. A node can only affect its neighbors, which will affect their neighbors in the same way. Finally, each node will affect the whole network. In general, one node will affect its community first and then the whole network via its community. So we can safely conclude that the nodes in the same community will affect the whole network in a similar way.

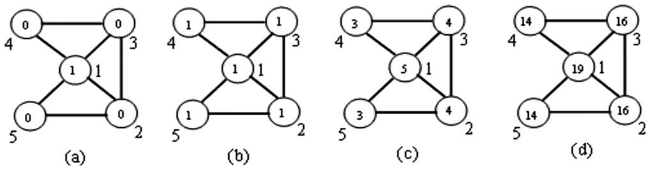

FIG. 1. Sketch of the signaling process. (a) We set a signal on node 1 and other nodes have no signal initially. (b) In the first step node 1 sends one signal to all its neighbors which are nodes 2, 3, 4, and 5 and itself. Then all nodes have one signal. (c) Next, each of them will send one signal to their neighbors and themselves, respectively, at the same time. After the second step, node 1 has five signals, both nodes 2 and 3 have four signals and nodes 4 and 5 have three signals. The vector [5,4,4,3,3] represents the effect of node 1 to the whole network in two steps. (d) Then every nodes send the same amount of signals as they received in the last step to their neighbors and themselves.

At the beginning, we select a node as the source and give it one unit of signal, but the other nodes have no signal. Then let the source node send a signal to all of its neighbors and itself. After the first step, the node and all its neighbors have a signal. In the second step, all the nodes with a signal will send the signals to their neighbors and themselves. Every node can record the amount of signals it received, and then it will send the same quantity of signals in the next time step. In this way, the process will be repeated continuously on the network. After T time steps, we can get an n-dimensional vector that records each node's signal quantity which represents the effect of the source node. The signaling process is sketched out in Fig. 1 by a simple network with five nodes. Choosing every node as the source node, respectively, we can get n such vectors. The reason that we let each node send a signal or signals to itself is to take account of the historical effects. This has been proven to be helpful to distinguish the amounts of signals between the nodes in the community and outside in a relatively short time period. Normalizing the n vectors, then the distance of each pair of vectors will represent the similarity of the corresponding nodes. Using this kind of similarity the communities can be detected.

Actually, the above signaling process could be described by a simple but clear mathematical mechanism. Suppose we have a network with n nodes; it can be represented mathematically by an adjacency matrix $\mathbf{A}$ with elements A_{ij} if there is an edge from i to j and 0 otherwise. Then the column i of matrix $\mathbf{V}=(\mathbf{I}+\mathbf{A})^T$ will represent the effect of source node i to the whole network in T steps. In order to get the relative effect, we should normalize each column of matrix $\mathbf{V}$. Assume the column i of $\mathbf{V}$ is $V_i=(v_{i1},v_{i2},\ldots,v_{in})$; then, the V_i can be normalized as $U_i=(u_{i1},u_{i2},\ldots,u_{in})$, here, $u_{ij}=\frac{v_{ij}}{\sum_j^n v_{ij}}$. Then to partition the network with n nodes is equivalent to a cluster of n vectors $U_1,U_2,\ldots,U_n$ in Euclidean space.

2. K-means clustering

It is well known that there are many clustering methods and algorithms for vectors in Euclidean space. In this paper, we choose the inexpensive KMC algorithm [31] to detect communities for the vectors given by the signaling process. The KMC algorithm is described as follows.

(i) Set K as the number of communities to partition.

(ii) Choose proper K vectors for the K communities as their centroids.

(iii) Randomly choose a vector. The vector will belong to the community when the distance between the vector and the centroid of the community is minimum among all the centroid of communities.

(iv) Recompute the communities' centroids which have added a vector or deleted a vector.

(v) Repeat step 3 to step 4 until all the centroids cannot be modified.

We know that there are many kinds of definitions for distance. In our algorithm we choose the Euclidean distance to measure the similarity between vectors of nodes. How to find the proper K centroids? We can randomly choose a vector as the first centroid. Then we choose a vector with the largest distance to the first centroid as the second one. In the same way, at the tth step, we always choose a vector with the largest sum of distance from it to all the $t-1$ centroids as a new centroid until we get K centroids.

3. F statistics

At the first step of the K-means clustering algorithm, we must set an extra parameter K which presents how many clusters we will partition. Here we use F statistics [31] to estimate a proper K. Suppose $U=\{u_1,u_2,\ldots,u_n\}$ is the set of vectors of all nodes and $u_j=(x_{j1},x_{j2},\ldots,x_{jn})$; here, x_{jk} is the kth character quantity of u_j. Suppose K is the number of communities and n_i is the number of nodes of the ith community. We name all the nodes' vectors of the ith community as $u_1^i,u_2^i,\ldots,u_{n_i}^i$. Let $\bar{x}_k^i=\frac{1}{n_i}\sum_{j=1}^{n_i}u_j^i(k)$, $k=1,2,\ldots,n$, be the mean characters of ith community, $\bar{u}^i=(\bar{x}_1^i,\bar{x}_2^i,\ldots,\bar{x}_n^i)$ be the ith community's centroid, and $\bar{u}=(\bar{x}_1,\bar{x}_2,\ldots,\bar{x}_n)$ be all the nodes' centroid; here, $\bar{x}_k=\frac{1}{n}\sum_{j=1}^{n}x_{jk}$ $(k=1,2,\ldots,n)$. Then F statistics is defined as

$$F=\frac{\sum_{i=1}^{K}\frac{n_i\|\bar{u}^i-\bar{u}\|^2}{K-1}}{\sum_{i=1}^{K}\sum_{j=1}^{n_i}\frac{\|u_j^i-\bar{u}^i\|^2}{n-K}}, \qquad (1)$$

where $\|\bar{u}^i-\bar{u}\|=\sqrt{\sum_{k=1}^{n}(\bar{x}_k^i-\bar{x}_k)^2}$ is the distance between $\bar{u}^i$ and $\bar{u}$, and $\|u_j^i-\bar{u}^i\|$ is the distance between the u_j^i node of the ith and the centroid $\bar{u}^i$. The numerator of F signifies the distance of intercommunities and the denominator the distance of intracommunities. So the F could be larger when the difference distance of intercommunities is larger and the difference distance of intracommunities is smaller. When F achieves the maximum, we can get the best partition.

For a binary *ad hoc* network which contains 128 nodes and 4 groups, we proceed with the signaling process as above to test the F statistics. The results show that F statistics is very efficacious. On weighted *ad hoc* networks, the results are similar to the binary ones. The clearer the community structure is, the more distinct the maximal value of the F statistics. The results are shown in Fig. 2.

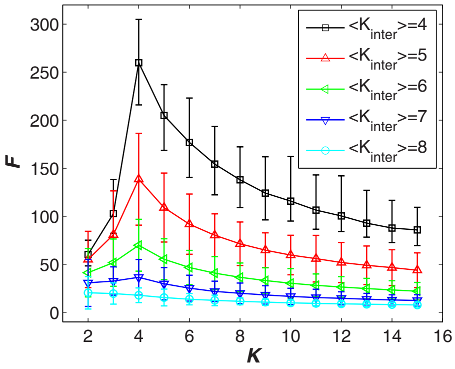

FIG. 2. (Color online) F statistics as a function of the number of clusters K. The plot shows the changes of F statistics when $T=3$ and $\langle k_{inter}\rangle$ changes from 4 to 8. It shows that when $\langle k_{inter}\rangle$ is smaller than 8, F statistics can identify the right number of communities. When the community structure is clearer, the maximal value of the F statistics is very distinct. The results are an average of 20 realizations of networks. Error bars show the differences among networks.

B. Some related problems

1. Most optimal T

The parameter T is an important factor for the results of community identification. We can image that the length T must be adequate to gather enough information about the topology of the network, but it should not be too long to overshadow the information we have gathered. In order to let the majority of nodes affect the whole network and not to overshadow the information about the topology of the network, we guess that it may be optimal when T is near to the average shortest path of the network. In order to verify it, some numerical experiments are done on binary networks which contain 128 nodes and 4 groups, the same as above. The results are shown in Fig. 3. The accuracy of the algorithm reaches optimum when T is 3 or 4. It seems that our guess is provable. Of course, we only do some numerical experiments on artificial networks. It is hard to say the rule satisfies all kinds of networks. The random walk method [14] has also encountered the same problem. How to find the optimal T? We think it is still an open question.

2. Time complexity analysis

The time complexity of our algorithm can be analyzed as follows. For a definite K which is the number of communities, the time complexity for K-means clustering is $O(Kn^2)$. The time complexity of the process of signal diffusion is $O(Tn^3)$ when we use multiplication of the matrix to simulate the process. But if we simulate the process in a network directly, the corresponding time complexity is $O(T(k+1)n^2)$, where k is the average degree of nodes in a network. If we do

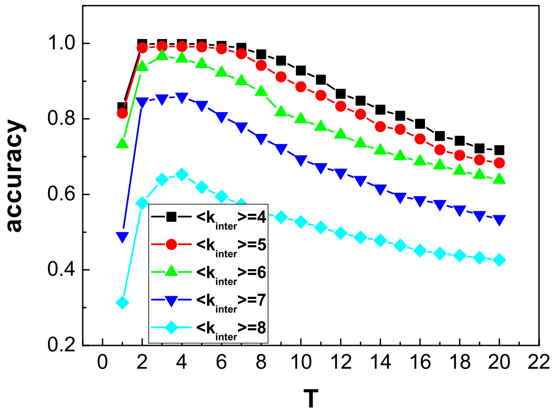

FIG. 3. (Color online) Let $K=4$ in the algorithm always. The plot shows the changes of accuracy measured by the similarity function S when T changes from 1 to 20 on binary *ad hoc* networks. From the plots we can see that $T\approx 3$ is the proper length. And here 3 is near the average shortest paths of each artificial network.

not have any information about the community number, the community number K should be tested from 2 to n; thus, the time complexity should be $O(n^4)$. But if we know the range of K, which is independent of n, the time complexity will decrease sharply.

3. Generalization to weighted networks

It is easy to generalize our algorithm to a weighted network. Suppose we have a weighted network with n nodes; it can be represented mathematically by an adjacency matrix $\mathbf{W}$ with elements W_{ij}. W_{ij} denotes the connection strength of node i and j (in some weighted networks, W_{ij} does not denote the strength of connection, we should transform the weight before the algorithm); then, $\mathbf{V}=(\mathbf{I}+\mathbf{W})^T$. The rest of the algorithm is the same as the algorithm on a binary *ad hoc* network.

4. Relation to other methods

There two points where our method differs from the random walk method [14] and circuit approach method [17]. First, we use the signal diffusion process to transfer the topology into geometrical structure. The mathematical form is $(\mathbf{I}+\mathbf{A})^T$. The distance of each pair of column vectors of the matrix is the intimacy of the corresponding pair of nodes. The random walk method obtains the intimacy of each pair of nodes by random walks. The mathematical form is $(\text{diag}(\frac{1}{d_1}\frac{1}{d_2}\cdots\frac{1}{d_n})\mathbf{A})^T$ where d_i denote the degree of node i, and "diag" means the diagonal matrix. Taking account of the effect of node degree, it also uses the Euclidean distance to define the intimacy. The circuit approach method obtains the intimacy of each pair of nodes by Kirchhoff's law. Adding the potential difference to the proper two nodes, by Kirchhoff's law, we can obtain the potential of each node. The closer of the potentials of two nodes are, the more intimate the two nodes. Suppose we add the potential difference of nodes 1 and 2; then, their potentials are $p_1=1$ and $p_2=0$. Let p_i denote the potential of node i. The mathematical form of Kirchhoff's law is $\mathbf{B}=(\mathbf{A}\,\text{diag}(\frac{1}{d_1}\frac{1}{d_2}\cdots\frac{1}{d_n}))$, $(p_3,p_4,\dots,p_n)'=(I-\tilde{\mathbf{B}})^{-1}C$, where $\tilde{\mathbf{B}}$ denotes the matrix $\mathbf{B}$ with eliminating the first and second columns and rows, $C=(\frac{A_{31}}{d_3},\frac{A_{41}}{d_4},\dots,\frac{A_{n1}}{d_n})'$.

Second, as to the method of clustering, we use the F statistics and classical KMC method to partition the vectors. When the F statistics achieves its maximum, we get the best partition. The random walk method and the circuit approach method also need the help of other indices to obtain the best partitions. One is the modularity Q; the other is tolerance. So we could say that the F statistics and KMC method are all based on the geometrical structure of the vector space, but the other two methods need the additional help of parameters. In the following discussion, we will compare the accuracy and precision with other famous algorithms which are not based on the dynamics of networks.

III. RESULTS AND COMPARISON WITH OTHER ALGORITHMS

In order to investigate the performance of our algorithm, the accuracy and precision of our algorithm will be compared with the Potts algorithm (Potts) [16], Girvan-Newman algorithm (GN) [25], and extremal optimal algorithm (EO) [13]. All these algorithms can be generalized to weighted networks [30]. Here we abbreviate the GN-weighted version as WGN and EO as WEO.

Accuracy means consistency when the community structure from the algorithm is compared with the presumed communities, and precision is the consistency among the community structures from different runs of an algorithm on the same network. They both need a measurement to compare two different communities. There are already several indices for this purpose. Danon *et al.* proposed a measurement $I(A,B)$ based on information theory [11]. It is based on the *confusion matrix* N, where the rows denote the presumed communities and the columns correspond to the communities found by some algorithms. The matrix element (N_{ij}) of N is the number of nodes in the presumed community i that appear in the found community j. A measure of similarity between the partitions, based on information theory, is then

$$I(A,B)=\frac{-2\sum_{i=1}^{c_A}\sum_{j=1}^{c_B}N_{ij}\ln\left(\frac{N_{ij}N}{N_{i.}N_{.j}}\right)}{\sum_{i=1}^{c_A}N_{i.}\ln\left(\frac{N_{i.}}{N}\right)+\sum_{j=1}^{c_B}N_{.j}\ln\left(\frac{N_{.j}}{N}\right)} \qquad (2)$$

where c_A is the number of presumed communities and c_B is the number of found communities, $N_{i.}$ is the sum over row i of matrix N_{ij}, and $N_{.j}$ is the sum over column j.

The information measurement mainly focus on the proportion of nodes which are correctly grouped. We proposed a similarity function S to measure the difference between partitions [30]. Starting from two community structures $\{A_1,A_2,\dots\}$ and $\{B_1,B_2,\dots\}$ over the same set N, first, we need to identify the correspondence between A's and B's by a similarity measurement. Second, for each pair of groups, the similarity of A_j and B_j is given by

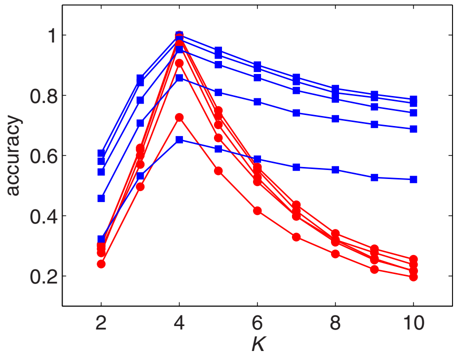

FIG. 4. (Color online) Accuracy as the function of K for binary *ad hoc* networks with 128 nodes and 4 presumed groups. Lines with squares are measured by the information index $I(A,B)$ and lines with circles are given by the similarity function S. $T=3$ in the algorithm and $\langle k_{inter}\rangle$ changes from 4 to 8 for lines from top to bottom.

$$s_j = \frac{|A_j \cap B_j|}{|A_j \cup B_j|} \tag{3}$$

and the total similarity can be calculated as

$$S = \frac{\sum_{i=1}^{K} s_i}{K}. \tag{4}$$

The similarity functions integrate the information about the proportion of the number of node coappearances in pair groups and the number of groups in A and B. It disfavors solutions with smaller or larger number of communities than target solution. In Fig. 4, the accuracies measured by both the information index $I(A,B)$ and similarity function S are given. They have similar qualitative behavior, but the similarity function S is more sensitive to the difference in number of clusters. In the following discussion, we mainly use the similarity function to measure the accuracy. In the empirical analysis for college football networks, the accuracies of different results are given by both measurements.

In the following numerical investigations of *ad hoc* networks, we first obtain 20 realizations of artificial community networks under the same conditions. Then we run each algorithm to find communities in each network 10 times. Based on these results, using the similarity function S, comparing each pair of these 10 community structures and averaging over the 20 networks (average of totally $C_2^{10}\times 20=900$ results) we could get the precision of the algorithm. Comparing each divided groups with the presumed structures, we can get the accuracy of the algorithm by averaging these $10\times 20=200$ results.

A. Results of *ad hoc* networks

1. Binary *ad hoc* networks

In order to compare our algorithm with others, we first test it on computer-generated random graphs with a well-known predetermined community structure [25]. Each graph has $N=128$ nodes divided into 4 communities of 32 nodes each. Edges between two nodes are introduced with different probabilities depending on whether the two nodes belong to the same group or not: every node has $\langle k_{intra}\rangle$ links on average to its neighbors in the same community and $\langle k_{inter}\rangle$ links to the outer world, keeping $\langle k_{intra}\rangle+\langle k_{inter}\rangle=16$. The precision of our algorithm is better than EO and almost the same as Potts and GN, while the accuracy of our algorithm is better than GN and almost the same as EO and Potts (as shown in Fig. 5).

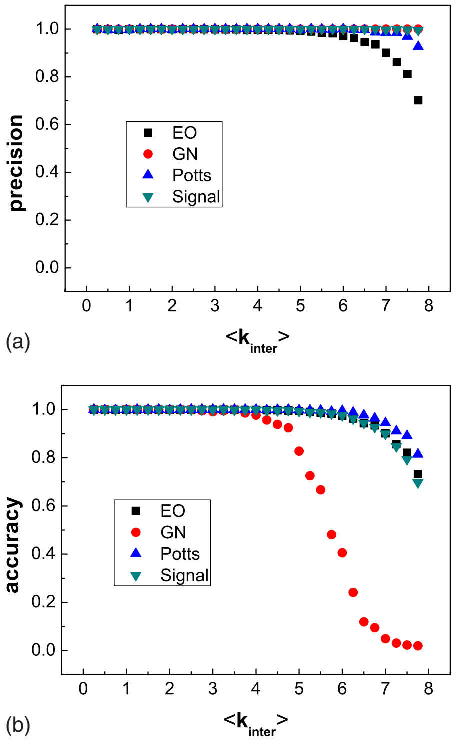

FIG. 5. (Color online) Algorithm performance as applied to *ad hoc* networks with $n=128$ and 4 communities of 32 nodes each. Total average degree is fixed to 16. The horizontal axis gives the change of $\langle k_{inter}\rangle$. We can see that the precision of our algorithm is better than EO and Potts and equal GN and the accuracy is better than GN and almost the same as EO and Potts.

2. Weighted *ad hoc* networks

In weighted networks, we use similarity link weight to describe the closeness of relations between nodes. Under the basic construction of *ad hoc* networks described above, the intragroup link weight is assigned as w_{intra}, while the intergroup link weight is assigned as w_{inter}. Similarly with $\langle k_{intra}\rangle+\langle k_{inter}\rangle=16$, we require the link weight on intra- and interlinks to follow the constraint $\langle w_{intra}\rangle+\langle w_{inter}\rangle=2$, where $\langle w_{intra}\rangle$ ($\langle w_{inter}\rangle$) is the average of all intragroup (intergroup) link weights. Here for simplicity, we assign the same weight $w_{inter}=w$ to all intergroup links and assign the same weight $w_{intra}=2-w$ to all intragroup links. From Fig. 6, we can find that the precision of our algorithm is better than WEO and Potts and equal to WGN; the accuracy of our algorithm is better than WGN, but almost equal to WEO and Potts. Even for the case with $\langle k_{inter}\rangle<\langle k_{intra}\rangle$, but $\langle w_{inter}\rangle>\langle w_{intra}\rangle$, or

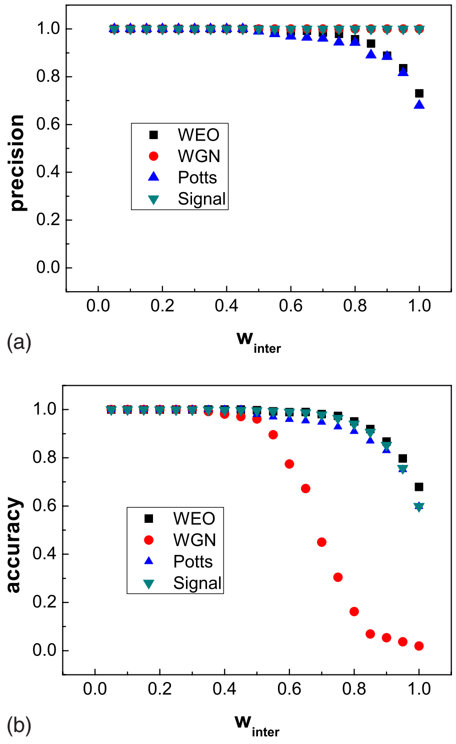

FIG. 6. (Color online) The performance of algorithms in weighted *ad hoc* networks with $n=128$ and 4 communities of 32 nodes each. $\langle k_{intra}\rangle = \langle k_{inter}\rangle = 8$, w_{inter} changes from 0.05 to 1. We can see that the precision of our algorithm is better than WEO and Potts and equal to WGN and the accuracy of our algorithm is better than WGN, but almost equal to WEO and Potts.

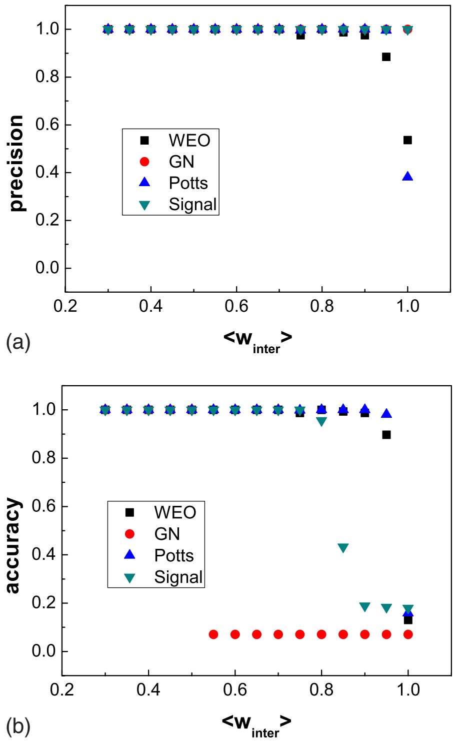

FIG. 7. (Color online) Precision and accuracy of algorithms in complete weighted networks with presumed communities. The complete network has 128 nodes and 4 communities of 32 nodes each. When $\langle w_{inter}\rangle \gg \langle w_{intra}\rangle$, the precision of our algorithm is better than WEO and Potts and equal to GN and the accuracy of our algorithm is better than GN always, but sharply declines to near 0.2 when $\langle w_{inter}\rangle$ is greater than 0.9.

with uniform distribution of link weights, we can get similar conclusions.

3. *Complete weighted networks*

An extreme idealized example is the complete network. In complete networks, we use uniform distribution of link weights. Weights are taken randomly from the interval $[\langle w_{intra}\rangle - 0.25, \langle w_{intra}\rangle + 0.25]$ and $[\langle w_{inter}\rangle - 0.25, \langle w_{inter}\rangle + 0.25]$ for intragroup and intergroup connections, respectively. The precision of our algorithm is better than WEO and Potts and equal to WGN when $\langle w_{inter}\rangle \ll \langle w_{intra}\rangle$. But its accuracy almost declines to zero when $\langle w_{inter}\rangle$ is greater than 0.9. Figure 7 shows the results in detail.

It should be mentioned that in the above discussion, the community structures in the binary or weighted *ad hoc* networks are actually determined by the topology or weight distribution of the networks. So the solution of the community should differ from the imposed structure especially when $\langle k_{inter}\rangle$ or $\langle w_{inter}\rangle$ is large. The imposed structure is not the appropriate communities at all. So when we compare communities obtained from an algorithm with the imposed structure, the drop in accuracy does not mean a failure of the method. The accuracy here cannot evaluate the performance of the algorithm, but only gives some descriptions of it.

B. Empirical results on some real networks

1. *Zachary's karate club*

The Zachary karate club network [28] has been considered as a simple sample for community-detecting methodologies [9,10,19,21,23,25]. This network was constructed with the data collected from observing 34 members of a karate club over a period of 2 years and considering friendship between members. We let $T=3$ and obtain the best partition (as shown in Fig. 8), which perfectly corresponds to the result given in Ref. [10]

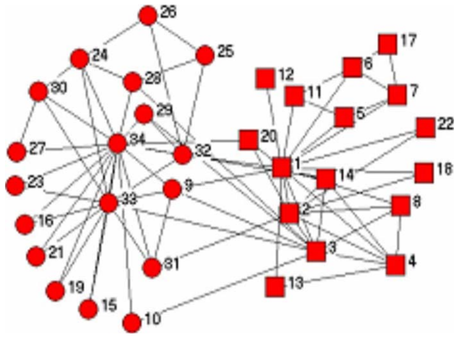

FIG. 8. (Color online) Our algorithm detects two communities from the Zachary karate club, which perfectly corresponds to the results given in [10].

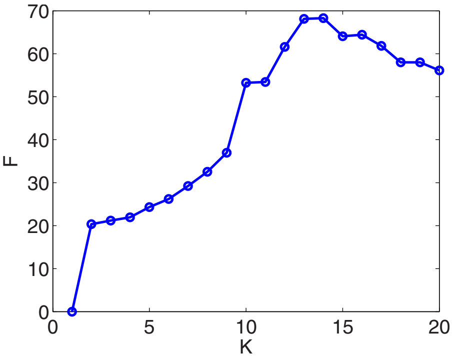

FIG. 9. (Color online) Plot shows the change of the F statistics with the number of communities when $T=2$ for college football network. F statistics achieves its maximum when the number of communities is 14.

2. College football network

The algorithm is also applied to college football network provided by Newman. The network is a representation of the schedule of Division I games for the 2000 season: nodes in the network represent teams and edges represent regular-season games between the two teams they connect. What makes this network interesting is that it incorporates a known community structure. The teams are divided into 12 conferences. Games are more frequent between members of the same conference than between members of different conferences. The average shortest path length of the football network is 2.5, so we let the signaling time $T=2$. When the F statistics achieve the maximum we get the best partition. We detect 14 communities when F reaches its maximum (Fig. 9). The accuracy measured by similarity function is 0.74, which is a little better than any of the others (Table I). We also use the information measurement $I(A,B)$ to evaluate the performance of the algorithms. The corresponding accuracy of our algorithm is 0.92, which is also better than any of the others.

IV. CONCLUSION

The investigation of community structures in complex networks is an important issue in many domains and disciplines. This problem is relevant to social tasks, biological inquiries, or technological problems. In this paper, we have introduced a method to detect communities based on the signaling process on networks.

In a complex networks with n nodes, every node is viewed as a system which can be excited. Each node sends its neighbors and itself signals and records the number of signals it receives at every time step. For each node of the network, we take it as the signal source one time. For the source node, we give it an initial unit quantity signal and other nodes have a signal of zero. Then after T steps on the network have taken place, the signal distribution of the nodes denoted by an n-dimensional vector can be viewed as the influence of the source node on the whole network. In complex networks, we can generally consider that the node always influences its community first and then the whole network. Thus, compared with nodes in other communities, nodes in the same community have a similar influence on the whole network. So through the signaling process, the network partition problem is transformed into the vector clustering problem in Euclidean space. The communities can be work out by the KMC method with the help of F statistics. Moreover, our algorithm can also be generalized to weighted networks when we think the weighted connections can magnify or dwindle the signals linearly. We have compared the precision and accuracy of our algorithm with EO, Potts, and GN algorithms. The numerical results for both *ad hoc* and

TABLE I. The accuracy of each detected community compared with the counterpart of a real-world community (measured by the similarity function). The last two rows give the accuracy of the communities measured by the similarity function (S) and information index (I), respectively.

Conference name	KMC accuracy	GN accuracy	EO accuracy	Potts accuracy
Atlantic Coast	1	0.9000	1	1
Big East	0.8000	1	0.8356	0.5600
Big10	1	1	0.9833	1
Big12	1	0.9231	1	0.9143
Conference USA	0.9000	0.9000	0.8400	0.7071
IA Independents	0.1818	0	0	0
Mid American	0.5385	0.8667	0.8852	0.8320
Mountain West	1	0	0.5143	0.3756
Pac10	1	0.5556	0.9374	0.7125
SEC	1	0.7500	0.7956	0.8100
Sunbelt	0.4444	0.4444	0.0444	0
Western Athletic	0.7273	0.7273	0.7273	0.5091
Accuracy (S)	0.7378	0.6723	0.7136	0.6184
Infor-accuracy (I)	0.9150	0.8787	0.8865	0.8601

real networks have proved that our algorithm works well, but the accuracy and precision are almost the same with the EO and Pott algorithms. Although the time complexity of our algorithm is $O(n^4)$ and it is not practically useful for large networks, it is useful for many small and important networks such as metabolic networks [32], protein networks [33], and some social networks [34].

ACKNOWLEDGMENTS

We thank Professor M. E. J. Newman very much for providing college football network data. We thank two anonymous referees for many useful suggestions. This work is partially supported by the 985 Project and NSFC under Grants No. 70771011, No. 70431002, and No. 60534080.

[1] R. Albert and A.-L. Barabasi, Rev. Mod. Phys. **74**, 47 (2002).
[2] M. E. J. Newman, SIAM Rev. **45**, 167 (2003).
[3] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, Phys. Rep. **424**, 175 (2006).
[4] R. Albert, H. Jeong, and A.-L. Barabasi, Nature (London) **401**, 130 (1999).
[5] S. Redner, Eur. Phys. J. B **4**, 131 (1998).
[6] M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **98**, 404 (2001).
[7] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabasi, Nature (London) **407**, 651 (2000).
[8] M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **103**, 8577 (2006).
[9] M. Girvan and M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2004).
[10] M. E. J. Newman, Phys. Rev. E **69**, 066133 (2004).
[11] L. Danon, J. Duch, A. Arenas, and A. Diaz-Guilera, J. Stat. Mech.: Theory Exp. (2005) P09008.
[12] S. Lehmann and L. K. Hansen, e-print arXiv:physics/0701348.
[13] J. Duch and A. Arenas, Phys. Rev. E **72**, 027104 (2005).
[14] M. Latapy and P. Pons, in *Proceedings of the 20th International Symposium on Computer and Information Sciences*, edited by P. Yolum, T. Güngör, F. Gürgen, and C. Özturan, Lecture Notes in Computer Science Vol. 3733 (Springer, New York, 2005), pp. 284–293.
[15] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, Proc. Natl. Acad. Sci. U.S.A. **101**, 2658 (2004).
[16] J. Reichardt and S. Bornholdt, Phys. Rev. Lett. **93**, 218701 (2004).
[17] F. Wu and B. A. Huberman, Eur. Phys. J. B **38**, 331 (2004).
[18] A. Clauset, Phys. Rev. E **72**, 026132 (2005).
[19] J. P. Bagrow and E. M. Bollt, Phys. Rev. E **72**, 046108 (2005).
[20] S. Muff, F. Rao, and A. Caflisch, Phys. Rev. E **72**, 056107 (2005).
[21] M. E. J. Newman and E. A. Leicht, Proc. Natl. Acad. Sci. U.S.A. **104**, 9564 (2007).
[22] C. P. Massen and J. P. K. Doye, Phys. Rev. E **71**, 046101 (2005).
[23] L. Donetti and M. A. Munoz, J. Stat. Mech.: Theory Exp. (2004) P10012.
[24] A. Capocci, V. D. P. Servedio, G. Caldarelli, and F. Colaiori, Physica A **352**, 669 (2005).
[25] M. E. J. Newman and M. Girvan, Phys. Rev. E **69**, 026113 (2004).
[26] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner, e-print arXiv:physics/0608255.
[27] S. Fortunato and M. Barthelemy, Proc. Natl. Acad. Sci. U.S.A. **104**, 36 (2007).
[28] W. Zachary, J. Anthropol. Res. **33**, 452 (1977).
[29] M. E. J. Newman, Phys. Rev. E **74**, 036104 (2006).
[30] Y. Fan, M. Li, P. Zhang, J. Wu, and Z. Di, Physica A **377**, 363 (2007).
[31] A. Li, *Fuzzy Mathematics and Application* (Metallurgical Industry Press, Beijing, 2005).
[32] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabasi, Nature (London) **407**, 651 (2000).
[33] H. Jeong, S. P. Mason, A.-L. Barabasi, and Z. N. Oltvai, Nature (London) **411**, 41 (2001).
[34] P. Gleiser and L. Danon, Adv. Complex Syst. **6**, 565 (2003).