Simple probabilistic algorithm for detecting community structure

Wei Ren,[*] Guiying Yan, Xiaoping Liao, and Lan Xiao

Academy of Mathematics and Systems Science, Chinese Academy of Sciences, No. 55 Zhongguncun East Road, Beijing, China

(Received 18 October 2007; revised manuscript received 1 December 2008; published 20 March 2009)

With the growing number of available social and biological networks, the problem of detecting the network community structure is becoming more and more important which acts as the first step to analyze these data. The community structure is generally regarded as that nodes in the same community tend to have more edges and less if they are in different communities. We propose a simple probabilistic algorithm for detecting community structure which employs expectation-maximization (SPAEM). We also give a criterion based on the minimum description length to identify the optimal number of communities. SPAEM can detect overlapping nodes and handle weighted networks. It turns out to be powerful and effective by testing simulation data and some widely known data sets.

DOI: 10.1103/PhysRevE.79.036111 PACS number(s): 89.75.Hc, 05.10.−a, 87.23.Ge

I. INTRODUCTION

Many systems can be represented by networks where nodes denote entities and links denote existing relation between nodes; such systems may include the web networks [1], the biological networks [2], ecological web [3], and social organization networks [4]. Many interesting properties have also been identified in these networks such as small world [5] and power-law distribution [6]. One property that attracts much attention is the network community structure, which is the phenomenon that nodes within the same community are more densely connected than those in different communities [7]. It is important in the sense that we can get a better understanding about the network structure.

This problem has been studied by researchers from different perspectives. Earlier approaches for identifying communities could be divided into two categories: the hierarchical approach and divisive approach. The former merged two closest nodes into one community recursively until the whole network became one single community, and the latter worked from the top to bottom which split the whole network into two communities recursively until every node was a community. These algorithms usually needed a measure to evaluate the closeness or dissimilarity between two nodes (see [7–11]).

An important modularity measure for evaluating the goodness of community structure was proposed by Newman and Girvan [12] and several algorithms worked by maximizing it [13–16]. This measure was very efficient in characterizing the community structure for networks with balanced structure; however, the internal scale problem in its definition [17] made it fail to work well for unbalanced networks such as those whose communities varied in size and degree sequence. Quite recently, an information-based algorithm by Rosvall and Bergstrom [18] accurately resolved communities and, in particular, can to some extent get over the scale problem of modularity.

Also, researchers [19] found that communities were overlapping rather than disjoint; subsequent algorithms [18,20,21] were designed to deal with overlapping communities. A mixture model by Newman and Leicht [22] could automatically detect patterns inside a network; meanwhile, it was able to detect overlapping nodes as a byproduct.

All these state-of-the-art algorithms motivate us to treat the community detection problem as a probabilistic inference problem; we should mind the internal information which determines the network topology. The internal information gives insight to the network structure. Our work is inspired by the probabilistic latent semantic analysis [23] which is a powerful algorithm in text mining; it models that a term occurs in a document if they are under the same latent topic. This idea is employed here to detect the community structure in complex networks. Compared to other algorithms [24,25], which also explore internal structure, our model possesses the mathematical simplicity and hence is easy to understand.

II. METHOD

Assume that the network considered is undirected and unweighted with n nodes; let A denote the adjacent matrix and $N(i)$ denote the neighbors of node i. Suppose c communities are to be detected, let π_r be the probability of community r, which can be viewed as the fraction of nodes in community $r, r=1,2,\ldots,c$. Instead of assigning a specific community to every vertex, we assume every vertex participates in every community, more specifically, community r selects node i with probability $\beta_{r,i}$ such that $\Sigma_{i=1}^{n}\beta_{r,i}=1$. For each community r, this is a multinomial distribution with parameters $\beta_{r,i}, i=1,2\ldots,n$, such that the community r chooses node i to represent itself with probability $\beta_{r,i}$; obviously, large value of $\beta_{r,i}$ indicates that node i is important in community r. Each node i participates in all community with $\beta_{1,i},\beta_{2,i},\ldots,\beta_{c,i}$. Note that there is no constraint $\Sigma_{r=1}^{c}\beta_{r,i}=1$, but different $\beta_{r,i}$ measures the relative importance of node i in community r.

For an edge between node i and j, consider it to be a node pair. Self-loop edge for node i is also allowed; for consistency, we model that a self-loop for node i is also regarded as pair $\{i,i\}$. The edge e_{ij} or—more precisely—the node pair $\{i,j\}$ is generated by the following finite mixture model where the community r is the latent variable.

[*] renwei@amss.ac.cn

(1) Select a community r with probability π_r. (2) Community r selects the node i with probability $\beta_{r,i}$; this is a multinomial distribution with parameters $(\beta_{r,1},\ldots,\beta_{r,n})$. (3) Meanwhile, community r selects the other node j with probability $\beta_{r,j}$, which is also a multinomial distribution with parameters $(\beta_{r,1},\ldots,\beta_{r,n})$

By assuming independence when community r selects node i,j in step 2 and step 3, the probability of choosing this particular node pair $\{i,j\}$ is

$$\mathrm{Prob}(e_{ij}|\pi,\beta) = \mathrm{Prob}(\{i,j\}|\pi,\beta) = \sum_{r=1}^{c} \pi_r \beta_{r,i}\beta_{r,j}.$$

Since each of the above three steps in this selecting process is a well-defined probability space, this probability $\mathrm{Prob}(e_{ij}|\pi,\beta)$ is indeed a legitimate probability due to the fact $\Sigma_{i=1}^{n}\Sigma_{j=1}^{n}\mathrm{Prob}(e_{i,j}|\pi,\beta)=1$. Note that $\mathrm{Prob}(e_{ij}|\pi,\beta)=\mathrm{Prob}(e_{j,i}|\pi,\beta)$; the above model makes no distinction between $\{i,j\}$ and $\{j,i\}$, which implies that this model only works for undirected networks.

Though $\mathrm{Prob}(e_{ij}|\pi,\beta)$ is defined from the process of selecting node pair $\{i,j\}$, it can also be viewed as the reliability of the edge between node i and j; high value of $\mathrm{Prob}(e_{ij}|\pi,\beta)$ is regarded as a reliable edge. To maximize the reliability of the edge, node i and node j should participate in some common community r with the large value of $\beta_{r,i},\beta_{r,j}$. For example, if there are totally two communities to detect, and suppose $\pi_1=\pi_2=0.5$, if $\beta_{1,i}=1/2n$, $\beta_{2,i}=2/n$, $\beta_{1,j}=1/2n$, and $\beta_{2,j}=2/n$ then $\mathrm{Prob}(e_{ij}|\pi,\beta)=17/8n^2$; but if $\beta_{1,i}=2/n$, $\beta_{2,i}=1/2n$, $\beta_{1,j}=1/2n$, and $\beta_{2,j}=2/n$ then $\mathrm{Prob}(e_{ij}|\pi,\beta)=1/2n^2$ which is a much lower reliability than the former one. In the former case, we see that i,j mainly participating in community 2 as is indicated by the value of $\beta_{2,i},\beta_{2,j}$; however, in the latter case, node i mainly participates in community 1 while node j mainly participates in community 2. So the idea in the definition of $\mathrm{Prob}(e_{ij}|\pi,\beta)$ is that if two nodes i,j are connected, they should participate mainly in some community r; in other words, important nodes in community r should be connected. This is exactly the common knowledge about the community structure.

Each edge is just the result of the above node pair selecting experiment. Naturally, the logarithm probability of network A under parameters π,β can be modeled as

$$\begin{aligned} L &= \ln \mathrm{Prob}(A|\pi,\beta) \\ &= \sum_{i=1}^{n} \sum_{j:j\in N(i)} \ln \mathrm{Prob}(e_{ij}|\pi,\beta) \\ &= \sum_{i=1}^{n} \sum_{j:j\in N(i)} \ln\left(\sum_{r=1}^{c} \pi_r\beta_{r,i}\beta_{r,j}\right). \end{aligned} \tag{1}$$

Parameters π,β should be estimated to maximize Eq. (1). However, L in Eq. (1) contains logarithm of sums and is difficult to optimize but can be optimized easily by the expectation-maximization (EM) algorithm.

Once all the parameters are estimated, the preference of node i belonging to community s is computed as $u_{s,i}=\pi_s\beta_{s,i}$, and node i is assigned to community r such that $r=\arg\max_s\{u_{s,i}=\pi_s\beta_{s,i},\ s=1,2,\ldots,r\}$. $u_{s,i}$s can be normalized so that their sum is 1 to comply with the probability normalization condition. In fact, this gives a soft assignment and can be used to detect overlapping nodes. Also this method does not definitely indicate whether an edge is an intercommunity or intracommunity, but in a probabilistic way [see Eq. (2)].

Suppose for node i, $r=\arg\max_s\{u_{s,i}=\pi_s\beta_{s,i},\ s=1,2,\ldots,r\}$, empirically node i is an overlapping node if there is another community s such that $\frac{u_{s,i}}{u_{r,i}}>1/10$. This algorithm appears to be highly similar to the mixture method by Newman and Leicht [22]: both methods are based on the mixture model and reply on EM algorithm to do the optimization. However, the model assumptions are quite different.

(1) First, in Newman's model, the parameter $\theta_{r,i}$ indicates the probability of group r linking toward node i which models linkage property; even if a node i does not belong to group r, there is still a chance that $\theta_{r,i}$ is large. However, in our model, $\beta_{r,i}$ is the probability of the node i appearing in community r which is the node property; the large value of $\beta_{r,i}$ indicates node i strongly belongs to community r.

(2) Second, in Newman's method, to evaluate the probability of a directed edge $e_{i,j}$, a group membership g_i has to be assigned to node i such that the probability of this edge is modeled as $\theta_{g_i,j}$. Quite differently, in our model, the probability of an edge $e_{i,j}$ or, precisely, the probability of choosing the node pair $\{i,j\}$ is modeled as $\Sigma_{r=1}^{c}\pi_r\beta_{r,i}\beta_{r,j}$. This means that the probability of this edge depends on the strength of node i,j's consistent participation in every communities. If both nodes strongly participate in some particular community r, $\mathrm{Prob}(e_{i,j}|\pi,\beta)$ tends to be large; on the other hand, if nodes i,j mainly participate in different communities, $\mathrm{Prob}(e_{i,j}|\pi,\beta)$ tends to be small. The above key difference differentiates the two models. Newman's mixture model consider a set of nodes to be a group if they have a similar linkage property while our model consider a set of node to be a community if they are densely connected. Of course as a result, the E step and M step of these two methods are different due to different model assumptions.

(3) The advantage of Newman's method is that it can detect a general pattern such as the bipartite structure; there are also potential disadvantages. The groups it detects consist of various kinds of patterns; say, some groups may be communities, other groups might be loosely interconnected. Our model is designed for detecting communities only. In Sec. III, we show experimentally that it does outperform Newman's model in detecting communities structure.

A. EM formula

The EM algorithm is proposed to maximize the probability that contains latent variables [26]; it computes the posterior probability of the latent variables under the observed data and currently estimated parameters in the E step, and updates parameters with these posterior probabilities in the M step. The posterior probability $\mathrm{Prob}(g_{ij}=r|A,\pi,\beta)$ of edge $e_{i,j}$ or node pair $\{i,j\}$ generated by community r conditional on the observed network A and parameter π,θ denotes this probability by $q_{ij,r}$, then

$$q_{ij,r} = \mathrm{Prob}(g_{ij} = r|A,\pi,\beta) = \frac{\mathrm{Prob}(g_{ij} = r, A|\pi,\beta)}{\mathrm{Prob}(A|\pi,\beta)} = \frac{\mathrm{Prob}(e_{ij}, g_{ij} = r, A|\pi,\beta)}{\mathrm{Prob}(A|\pi,\beta)}.$$

By simple deduction the E-step formula can be obtained,

$$q_{ij,r} = \mathrm{Prob}(g_{ij} = r|A,\pi,\beta) = \frac{\pi_r\beta_{r,i}\beta_{r,j}}{\sum_{s=1}^{c}\pi_s\beta_{s,i}\beta_{s,j}}. \tag{2}$$

In fact, $q_{ij,r}$ is the fraction of contribution from community r under the observed matrix A and parameters π,β. Obviously, the expected logarithm probability of the network is

$$\vec{L} = \sum_{i=1}^{n}\sum_{j:j\in N(i)}\sum_{r=1}^{c} q_{ij,r}\ln \mathrm{Prob}(e_{i,j}, g_{ij} = r|\pi,\beta) = \sum_{i=1}^{n}\sum_{j:j\in N(i)}\sum_{r=1}^{c} q_{ij,r}\ln(\pi_r\beta_{r,i}\beta_{r,j}). \tag{3}$$

Combining with the constraints that $\Sigma_r\pi_r=1$, $\Sigma_{i=1}^{n}\beta_{r,i}=1$, and $r=1,2,\ldots,c$, the Lagrange form of $\vec{L}$ is

$$D = \sum_{i=1}^{n}\sum_{j:j\in N(i)}\sum_{r=1}^{c} q_{ij,r}\ln(\pi_r\beta_{r,i}\beta_{r,j}) + \alpha\left(\sum_{r=1}^{c}\pi_r - 1\right) + \sum_{r=1}^{c}\gamma_r\left(\sum_{i=1}^{n}\beta_{r,i} - 1\right), \tag{4}$$

where $\alpha,\gamma_r, r=1,2,\ldots,c$ are Lagrange multipliers. The derivatives of D in Eq. (4) are

$$\frac{\partial D}{\partial \pi_r} = \sum_{i=1}^{n}\sum_{j:j\in N(i)} q_{ij,r} + \alpha, \tag{5}$$

$$\frac{\partial D}{\partial \beta_{r,i}} = \sum_{j:j\in N(i)} q_{ij,r} + \gamma_r. \tag{6}$$

By setting the derivative in Eqs. (5) and (6) to zero and combining the constraints $\Sigma_r\pi_r=1$, $\Sigma_{i=1}^{n}\beta_{r,i}=1$, and $r=1,2,\ldots,c$, the M-step formulas are

$$\pi_r = \frac{\sum_i\sum_{j:j\in N(i)} q_{ij,r}}{\sum_i\sum_{j:\in N(i)}\sum_{s=1}^{c} q_{ij,s}}, \tag{7}$$

$$\beta_{r,i} = \frac{\sum_{j:j\in N(i)} q_{ij,r}}{\sum_{k=1}^{n}\sum_{j:j\in N(k)} q_{kj,r}}. \tag{8}$$

In the E step, the membership of an edge is influenced by its nodes; while in the M step, the node importance in communities is influenced by the membership of all its links. By iterating E and M steps, L in Eq. (1) will increase.

Parameters π,β are initialized with random values and iterated using E and M steps until L stabilizes. To avoid the algorithm getting stuck in a local maxima, we adopt the restart strategy which runs the EM algorithm several times with different initial parameter values.

Suppose the network has totally l edges; obviously the algorithm has a linear time complexity $O(cl)$, which makes it an appealing approach for detecting large scale networks. Note that the actual running time is also relevant to the number of EM iterations and the number of restarts. We name our model, for easier representation, simple probabilistic algorithm which employs the idea of expectation and maximization (SPAEM) framework.

B. Model selection issue

SPAEM needs a prespecified community number c and this is regarded as *prior* knowledge. However, the determination of c is a nontrivial task and is difficult when no prior knowledge can be obtained. We try to handle it by using minimum description length principle [27].

In general, L in Eq. (1) increases as c increases; meanwhile, an extra cost has to be paid due to the increase in the number of parameters $K=(c-1)+c(n-1)$. There should be some balance between L and K, and the idea of minimum description length principle can be employed here [27]. According to this principle, the code length needed to describe the network data is composed of two parts where the first part describes the coding length of the network using SPAEM while the second part gives the length for coding all parameters of SPAEM itself. The length needed for the coding network using SPAEM is obviously $-L/2$ (note that every edge is added twice). To code the parameters, a precision ϵ has to be prespecified. With this precision ϵ, parameters smaller than ϵ are not coded and get a description length of 0; otherwise coding the parameter π_r needs length $\ln(\frac{\pi_r}{\epsilon})$ and $\beta_{r,i}$ needs length $\ln(\frac{\beta_{r,i}}{\epsilon})$, so the total length H for coding the model is

$$H = -L/2 + \sum_{r=1}^{c}\ln\left(\frac{\pi_r}{\epsilon}\right)I(\pi_r \geq \epsilon) + \sum_{r=1}^{c}\sum_{i=1}^{n}\ln\left(\frac{\beta_{r,i}}{\epsilon}\right)I(\beta_{r,i} \geq \epsilon). \tag{9}$$

Value c should be chosen as the one which generates the minimum description length H in Eq. (9). Choosing precision ϵ is tricky but is very important in Eq. (9). Smaller ϵ may cause longer code for parameters and hence will always prefer models with small c. In fact, it is shown that networks are organized in a hierarchical way [28]; the choice ϵ gives a lever for viewing networks in different resolutions. It is intuitively clear that ϵ should be on the scale of $1/n$ due to the normalization condition $\Sigma_{i=1}^{n}\beta_{r,i}=1$. Typically, if node i belonging to community r, $\beta_{r,i}$ will be on the scale of $1/n$ and be much smaller than $1/n$ if not belongs to this community. Here ϵ is set to $1/(3n)$. This precision is totally empirical but

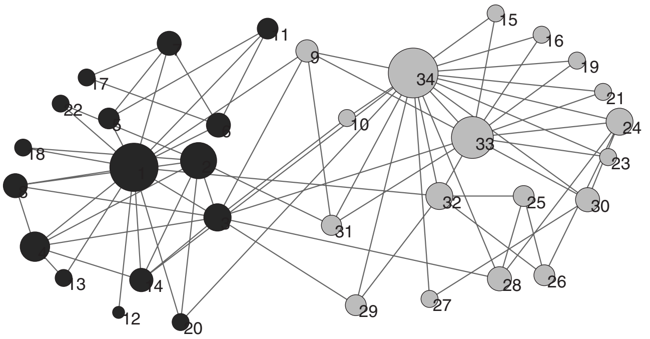

FIG. 1. Zachary club network: node color indicates community and node size indicates $u_{r,i}$. Clearly nodes 9, 10, and 31 are overlapping nodes and have been identified by our algorithm.

as will be shown in Sec. III that for well-clustered networks, the model selection results are robust to the choice of ϵ ranging from $1/n$ to $1/(7n)$.

III. EXPERIMENT

A. Zachary club network

The famous Zachary club network is about acquaintance relationship between 34 members [4]. The club splits into two parts due to an internal dispute so it naturally has community structure. By setting $c=2$, we run our algorithm and get exactly the original two communities (see Fig. 1). Node color indicates community and node size indicates the value of $u_{r,i}=\pi_r\beta_{r,i}$, which can partially measure the importance of node i in community r. Nodes 1, 2, 33, and 34 are important nodes found by SPAEM and can be verified intuitively from the network.

SPAEM gives soft assignment to each node so is capable of detecting overlapping nodes (see Table I). To compare the ability in detecting overlapping nodes, we also include q_{ir} used to assign communities in the mixture model [22]. Clearly, nodes 1, 2, 33, and 34 are not overlapping nodes but node 9 is. The mixture model also can detect this; however, by checking corresponding probabilities (see Table I) SPAEM shows more accuracy by revealing the extent of overlapping.

TABLE I. Result on Zachary network. $u_{r,i}$ is calculated by $u_{r,i}=\pi_r\times\beta_{r,i}$, which is interpreted as the preference of node i belonging to community r. The q_{ir}s in the mixture mode [22] are also included. To facilitate comparison, we normalize $u_{s,i}$ so they add up to 1.

Node ID	$u_{1,i}$	$u_{2,i}$	$\frac{u_{1,i}}{u_{1,i}+u_{2,i}}$	q_{i1} [a]
1	3.30×10^{-5}	0.1025	0.00	0.00
2	4.86×10^{-6}	0.0577	0.00	0.00
9	0.0219	0.0101	0.69	0.96
13	5.83×10^{-36}	0.0128	0.00	0.00
31	0.0179	0.0078	0.70	0.92
33	0.0769	1.55×10^{-8}	1.00	1.00
34	0.1090	8.20×10^{-6}	1.00	1.00

[a] q_{ir} is defined in [22] as the probability of node i belonging to community r.

B. American college football team network

The second network investigated is the college football network which represents the game schedule of the 2000 season of Division I of the U.S. college football league [7]. The nodes in the network represent the 115 teams, while the links represent 613 games played. The teams are divided into 12 conferences and generally games are more frequent between members of the same conference than between teams of different conferences.

The result of SPAEM and the mixture model [22] is depicted in Figs. 2 and 3, respectively. SPAEM basically uncovers the original community structure. However, the mixture model gets a very different result (see Fig. 3). This is because the groups it detects is a set of nodes with similar linkage property so it may not be common sense community. The three node group in the middle of Fig. 3 is obviously not a community. There are still other groups consisting of nodes from different communities (see Fig. 3). The mixture model can detect patterns but it cannot differentiate different kinds of patterns; in other words, it cannot tell whether a detected group is a community.

C. Comparison with other methods

A modularity measure $Q=\Sigma_{r=1}^{c}[\frac{l_{rr}}{l}-(\frac{d_r}{2l})^2]$ was proposed by Newman and Leicht [12], where l_{rr} is the number of links in community r, d_r is the total degree in community r, and l is the total number of edges in the network. Good community structure usually indicates a large value of Q. But there is a scale l in the definition of Q and this may cause a problem in some networks [17,18]. Such networks include those whose communities vary in size and degree sequence.

Dolphin social network reported by Lusseau *et al.* [3] provides a natural example where communities vary in size. In this network, two dolphins have a link with each other if they are observed together more often than expected by chance. The original two communities have different sizes, with one containing 22 dolphins and the other 40. Setting $c=2$, SPAEM only misclassifies one node and gets exactly the same result as the edge—betweenness algorithm [7] and the information-based algorithm [18]; however, the modularity based method [15] gets different result, as depicted in Fig. 4.

It is shown that the modularity algorithm works well for networks whose communities roughly have the same size and degree sequence but may not provide very competitive results when the communities differ in size and degree sequence [18]. To show the way SPAEM handles these situations, we conduct the same three sets of test as done in [18]: symmetric, node asymmetric, and link asymmetric. In the symmetric test, each network is composed of four communities with 32 nodes each; each node has an average degree of 16. k_{out} is the average number of edges linking to nodes in different communities. In the node asymmetric test, each network is composed of two communities with 96 and 32 nodes, respectively. k_{out} has the same meaning as in the symmetric

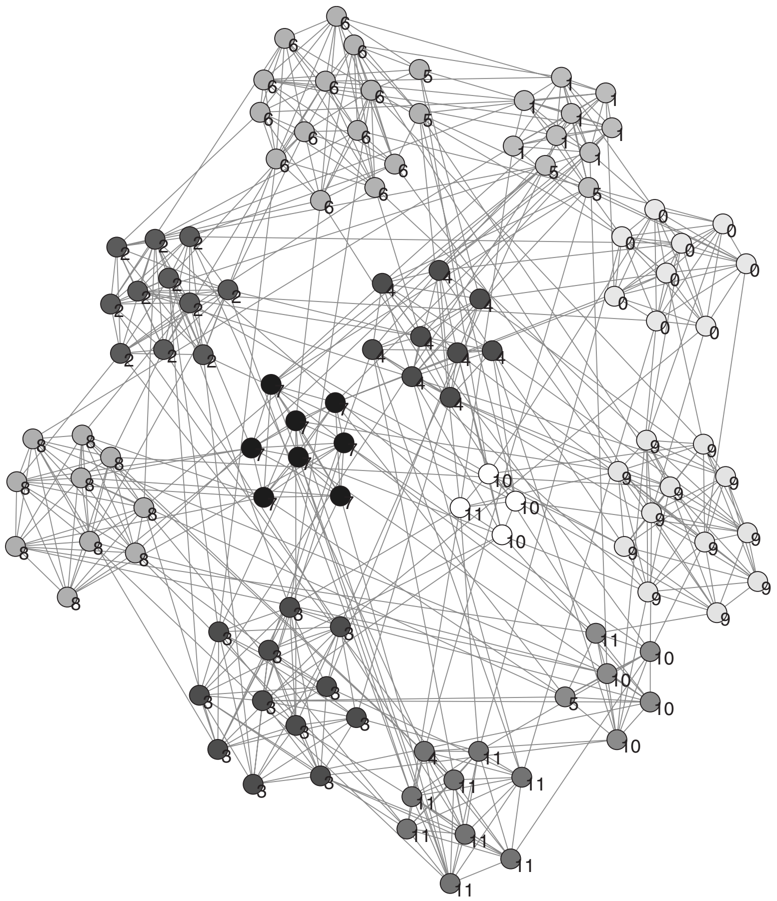

FIG. 2. Result of SPAEM for American football network: node label indicates the real community membership. Nodes belonging to the same community detected by SPAEM are placed adjacently.

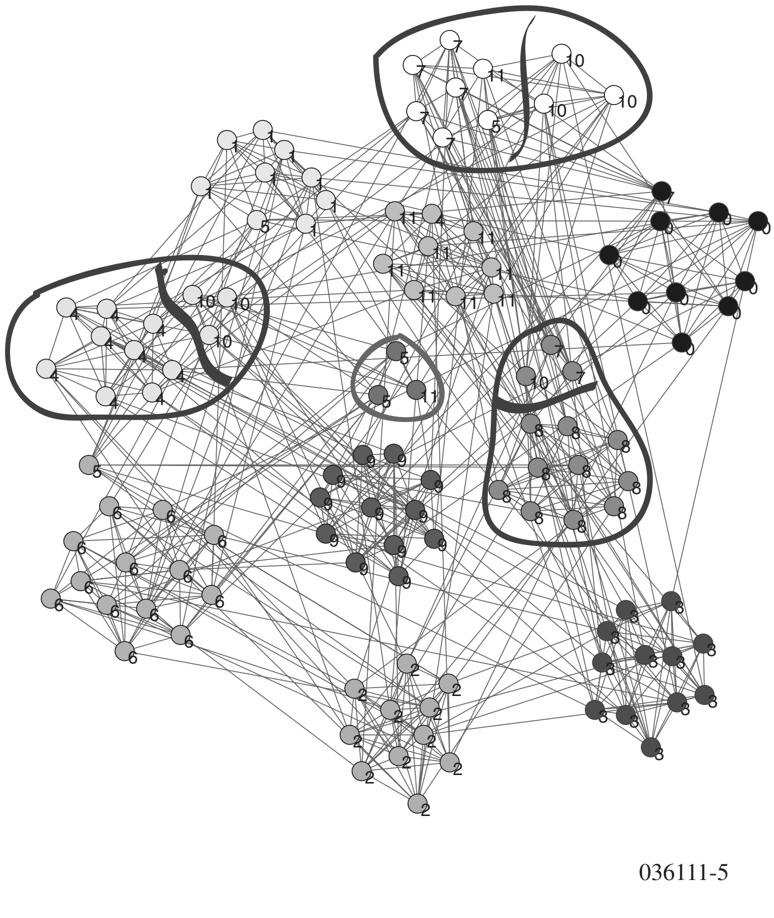

FIG. 3. Result of the mixture model [22] for American football network: node label indicates the real community membership. Nodes belonging to the same group detected are placed together; groups which are not the common sense community structure are marked using cycled line. Some of these groups are formed by nodes from two real communities. Also there is a three node group which is clearly not a community.

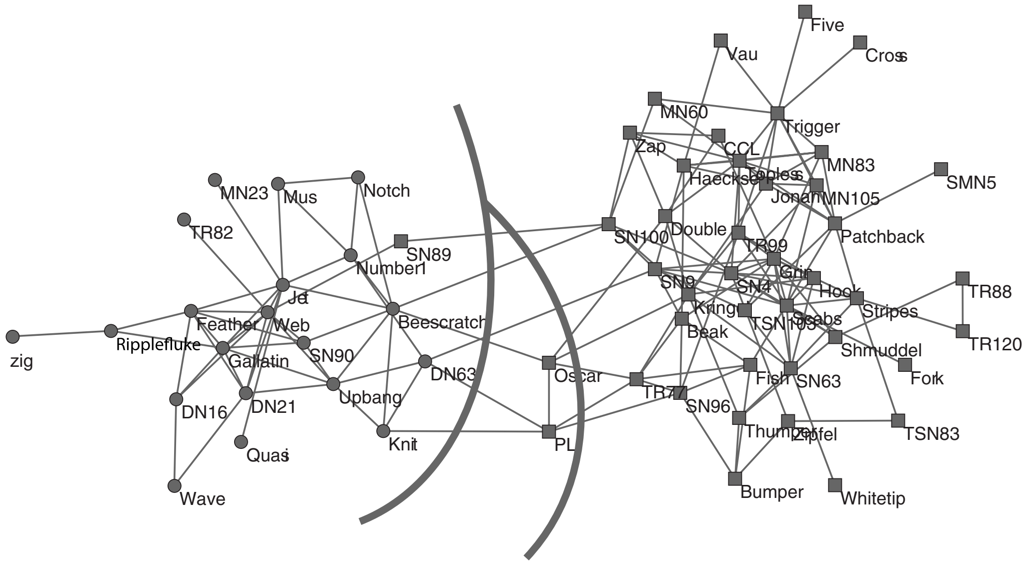

FIG. 4. Dolphin network: node shape denotes the real split. The left line shows the result by SPAEM with only one mistake, while the right line indicates the result in [15].

test. k_{out} is set to 6, 7, and 8 in both the symmetric and node asymmetric cases; as k_{out} increases, it becomes difficult to detect real community structure. In the link asymmetric test, two communities each with 64 nodes differ in their average degree sequence: nodes in one community have average 24 edges and in the other community have only eight edges, setting $k_{\text{out}}=2,3,4$. Table II gives the results of our algorithm compared to other algorithms [12,18,22]. Note that the results of the information algorithm and the modularity algorithm are cited from [18] while results of the mixture model are calculated by the authors. We have to admit that the information algorithm outperforms all three algorithms, especially in the node asymmetric and link asymmetric tests. SPAEM outperforms the modularity algorithm [12] in the symmetric and node asymmetric tests. The mixture model [22] seems to perform not so well in the symmetric test; this might be due to the fact that the groups it discovers may not be communities due to fuzzy structure of these networks as k_{out} increases.

D. Handling weighted network

SPAEM can also be extended to handle weighted networks. Suppose the weighted adjacent matrix of the network is $W_{n\times n}$ with its entries $w_{i,j}, i=1,2,\ldots,n$, $j=1,2,\ldots,n$; then the logarithm likelihood of the network becomes

$$L=\sum_{i=1}^{n}\sum_{j:j\in N(i)} w_{i,j}\ln\left(\sum_{r}\pi_r\beta_{r,i}\beta_{r,j}\right). \tag{10}$$

$\vec{L}$ becomes

$$\begin{aligned}\vec{L}&=\sum_{i=1}^{n}\sum_{j:j\in N(i)}\sum_{r} w_{i,j}q_{ij,r}\ln \Pr(e_{i,j}\in r)\\&=\sum_{i=1}^{n}\sum_{j:j\in N(i)}\sum_{r} w_{i,j}q_{ij,r}\ln(\pi_r\beta_{r,i}\beta_{r,j}).\end{aligned} \tag{11}$$

The E step is unchanged but M step becomes

TABLE II. Results on the benchmark test on three experiments: symmetric, node asymmetric, and link asymmetric.

Test	k_{out}	SPAEM	Compression[a]	Modularity[b]	Mixture[c]
Symmetric	6	0.99	0.99	0.99	0.92
	7	0.95	0.97	0.97	0.81
	8	0.84	0.87	0.89	0.64
Node	6	0.97	0.99	0.85	0.97
Asymmetric	7	0.92	0.96	0.80	0.92
	8	0.79	0.82	0.74	0.74
Link	2	0.98	1.00	1.00	0.99
Asymmetric	3	0.94	1.00	0.96	0.94
	4	0.84	1.00	0.74	0.70

[a] Information method in [18].
[b] Modularity based method in [12].
[c] Mixture model in [22].

TABLE III. Benchmark test on weighted network designed by [29]. There are four communities each with 32 nodes in the network with $k_{\text{out}}=8$. As w increases from 1.4 to 2, both methods respond positively but SPAEM gets better results.

	SPAEM	Markov[a]
$w=1.4$	0.96	0.89
$w=1.6$	0.98	0.94
$w=1.8$	0.99	0.97
$w=2$	0.99	0.98

[a] Random walk model in [29].

$$\pi_r = \frac{\sum_i \sum_{j:j\in N(i)} w_{i,j} q_{ij,r}}{\sum_i \sum_{j:j\in N(i)} \sum_s w_{i,j} q_{ij,s}},$$

$$\beta_{r,i} = \frac{\sum_{j:j\in N(i)} w_{i,j} q_{ij,r}}{\sum_{k=1}^{N} \sum_{j:j\in N(k)} w_{k,j} q_{kj,r}}.$$

Intuitively the M-step formula is reasonable since links with greater weights contribute more to corresponding parameters.

To test SPAEM on weighted networks, the simulation test is done as that in [29]. This set of tests is based on the above symmetric test when $k_{\text{out}}=8$. For each of the 100 networks in the symmetric test with $k_{\text{out}}=8$, the weight of edges within a certain community is raised to $w=1.4, 1.6, 1.8, 2$, while the weight of edges running between communities is unchanged (with weight 1). As weight w increases from 1.4 to 2, models should improve their power in detecting community structure. Results of SPAEM are shown in Table III as well as the results in [29] for comparison (note that the results in [29] are directly cited rather than recalculated). SPAEM generally outperforms the model in [29].

The limitation with the above simulation test is that any algorithm will respond positively when w increases and that the original unweighted networks already have clear community structure. Now we devise a more elaborate example. Consider a network with 32 nodes, each node pair has an edge with probability p_{rand}; obviously, this network has no community structure. Let nodes 1–16 be in group 1 and nodes 17–32 be in group 2. Weight of edges inside each group is raised to 1.5 with probability p_{weight} but the weight of edges running between groups is unchanged. Now the only thing that can differentiate these two groups is the weight of edges. By setting $p_{\text{rand}}=0.8$ and $p_{\text{weight}}=0.8$, SPAEM uncovers the two groups with only three mistakes (see Fig. 5). This shows that SPAEM is able to make good use of edge weight.

E. Model selection test

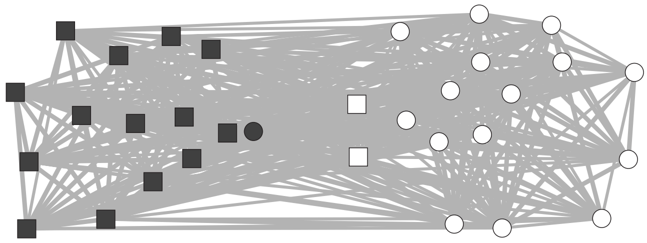

FIG. 5. Results on the simulated weighted network. Node shape shows the original community, while node color indicates the community structure detected by SPAEM.

Now, the minimum description length H defined in Eq. (9) is employed for SPAEM to select c, the optimal number of communities, and the precision is empirically set to $1/3n$. The criterion indicates that 11 communities in the American football network [7] should be detected [see Fig. 6(a)], the result seems to be wrong since there should be 12 communities; however, there is a conference "independents" which cannot be really a conference because teams in it play games with adjacent conferences. This criterion also determines four communities in the journal citation network [see Fig. 6(b)]. These two results show that H in Eq. (9) and precision $1/3n$ are sound.

To further test the validity of the model selection principle, model selection results on the above simulation experiments (symmetric, node asymmetric, and link asymmetric) are presented in Table IV. Combined with the model selection principle, SPAEM gives very competitive results in all these three tests. One weird thing is that in the node asymmetric case, the accuracy of SPAEM increases as k_{out} increases; this is partly because the penalty term for describing the model parameters in Eq. (9) favors small number communities. This also in turn verifies that selection criterion and the precision are reasonable.

F. Model selection discussion

The model selection criterion in Eq. (9) is sensitive to the choice of the accuracy ϵ; different ϵ would lead to different

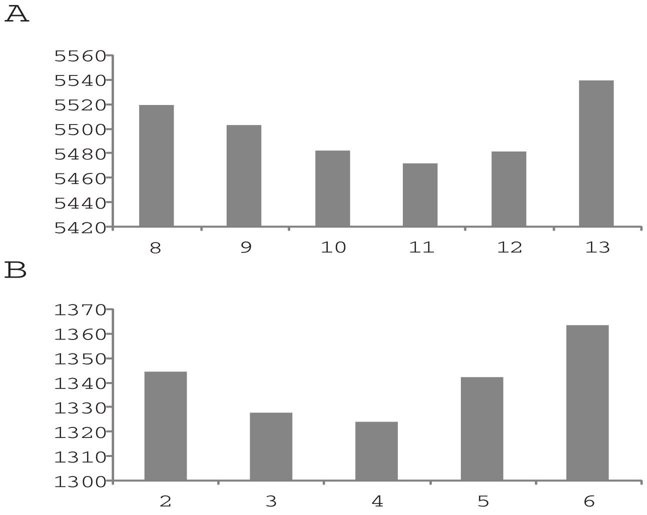

FIG. 6. (a) Model selection result for American football team network. (b) Model selection result for the journal citation network.

TABLE IV. Model selection result: each entry is the fraction of networks identified with the correct number of communities; the number in the parentheses indicates the average number of communities identified by the corresponding algorithm.

Test	k_{out}	SPAEM-MDL	Information[a]	Modularity[b]
Symmetric	6	1.00(4.00)	1.00(4.00)	1.00(4.00)
	7	1.00(4.00)	1.00(4.00)	1.00(4.00)
	8	0.65(3.60)	0.14(1.93)	0.70(4.33)
Node	6	0.82(2.18)	1.00(2.00)	0.00(4.95)
Asymmetric	7	0.83(2.17)	0.80(1.80)	0.00(4.97)
	8	0.93(2.07)	0.06(1.06)	0.00(5.29)
Link	2	1.00(2.00)	1.00(2.00)	0.00(3.10)
Asymmetric	3	1.00(2.00)	1.00(2.00)	0.00(4.48)
	4	1.00(2.00)	1.00(2.00)	0.00(5.55)

[a] Information method [18].
[b] Modularity method [12].

model selection results. Intuitively, small ϵ will favor smaller numbers of communities and large ϵ tends to identify large numbers of communities. In fact, it is shown that complex networks may be organized in the hierarchical structure which allows us to view them in different resolutions [28]. The accuracy ϵ indeed provides the capacity to detect communities in different resolutions.

However, it is expected that for networks with well-defined community structure, the model selection criterion should be robust to the choice of accuracy ϵ. To verify this, different accuracy ϵ ranging from $1/n$ to $1/7n$ are tested on the journal citation network [18]; this criterion identifies four communities for ϵ ranging from $1/n$ to $1/6n$ and three communities when $1/7n$, strongly indicating that this network actually has four communities. We further test how different ϵ will impact on the model selection result using the symmetric test when $k_{\mathrm{out}}=6,7,8$, respectively. For ϵ ranging from $1/2n$ to $1/7n$, this criterion nearly always identifies the correct number of communities when $k_{\mathrm{out}}=6,7$; however, when $k_{\mathrm{out}}=8$, the accuracy drops drastically, this is due to the fuzzy structure when there are too many edges linking to other communities. The above results show that the model selection criterion for SPAEM indeed is robust to the choice of ϵ for well-clustered networks.

IV. CONCLUSION

In this paper, we propose a probabilistic algorithm SPAEM to resolve the community structure. We have showed the power of SPAEM in detecting the community structure as well as providing more useful information. SPAEM is also extended to handle weighted network. To determine the optimal number of communities, the minimum description length principle is employed and tested on a variety of networks. Though the test networks in this study are mainly social networks, it should be claimed that the applicability of SPAEM is not confined to social networks but also include other types of network such as the biological network. To allow researchers to better use our algorithm, we make source code available [30,31].

TABLE V. Summary table: features of SPAEM and the mixture model [22].

	SPAEM	Mixture
Time Cost	$O(cl)$	$O(cl)$
Model Selection?	Yes	No
Weighted Graph?	Yes	No
Directed Graph?	No	Yes
Detect Pattern?	No	Yes

The mixture model in [22] is a good algorithm capable of detecting patterns and handling directed networks, while SPAEM focuses on detecting community structure. Experimentally SPAEM does perform better in uncovering community structure and identifying overlapping nodes. Though these two algorithms seem to be similar to each other, they are based on different model assumptions. Table V gives a summary on features of the two algorithms.

ACKNOWLEDGMENTS

This research is supported by the National Natural Science of Foundation of China under Grant Nos. 10531070, 10721101, and KJCX-YW-S7. The authors thank Dr. Martin Rosvall in Washington University and Dr. Lingyun Wu and Shihua Zhang in Chinese Academy of Science for their insightful comments on this paper. Dr. Chris Wiggins from Columbia University also gave valuable advice on this paper. The authors are also grateful to Professor Fengzhu Sun from University of Southern California. We specially thank the anonymous reviewers for their comments which help us to further explore the feature of our algorithm.

[1] L. C. Freeman, Am. J. Sociol. **98**, 152 (1992).
[2] L. H. Hartwell, J. J. Hopefield, S. Leibler, and A. W. Murray, Nature (London) **402**, C47 (1999).
[3] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, Behav. Ecol. Sociobiol. **54**, 396 (2003).
[4] W. W. Zachary, J. Anthropol. Res. **33**, 452 (1977).
[5] D. S. Watts, Nature (London) **4**, 409 (1998).
[6] A.-L. Barabasi and R. Albert, Science **286**, 509 (1999).
[7] M. Girvan and M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002).
[8] F. Luo, Y. Yang, C.-F. Chen, R. Chang, J. Zhou, and R. H. Scheuermann, Bioinformatics **23**, 207 (2007).
[9] H. Zhou, Phys. Rev. E **67**, 041908 (2003).
[10] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, Proc. Natl. Acad. Sci. U.S.A. **101**, 2658 (2004).
[11] S. Zhang, X. Ning, and X. Zhang, Eur. Phys. J. B **57**, 67

(2007).
[12] M. E. J. Newman and M. Girvan, Phys. Rev. E **69**, 026113 (2004).
[13] M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **103**, 8577 (2006).
[14] M. E. J. Newman, Phys. Rev. E **74**, 036104 (2006).
[15] S. White and P. Smyth, in SIAM International Conference on Data Mining 2005, *A Spectral Clustering Approach to Finding Communities in Graphs* (2005, unpublished).
[16] J. Duch and A. Arenas, Phys. Rev. E **72**, 027104 (2005).
[17] S. Fortunato and M. Barthlemy, Proc. Natl. Acad. Sci. U.S.A. **104**, 36 (2007).
[18] M. Rosvall and C. T. Bergstrom, Proc. Natl. Acad. Sci. U.S.A. **104**, 7327 (2007).
[19] G. Palla, I. Dernyi, I. Farkas, and T. Vicsek, Nature (London) **435**, 814 (2005).
[20] S. Zhang, R. S. Wang, and X. S. Zhang, Physica A **374**, 483 (2007).
[21] S. Zhang, R. S. Wang, and X. S. Zhang, Phys. Rev. E **76**, 046103 (2007).
[22] M. E. J. Newman and E. A. Leicht, Proc. Natl. Acad. Sci. U.S.A. **104**, 9564 (2007).
[23] T. Hofmann, Mach. Learn. **42**, 177 (2001).
[24] A. Vazquez, Phys. Rev. E **77**, 066106 (2008).
[25] J. J. Ramasco and M. Mungan, Phys. Rev. E **77**, 036122 (2008).
[26] A. Dempster, N. Laird, and D. Rubin, J. R. Stat. Soc. Ser. B (Methodol.) **39**, 1 (1977).
[27] J. Rissanen, Automatica **14**, 465 (1978).
[28] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A. L. Barabasi, Science **297**, 1551 (2002).
[29] N. A. Alves, Phys. Rev. E **76**, 036101 (2007).
[30] http://www.code.google.com/p/spaem
[31] See EPAPS Document No. E-PLEEE8-79-038903 for the C# source code of SPAEM. For more information on EPAPS, see http://www.api.org/pubservs/epaps.html.