Physica A 460 (2016) 98–104

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

An improved label propagation algorithm using average node energy in complex networks

Hao Peng [a,1], Dandan Zhao [a,1], Lin Li [b,1], Jianfeng Lu [a], Jianmin Han [a], Songyang Wu [c,*]

[a] *College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua 321004, China*

[b] *China Electronics Standardization Institute, Beijing 100007, China*

[c] *The Third Research Institute of Ministry of Public Security, Shanghai 201204, China*

HIGHLIGHTS

- We propose an improved label propagation algorithm (LPA) to uncover overlapping community structure.
- The improved LPA can identify the bridge nodes in each iteration and then we can uncover overlapping communities when the iteration terminates.
- The introduced algorithm can effectively uncover reasonable overlapping community structures in the real-world and social networks.

ARTICLE INFO

Article history:
Received 24 October 2015
Received in revised form 11 April 2016
Available online 10 May 2016

Keywords:
Community structure
Label propagation
Average node energy

ABSTRACT

Detecting overlapping community structure can give a significant insight into structural and functional properties in complex networks. In this Letter, we propose an improved label propagation algorithm (LPA) to uncover overlapping community structure. After mapping nodes into random variables, the algorithm calculates variance of each node and the proposed average node energy. The nodes whose variances are less than a tunable threshold are regarded as bridge nodes and meanwhile changing the given threshold can uncover some latent bridge node. Simulation results in real-world and artificial networks show that the improved algorithm is efficient in revealing overlapping community structures.

1. Introduction

A wide variety of complex systems can be regarded as complex networks which are composed of vertices connected by edges [1–3], and understanding the properties of these networks is a powerful tool for analyzing complex systems. Community structure is one of the important topological property in complex networks. The communities are groups of nodes within which nodes are densely connected. Since this property gives a valuable insight into the functional behavior and topological feature of various complex systems [4], detecting and analyzing community structure has attracted great attentions in recent years.

Some existing works have been developed to uncover reasonable community structure, especially in the last few years. The Kernighan–Lin algorithm [5] and the spectral analysis algorithm [6] are two typical methods to detect communities.

[*] Corresponding author.
E-mail address: wusongyang@stars.org.cn (S. Wu).

[1] These authors contributed equally to this work.

http://dx.doi.org/10.1016/j.physa.2016.04.042
0378-4371/© 2016 Elsevier B.V. All rights reserved.

Recently, Newman and Girvan proposed a seminal algorithm to split the network into hierarchical communities using the definition of betweenness which is the fraction of the shortest paths passing on each edge [7]. Then Clauset and Newman proposed the definition of Modularity which can evaluate the partition of communities in the complex network [8]. Based on the definition of Modularity, many kinds of algorithms are proposed to uncover community structure in complex networks. For example, Fang Wei et al. improved Modularity to uncover local community structure which lead to a local expansion algorithm [9]. Liu Xu et al. proposed an agglomerative clustering algorithm which based on neighborhood similarity to reveal community structure in complex networks [10]. Gong Maoguo et al. solved the community detection as a multiobjective optimization problem using multiobjective evolutionary algorithm [11]. To uncover community structure in a near linear time, Raghavan et al. proposed a notable algorithm named label propagation algorithm (LPA) by employing a simple label propagation in each iteration [12]. The LPA identified communities after propagating label among nodes until each node owns the label shared by most of its neighbors. All the algorithms above can only generate the hard-partition of a complex network. Recently, however, many modules are proofed to be overlapping communities. Thus many of the fore-mentioned algorithms are expanded to reveal overlapping communities in complex networks. Palla et al. proposed a clique percolation method (CFinder) to uncover overlapping community structure using the detected cliques [13]. Lancichinetti et al. developed an algorithm named OSLOM to uncover overlapping communities [14]. OSLOM is a local optimization of a fitness function which expresses the statistical feature of a community structure. Gregory et al. employed a tunable parameter υ to allow each node belong to υ communities at the most and improved classic LPA to uncover overlapping community structure (COPRA) [15]. All these valuable works can detect overlapping communities in the complex network. However, when detecting overlapping scopes, they also have some limitations in function or accuracy.

In this paper, we introduce the average node energy to express the bridge nodes and the improved LPA algorithm to detect overlapping communities. Under this way, we use a given parameter δ_{ave} to calculate a threshold which labels the latent bridge nodes and any nodes whose variance are less than the threshold will not update its label in the iteration. When all the labels stop updating, the overlapping community structure and bridge nodes are unambiguous.

2. The improved LPA using average node energy

In order to describe the algorithm proposed in this paper, we first define some notations as follows. Let $G = (V, E)$ be an undirected weight network, where V is node set and E is edge set. $L = \{L_1, L_2, \ldots, L_M\}$ is the label set which presents all active labels of nodes in network. $|L_i|$ $(i = 1, 2, 3, \ldots, M)$ is the number of nodes whose label is L_i. l_i is the label set and all elements in l_i present the label obtained by node i's nearest neighbors. Let $N_i(k)$ be the node set. All nodes in $N_i(k)$ are neighbors of i and labeled by k. $|N_i|$ is the number of nodes connecting with node i. $W_i(k) = \sum_{j \in N_i(k)} w_{ij}$ where w_{ij} is weight of edge between node i and j.

2.1. Label propagation

To detect community structure in a near linear time, Raghavan et al. proposed the label propagation algorithm which can reveal community structure following a simple workflow. Each node is identified by a unique label which implies different community identifier. Then each node asynchronous or synchronous updates its label with the label shared by most of its neighbors iteratively. If more than one label to which the maximum number of its neighbors belongs, one of them is chosen randomly. When all nodes own the label shared by most of its neighbors, the iteration terminates and meanwhile nodes with the same label belong to one community. The LPA workflow is simple and can detect some reasonable communities in the near linear time. However, LPA still owns some defects which limit its performance and application. For example, the propagation nature of LPA can cause the detected communities fall into local optimum. The label feature that limits the LPA cannot uncover overlapping community structure in the complex network. Some valuable works are proposed to solve the existing problems. Leung et al. used geodesic distance to uncover the overlapping communities. However it is difficult to know the diameter of all communities in advance. COPRA allows each node associate with more than one label which is more reasonable, but the parameter υ which means each node belong to υ communities at the most is also unpredicted for many complex networks. For this reason, in this paper, we introduce an improved LPA to detect overlapping community in the complex networks.

2.2. The improvements of LPA

The classic LPA updates the label of each node with the label used by the greatest number of its neighbors. Thus all nodes choose only one label to update in each iteration, and this rule results in the hard partitions for LPA. If some bridge nodes which belong to more than one community exist in the complex network, they will oscillate among different communities in iterations. Although there is no quantitative definition for a bridge node, most researchers identify a node almost equally connecting with more than one community as a bridge node. If these nodes can be identified and do not update their labels, the overlapping communities will be uncover when the algorithm terminates.

As is shown in Fig. 1, let $W_R = \sum_{t=1}^{4} R_t$, $W_Y = \sum_{t=1}^{3} Y_t$ and $W_B = \sum_{t=1}^{3} B_t$. Then if $W_R \approx W_Y \approx W_B$, node A is a bridge node. If we associate each node with a random variable which means the sum of weight between this node and the nodes

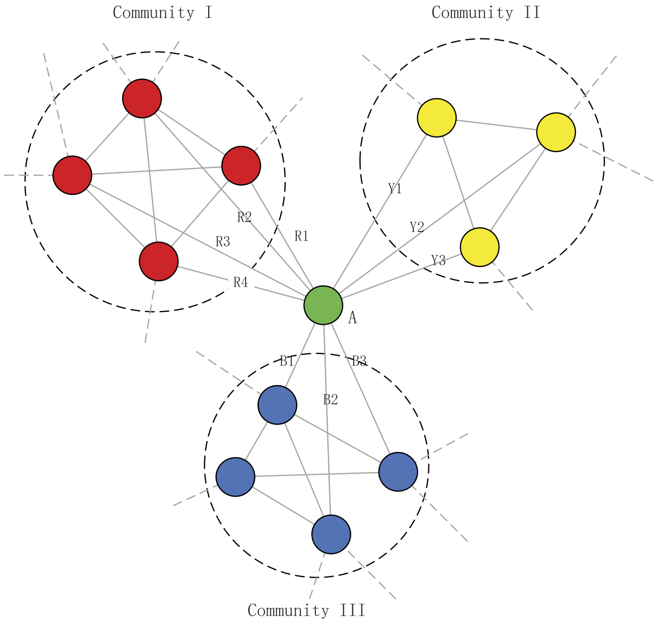

Fig. 1. A local topology with a bridge node and three communities.

belonging to its one neighbor community, the weight sum between a node and one of its nearest neighbor community can be regarded as sample of the random variable. Then we use variance of the proposed random variable to identify the balance of weight sum of node i and use the formula (1) to calculate:

$$E(i) = \frac{\sum\limits_{k\in l_i}(W_i(k) - \overline{W_i})^2}{n} \tag{1}$$

where $\overline{W_i} = \frac{\sum_{k\in l_i} W_i(k)}{n}$ is the average value of $W_i(k)$ and n is the number of elements in l_i. Once labels update, in practice, $E(i)$ $i \in [1, N]$ is clustered into two groups: a group whose value approaches to zero while the other group tends to a large value. From Eq. (1) and the practice, we get the conclusion: the smaller $E(i)$ is, the more probable than node i is the bridge node ($E(i) \neq 0$). Since the $E(i)$ represents the steady state of the node, we can regard $E(i)$ as the energy of node i. The smaller $E(i)$ is, the steadier of the state of node i. For a network which owns N nodes, $E_G = \sum_{i=1}^{N} E(i)$ is the total energy of the system. Based on E_G, we can define average node energy and calculate it in Eq. (2)

$$E_{ave} = E_G/N. \tag{2}$$

For one node i in the network, if $E(i)$ is smaller than a ratio of E_{ave}, it is a bridge node. So we define a threshold δ_{ave} which means a ratio of the average node energy, any nodes whose $E(i)$ is less than the value $E_{ave} \times \delta_{ave}$ will be regarded as bridge node and not update its label in iterations. We notice that δ_{ave} can be used as the focal length of the algorithm, larger δ_{ave} can identify more latent nodes as bridge nodes while smaller one shows a large community structure. However one bridge node i does not mean that the $E(i)$ is large. For example, in Fig. 1, if $W_R = 45$, $W_Y = 45$ and $W_B = 10$, the $E(i)$ is large but it is a bridge node which connects with red and yellow communities. In order to detect these nodes, we further process the samples of the nodes i whose $E(i)$ is larger than the given threshold. For a node i who belongs to M communities, its connections with the nearest neighbor community are $W_i(k),\ k \in [1, M]$. Without loss of generality, we assume that $W_i(k),\ k \in [1, M]$ is sorted from large to small. Then we can get another sequence $\Delta W_i(1),\ \Delta W_i(2), \ldots, \Delta W_i(M-1)$, where $\Delta W_i(k) = W_i(k) - W_i(k+1)$. After finding maximum of $\Delta W_i(t)$, $t \in (1, M-1]$ and abandoning the samples behind t, we can calculate the $E'(i)$ using (1) with the left samples. The nodes whose $E'(i)$ are smaller than the second threshold δ' still will not update its label. We apply the proposed algorithm in many networks and find that the second threshold δ' has similar performance when δ' is little larger than zero. Thus in our design, the default value of δ' is 0.001.

Since the nodes update their labels only depending on the local densely connected substructures, LPA may involve most of nodes into one giant community with certain probability. For example, we run LPA algorithm 500 times in the Karate club network [16], and the LPA clusters all nodes into one giant community in about 80 times, the percentage is 16%. In this paper, we consider community size into process of updating node label to prevent the unreasonable giant community structure. The updated label of node i is generalized by:

$$L'_i = \arg\max_{K} \frac{W_i(K)}{|K|^{1/3}}. \tag{3}$$

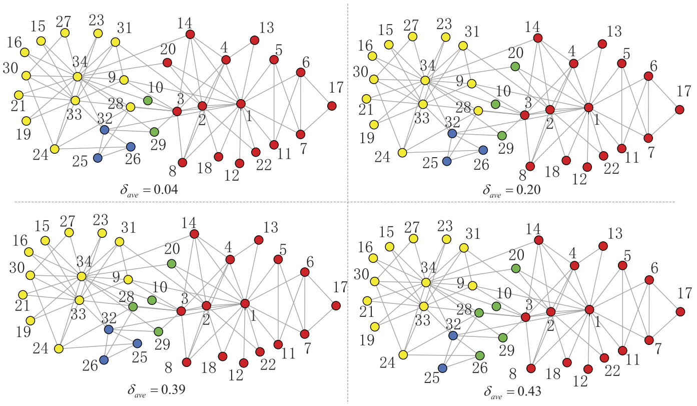

Fig. 2. The result of the Zachary's Karate network using the proposed method. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

When we use Eq. (3) to update node label in Karate club network, the frequency of giant community is less than 2%.

2.3. Workflow

According to the forementioned concepts and formulas, we proposed the improved LPA algorithm to detect overlapping community structure in the complex networks. The detailed description of the introduced algorithm is as follows:

***Step* 1**: Initialize each node with a unique label.

***Step* 2**: For each pair of nodes i and $j(i \neq j)$, if there is an edge connecting them, use $|N_i \cap N_j|$ as the similarity of the edge.

***Step* 3**: For each node i, calculate the $E(i)$ using (1) of each node and E_{ave} using (2).

***Step* 4**: For the nodes i whose $E(i) > E_{ave} \times \delta_{ave}$, calculate the sequence $\Delta W_i(1), \Delta W_i(2), \ldots, \Delta W_i(M-1)$ and $E'(i)$. if $E'(i) > \delta'$, update the label using (3).

***Step* 5**: If all nodes have a label which has shared by most of its neighbors or the updating is steady, then stop the algorithm. Else, go back to ***Step* 3** and iterate the procedure.

3. Experiments and discussion

To validate the performance of the proposed algorithm, we test and verify it to a number of real-world networks with known community structures and artificial networks. These networks include the Karate club network [13], the National Collegiate Athletic Association(NCAA) College-Football network, the classic Girvan–Newman artificial network [6] (GN artificial network) and LFR benchmark networks [14].

3.1. Karate network

The Zachary's karate network is a well-known benchmark network in social network. In the 1970s, Zachary observed and described the karate network with a graph which contained 34 nodes and 78 edges. During the two years observation, the administrator and instructor arose a dispute and then the club was divided into two smaller communities.

As shown in Fig. 2, the proposed algorithm divides the network into three communities. The red and yellow partition respectively represent administrator and instructor community, blue community is a small community which corresponds to the definition of community structure and many works have proofed the rationality of this partition. Green nodes represent the bridge nodes who belong to more than one community. Comparing the partition result when $\delta_{ave} = 0.04$ with the modularity-based algorithm [6], the proposed algorithm identifies red and yellow communities successfully. Modularity-based algorithm also finds community 24, 25, 26, 28, 29, 32 which is clearly unreasonable, because node 24 connects with the community of instructor more densely than with blue community. When the parameter δ_{ave} changes from 0.04 (4%) to 0.43 (43%), more and more nodes are identified as the bridge node. For example, only node 10 and 29 are identified as bridge nodes if $\delta_{ave} = 0.04$. When $\delta_{ave} = 0.39$, node 20 and 28 are labeled as green nodes. Although not all bridge nodes are reasonable when δ_{ave} changes, Fig. 1 implies that the parameter δ_{ave} can be used as focal length to uncover the scope of the communities and some latent bridge nodes.

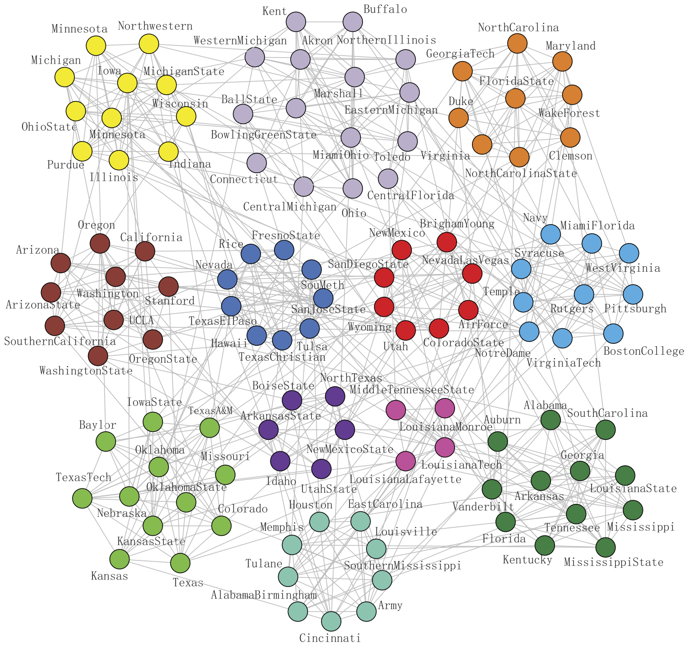

Fig. 3. Clustering result of NCAA football network with communities are shown in different colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.2. NCAA College-football network

The second benchmark network is NCAA College-football network. In Ref.4, Girvan and Newman proposed the football network whose 115 nodes were described as college football teams and edges represent regular season game between the two teams they connect.

As is shown in Fig. 3, the proposed algorithm divide the football network into 12 communities and nodes within the same community are expressed by the same color. Five of the communities are correctly detected: Atlantic Coast, Pac10, Big10, Big12 and Mountain West. All of the other communities miss one or two nodes and totally only eight nodes are included into the incorrect conferences. Further analysis shows that the misclassified nodes are also reasonable for the community definition. For example, in the dark blue community, node Texas Christian does not belong to this cluster in the real-world. However, when we check the topology structure of the dark blue community, we realize that this node connects with all the other nodes in the cluster which means the Texas Christian team is reasonable in this community. This eleven-community partition is better than the results got from Newman in Ref. [4] and classic LPA. The Girvan and Newman misclassified 11 nodes and LPA identified 10 nodes into the incorrect communities. There are no bridge nodes because each team plays more games within conference.

3.3. GN artificial network

Based on the forementioned analysis, the proposed algorithm can uncover reasonable community structure in many real-world networks. Since real-world network can only give a reasonable community structure instead of the unique correct cluster result, to further test the proposed algorithm, we use some artificial networks to evaluate the performance of the algorithm. Computer-generated networks obtain clear and predetermined community structure, thus we can use the Normalized Mutual Information (NMI) [17] to compare the quality of clusters generated by different algorithms. In Ref. [4], Girvan and Newman proposed an artificial network named GN artificial network. The GN network contains 128 nodes which are divided into 4 equal sized communities with 32 nodes for each. The average degree $\langle k\rangle$ of each node is equal to 16 and the edges are generated independently at random between pairs of nodes with probabilities depending on whether the two nodes belong to the same community or not. Each node has k_{in} edges on average connecting with nodes in the same community and k_{out} edges between communities, and $k_{in} + k_{out} = 16$.

Fig. 4 shows the cluster results of four algorithms: Spectral Partitioning [18], LPA, Hierarchical Clustering [19] and the algorithm proposed in this paper. When $k_{out} < 3$, the proposed algorithm performs similar with LPA and better than Spectral and Hierarchical Clustering. The LPA performs worst when $k_{out} > 4$. The proposed algorithm outperforms Spectral Partitioning and Hierarchical Clustering algorithm when k_{out} is less than 7. When $k_{out} > 8$, each node has approximate intra-community edges and inter-community edges which leads to the unreasonable community structure, so all the algorithms fail to uncover community structure in the GN network.

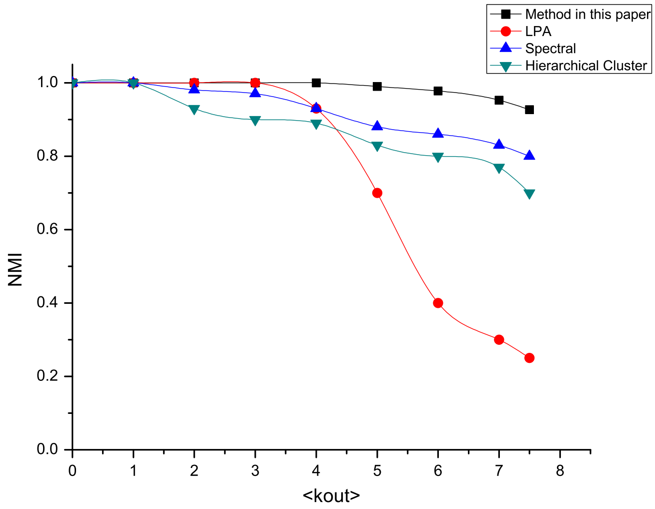

Fig. 4. Compare the proposed algorithm against other three typical methods in GN artificial network. The artificial network consists of 128 nodes which are divided into 4 communities. $k_{in} + k_{out} = 16$.

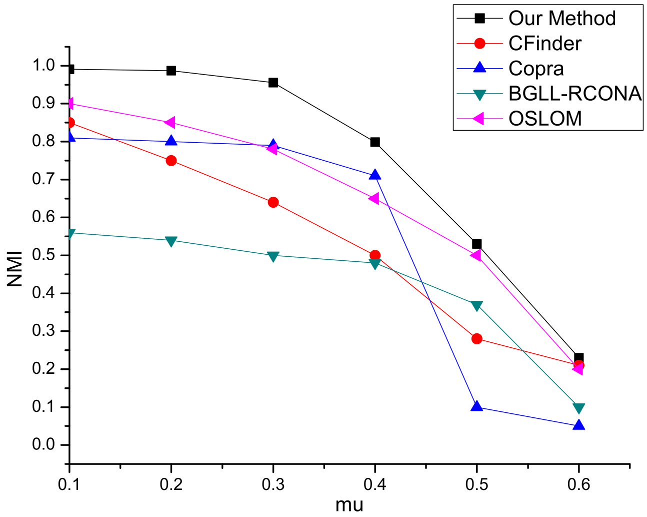

Fig. 5. Clustering results in LFR artificial network with community size changing from 10 to 50.

3.4. LFR benchmark network

Since the GN artificial network only generates communities with equal size and hard-partition. In Ref. [20], Lancichinetti et al. proposed a more realistic benchmark networks named LFR benchmark networks which own heterogeneous distribution of community size and node degree. So we can further compare the proposed algorithm with other four methods. In this paper, we generate the LFR benchmark with 1000 nodes and the community size changes from 10 to 50 and 20 to 100. The mixing parameter μ shows the fraction of one node's links connecting with the nodes outside its community. In order to evaluate the performance of five methods in overlapping community, we use the improved NMI proposed by Lancichinetti [21]. The improved NMI allows one node belong to more than one community, thus it can be used to evaluate overlapping communities.

Fig. 5 shows the cluster results of the five algorithms with the community size changing from 10 to 50. All of them are proposed to uncover overlapping community. When $\mu \leq 0.4$, the proposed algorithm performs best. $\mu = 0.5$ implies that most nodes in the network have equal number of intra-community edges and inter-community edges, thus all the five algorithms are difficult to uncover communities. Fig. 6 shows the NMI values of the five algorithms when the community size changing from 20 to 100. Larger community size implies more dense connection between pairs of communities, and this leads to a fuzzy community structure in the complex network. We notice that the proposed algorithm performs best when $\mu \leq 0.3$ and has the similar cluster results for $\mu \geq 0.4$. From Figs. 5 and 6, We can get the conclusion that the proposed algorithm can detect reasonable communities and bridge nodes in LFR artificial networks. For the groups of nodes which

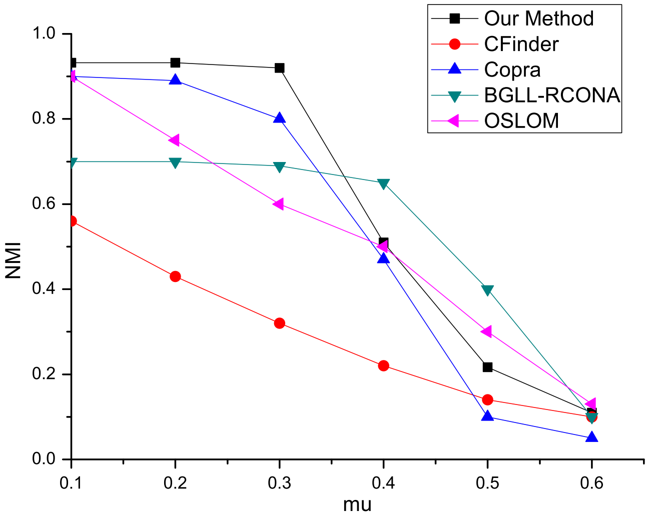

Fig. 6. Clustering results in LFR artificial network with community size changing from 20 to 100.

do not corresponding to the definition of communities, the proposed algorithm performs similar with some other famous methods.

4. Conclusion

In this paper, we expand the classic LPA to detect overlapping communities in the complex network. The improved LPA can identify the bridge nodes in each iteration and then we can uncover overlapping communities when the iteration terminates. At the same time, we notice that the introduced algorithm can effectively uncover reasonable overlapping community structures in the real-world and artificial networks. However, since the topological complexity of the large-scale social network restricts the robustness of community detection, in the future, we will focus on improving the performance of the proposed algorithm in large-scale social networks.

Acknowledgments

This work was partly supported by Zhejiang Provincial Natural Science Foundation of China (Grant No. LQ16F020002), Social development project of Zhejiang provincial public technology research (Grant No. 2016C33168), MOE (Ministry of Education in China) Project of Humanity and Social Science (Grant No. 15YJCZH125 and Grant No. 12YJCZH142), National Natural Science Foundation of China (Grant Nos. 61170108 and 61402418), Key Lab of Information Network Security, Ministry of Public Security (Grant No. C15610) and Shanghai Information Security Key Laboratory of Integrated Management of Technology (Grant No. AGK2013003).

References

[1] P. Latouche, E. Birmele, C. Ambroise, Ann. Appl. Stat. 5 (2011) 309–336.
[2] J. Gao, B. Barzel, A.L. Barabsi, Nature 530 (2016) 307–312.
[3] G. Dong, J. Gao, R. Du, et al., Phys. Rev. E 87 (2013) 052804.
[4] X. Liu, H.E. Stanley, J. Gao, Proc. Natl. Acad. Sci. 113 (2016) 201523412.
[5] B.W. Kernighan, S. Lin, Czech Math. J. 23 (1970) 298.
[6] M. Fiedler, Phys. Rev. E 69 (1997) 298.
[7] M.E.J. Newman, Girvan, Phys. Rev. E 69 (2004) 026113.
[8] A. Clauset, M.E.J. Newman, C. Moore, Phys. Rev. E 70 (2004) 066111.
[9] W. Fang, W. Chen, L. Ma, A.Y. Zhou, Prog. WWW Res. Dev. (2008) 43–55.
[10] X. Liu, Z. Xie, D.Y. Yi, Chin. Phys. Lett. 29 (4) (2012) 048902.
[11] M.G. Gong, L.J. Ma, Q.F. Zhang, L.C. Jiao, Physica A 390 (2012) 4050–4060.
[12] U.N. Raghavan, R. Albert, S. Kumara, Phys. Rev. E 76 (2007) 036106.
[13] G. Palla, I. Derenyi, I. Farkas, T. Vicsek, Nature 435 (2005) 814.
[14] A. Lancichinetti, F. Radicchi, J. Ramasco, PloS One 6 (2011) e18961.
[15] S. Gregory, New J. Phys. 12 (2010) 103018.
[16] W.W. Zachary, J. Anth. Res. 33 (1977) 452.
[17] H. Peng, D. Zhao, X. Liu, et al., PloS One 10 (2015) e0144153.
[18] M.E.J. Newman, Phys. Rev. E 74 (2006) 036104.
[19] R.J. Snchez-Garca, M. Fennelly, S. Norris, et al., IEEE Trans. Power Syst. 29 (2014) 2229–2237.
[20] A. Lancichinetti, S. Fortunato, F. Radicchi, Phys. Rev. E 78 (2008) 046110.
[21] A. Lancichinetti, S. Fortunato, J. Kertesz, New J. Phys. 11 (2009) 033015.