Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 15, no. 12, December 2019, pp. 3161-3170
DOI: 10.23940/ijpe.19.12.p7.31613170

A Complex Network Overlapping Community Detection Algorithm based on K-Cliques and Fitness Function

Jian Ma[a,b,*] and Jianping Fan[a,b]

[a]*School of Computer and Information Technology, Beijing Jiaotong University, Beijing, 100044, China*
[b]*Research Center for High-Speed Railway Network Management of Ministry of Education, Beijing Jiaotong University, Beijing, 100044, China*

Abstract

This paper presents an algorithm for detecting overlapping communities in complex networks. The algorithm draws on the idea of the clique as the core of the community, and proposes to treat the overlapping community as a collection of all k-cliques. The algorithm uses random nodes as the initial community, and each iteration selects the node with the maximum fitness value of the community neighbor. All k-cliques of the node are added to the community. During the process, nodes with negative fitness are removed. It then realizes the partition of network community structures and detects overlapping nodes. In many experiments of computer-generated networks and real-world networks, algorithms based on this idea have achieved good experimental results, which also illustrates the feasibility of this idea. Furthermore, the time efficiency and complexity of the algorithm is also acceptable. This algorithm also has better community discovery results.

Keywords: complex network; community detection; overlapping community detection; fitness function; k-clique

(Submitted on July 11, 2019; Revised on November 14, 2019; Accepted on December 14, 2019)

1. Introduction

Complex network cluster structure is one of the most important and common topological properties of complex networks, also known as community. The internal nodes of the community are closely connected and the nodes of different communities are sparsely connected. Communities can overlap with each other. For example, a person in a social network can have multiple groups of friends, a word in a word network can have many different semantics, and a neuron can belong to different parts of the nervous system.

The clique Percolation Method (CPM) proposed by Gergely Palla et al. in Nature in 2005 was an algorithm that could find overlapping community structures, and it held that the network community was a set of completely connected subgraphs (cliques) with common nodes. CFinder was a software tool based on the CPM algorithm [1]. The LFM algorithm could also detect overlapping community structures, which was a greedy optimization algorithm [2]. [3-7] showed that the clique's social network theory that can be detected. [8] related an optimization method based on nonnegative matrix factorization (NMF algorithm). In 2010, Lee et al. proposed the Greedy Clique Expansion algorithm (GCE algorithm). The algorithm first found the largest clique as the seed and then performed local optimization on the community function through the greedy search strategy of the LFM algorithm, so as to expand these seeds to detect overlapping communities [9]. In the Speaker-Listener Label Propagation Algorithm (SLPA algorithm) [10], each node could be a listener or speaker. The role changes depending on whether the node acted as an information provider or consumer. It is possible that nodes may have any number of labels. In general, the more a node is observed, the more likely it is to propagate their labels to other nodes. [11] studied the common overlapping and hierarchical structure of communities, in order to unite overlapping hierarchies. A general approach based on hierarchical linked communities was proposed.

[12] proposed a new method to detect community nodes. First, the local node with the maximum degree associated with a given node is found, and then the local modularity of the community was calculated through the found node to

* Corresponding author.
E-mail address: 13112083@bjtu.edu.cn

detect the community structures. The experimental results showed that the method was feasible, effective and flexible, especially when the given node was at the boundary of the community. [13] proposed a complex network overlapping community detection algorithm based on single-step adding cliques. This algorithm took a clique as the initial community and took into account the relationship between adding nodes and existing communities, as well as the internal tightness of adding nodes. [14] put forward an algorithm based on Optimization over Maximal Cliques (OMC). The algorithm extracted all the maximum cliques in the network as the initial communities and selected the direction of increasing the maximum value or decreasing the minimum of the module metric function EQ to unite the community at each step. Once united into a community, the algorithm stopped. In this process, the algorithm got a "dendrogram". [15] proposed an improved LPA algorithm named MCNLPA to detect overlapping community. The algorithm transformed the network into a maximum clique matrix and applied the LPA algorithm to the matrix. [16] proposed a complex network detection algorithm for detecting overlapping communities. The algorithm extracted the maximal cliques from the network and united them with the clustering coefficients of two adjacent maximal subgraphs. Meanwhile, the algorithm was able to detect overlapping nodes.

This paper presents an algorithm (k-clique clustering, KCC) for overlapping community discovery in complex networks. A community can be thought of as a collection of perfectly connected subgraphs (cliques). In this paper, the algorithm chooses a random node and takes the initial node as the initial community to expand. With each time expansion, the KCC algorithm finds a community neighbor node. With the maximum fitness value, the algorithm adds all k-cliques including this node to the community, during which the nodes with negative fitness are removed. Based on the community core of cliques, cliques are taken as the basic unit of the network. Since many nodes may belong to multiple cliques, the algorithm is able to find overlapping community structures.

2. Related Work

2.1. Basic Concepts

Graph $G = (V, E)$ can represent the network, where $V = \{v_1, v_2, v_3, \cdots, v_n\}$ represents node set and $E = \{e_1, e_2, e_3, \cdots, e_m\}$ represents edge set.

Palla [1] et al. argued that communities could be seen as a collection of completely connected subgraphs (cliques). K-clique indicated that the number of nodes in the fully subgraph was *k*. If there were two k-clique with *k*-1 shared nodes, then the two k-clique were considered adjacent. If we started from a k-clique and eventually reached another k-clique through a series of adjacent k-cliques, the two k-cliques were connected. A maximal subgraph consisting of k-cliques is the community. Since a node might belong to multiple k-cliques simultaneously, the CPM algorithm has the ability to detect overlapping community structures.

2.2. Fitness Function

Lancichinetti et al. proposed a fitness function that identifies local communities by maximizing fitness values as shown in Equation (1).

$$f_G = \frac{k_{in}^G}{\left(k_{in}^G + k_{out}^G\right)^\alpha} \tag{1}$$

Where k_{in}^G and k_{out}^G are the internal and external node degrees of community G respectively, and α is a positive real tunable parameter to control the size of a community. The larger the value of α, the smaller the community size. k_{in}^G represents the inner community degree that is equal to twice the number of edges in the community. k_{out}^G represents the external degrees, which is the number of edges that connect each member of the community to the rest of the module.

Equation (2) is given as a fitness function for a node A. The fitness for subgraph f_G^A is defined as the fitness difference value of the subgraph G with and without A, $G + \{A\}$ and $G - \{A\}$, which represent whether node A is included from the subgraph G [2].

$$f_G^A = f_{G+\{A\}} - f_{G-\{A\}} \tag{2}$$

3. KCC Algorithm

3.1. Algorithm Implementation

In Figure 1, the network has 9 nodes, and the current network consists of nodes {1, 2, 3, 4, 5, 6, 7, 8, 9}. The algorithm first finds all 3-cliques, which are {1, 2, 3}, {1, 2, 4}, {1, 2, 5}, {1, 3, 4}, {1, 3, 5}, {1, 4, 5}, {2, 3, 4}, {2, 3, 5}, {2, 4, 5}, {3, 4, 5}, {6, 7, 8}, {6, 8, 9} respectively. The algorithm randomly selects node 1. Node 3 has the largest fitness value among its neighbor nodes. All cliques that include node 3 are added to the community to form a new community {1, 2, 4, 5}. At this time, there is no node with a negative fitness value. Node 6 is randomly selected from the nodes that have not been visited. The above steps are repeated to get another community {6, 7, 8, 9}. The algorithm quickly found all the communities through pre-cliques.

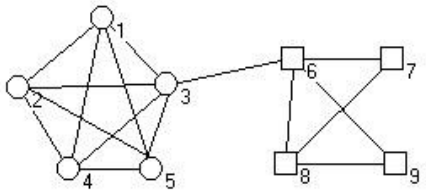


Figure 1. An example of community structures

The algorithm based on cliques is the core strategy of the community. Clique is the basic unit of the network. Since many nodes will belong to multiple cliques, the algorithm has the ability to identify overlapping community structures.

The main process of the algorithm is as follows:

- **Step 1** Find all k-cliques of *k* in the network;
- **Step 2** Select a subgraph *G*' containing node *A*;
- **Step 3** Find all neighbor nodes of *G*' and calculate the fitness function value of each neighborhood node of the subgraph *G*';
- **Step 4** Add the neighbor node with the largest fitness value to subgraph *G*', and add all k-cliques containing the neighborhood node to subgraph *G*' to form a larger subgraph *G*''; if it exists, any k-clique containing this node is selected.
- **Step 5** Recalculate the fitness of each node in *G*'';
- **Step 6** If the fitness of a node is negative, it will be removed from *G*'' to generate a new subgraph *G*''';
- **Step 7** If there is 6, repeat from 5; otherwise, repeat from step 3 with subgraph *G*' until all neighbor nodes in step 3 have negative fitness functions for subgraph *G*';
- **Step 8** Randomly select a node from the nodes that have not been visited, and return to step 2 to continue executing until all nodes in the graph are divided into at least one community.

Algorithm 1
Input: Graph *G*
Output: a community of the Graph *G*
(1) randomly select a node *A* of the graph to generate subgraph *G*';
(2) calculate the fitness values of all neighbor nodes in *G*' and find the neighbor node *v* with the largest fitness value;
(3) if the fitness value f(*v*) of node *v* is greater than or equal to 0
(4) if *v* exists in any k-clique
(5) add all these k-cliques into subgraph *G*';
(6) else
(7) add node *v* to *G*';
(8) While
(9) recalculate the fitness of the node in *G*';
(10) if there is node v_0 with a negative fitness value
(11) delete v_0 from subgraph *G*';
(12) continue;
(13) else
(14) return to step (2) to continue
(15) else
(16) return a community

In this paper, nodes are used as seed nodes rather than cliques because the clique is the initial community seed. The

algorithm finally needs to fuse similar communities, because some clique structures are similar with each other. Otherwise, if a single node is added to the community every time the community expands, there will be repeated calculation of the fitness value of nodes when obtaining the natural community, which is a huge amount of work. However, if the node is used as the seed node and a series of k-cliques are added each time, the efficiency of the algorithm is able to be improved.

3.2. Time Complexity

The time complexity of the KCC algorithm solves k-clique using a recursive algorithm. *n* is the number of network nodes.

Let the average maximum degree of nodes in the network be $\bar{k}\max$. The time complexity solves k-cliques by $O(n\times \bar{k}\max \text{^}2)$. Algorithm 1 has a worst-case time of $O(n^2 \log n)$.

3.3. Experimental Analysis

3.3.1. LFR Benchmark Network

The LFR reference networks, proposed by Lancichinetti et al., are a kind of artificial networks with real-world network characteristics. The networks are widely used in overlapping network community detection algorithms. The meaning of the parameters are in Table 1. Table 2 lists the parameters of the LFR reference networks in this experiment.

Table 1. The meaning of the parameters

N	the number of nodes
k	the average degree of the nodes
maxk	nodes' maximum degree
minc	nodes' minimum cluster size
maxc	nodes' maximum cluster size
*t*1	the exponent of the nodes degree distribution
*t*2	the exponent of the cluster size
mu	the mixed parameter
om	the number of communities to which the overlapping nodes belong
on	the number of nodes belonging to multiple communities

Table 2. LFR benchmark network parameters

Parameter	Network(a)	Network(b)	Network(c)	Network(d)	Network(e)	Network(f)	Network(g)	Network(h)
N	1000	1000	1000	1000	1000	1000	5000	5000
k	20	20	20	20	20	20	20	20
*t*1	2	2	2	2	2	2	2	2
*t*2	1	1	1	1	1	1	1	1
minc	10	10	20	20	10	20	10	20
maxc	50	50	100	100	50	100	50	100
maxk	50	50	50	50	50	50	50	50
om	2	2	2	2	1	1	1	1
on	0-700	0-700	0-700	0-700	0	0	0	0
mu	0.1	0.3	0.1	0.3	0.1-0.6	0.1-0.6	0.1-0.6	0.1-0.6

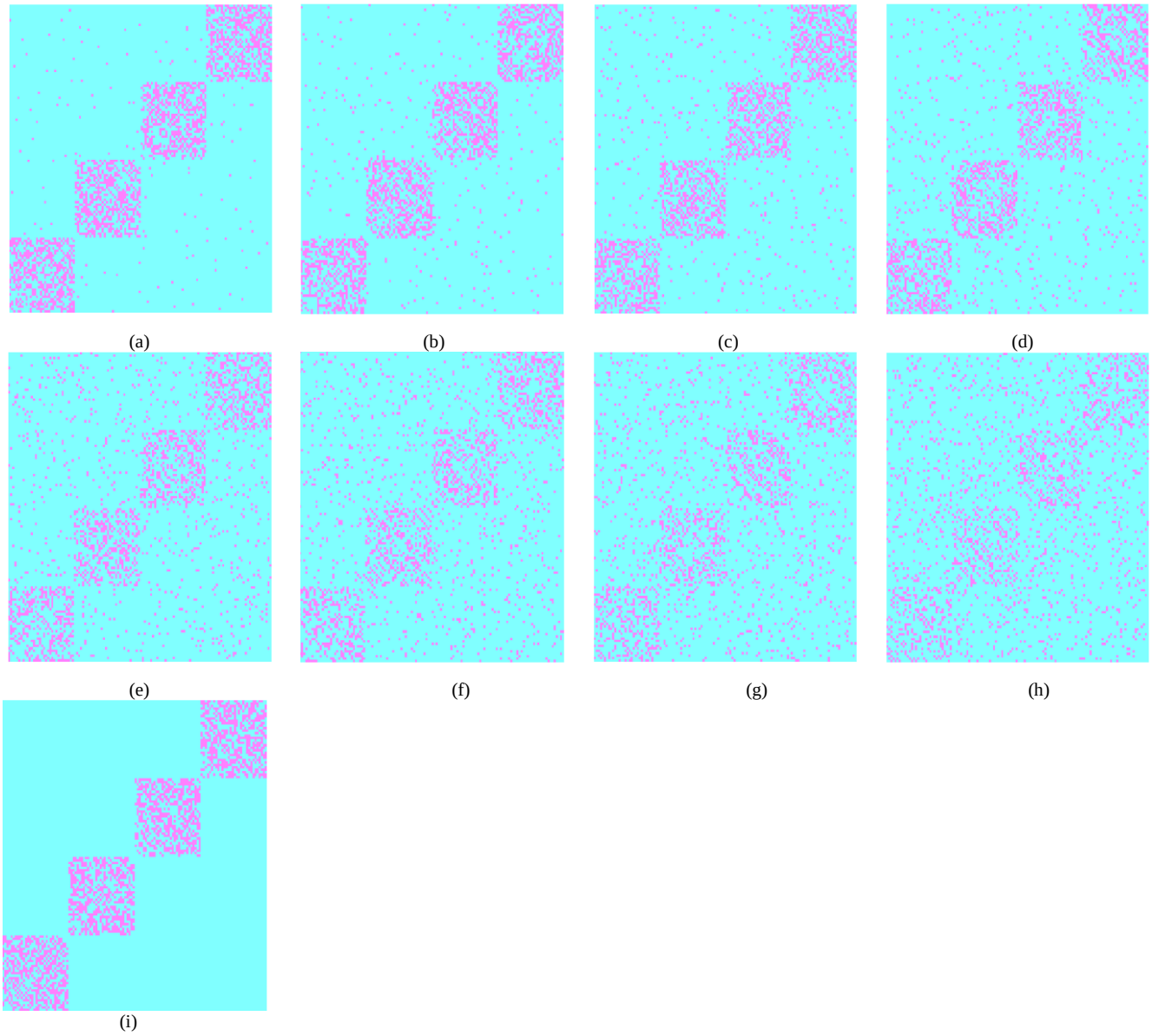
We tested our algorithms on different datasets of computer-generated networks and real-world networks, and compared it to other algorithms, including: GCE[9], SLPA[10], LINK[11], MCNLPA[15], ACC[16].

Lancichinetti et al. extended the (Normalized Mutual Information, NMI) standard mutual information to be able to detect the accuracy of overlapping communities [17].

$$NMI(x|y) = 1 - \frac{1}{2}\left[H(x|y)_{norm} + H(y|x)_{norm}\right] \tag{3}$$

Figure 2 shows the LFR computer-generated networks. The parameters are LFR (*N* = 128, *k* = 16, *maxk* = 16, *minc* = 32, *maxc* = 32, *mu* = 0 - 0.8, *t*1 = 2, *t*2 = 1).

Figure 3 shows the comparison of the experimental results of the algorithm. The *x*-axis represents *on*/*N*, *mu* and the *y*-axis represents NMI. The experimental data is the average of 20 runs through the algorithm. In the experiment, the tunable parameter α in the fitness function of the algorithm is set to 1, and the *k* value of the k-clique is set to 3 or 4. The algorithm performed well in practice.

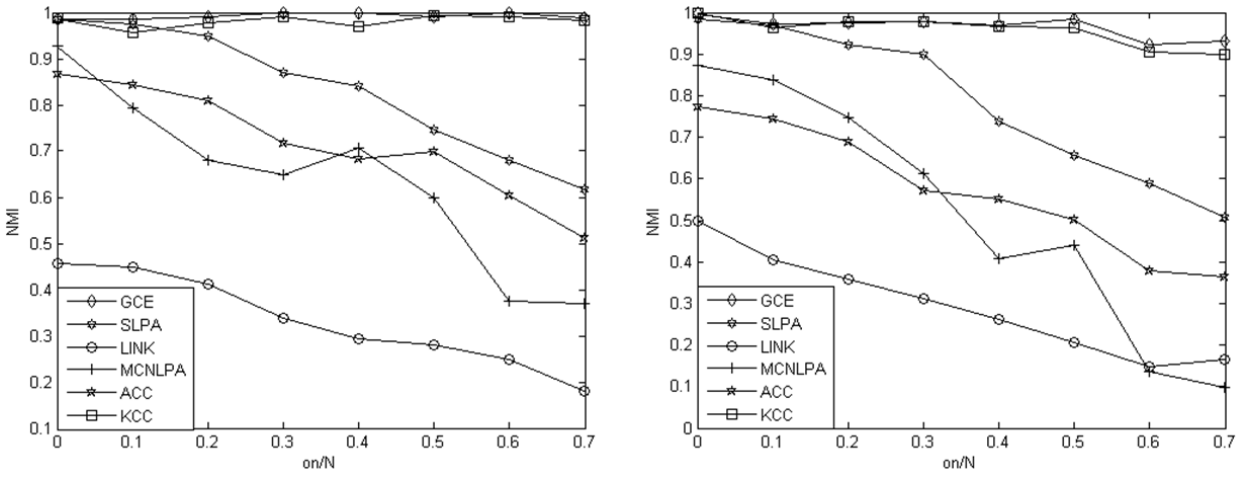


(a) (b) (c) (d)

(e) (f) (g) (h)

(i)

Figure 2. Heatmap of computer-generated networks



(a) (b)

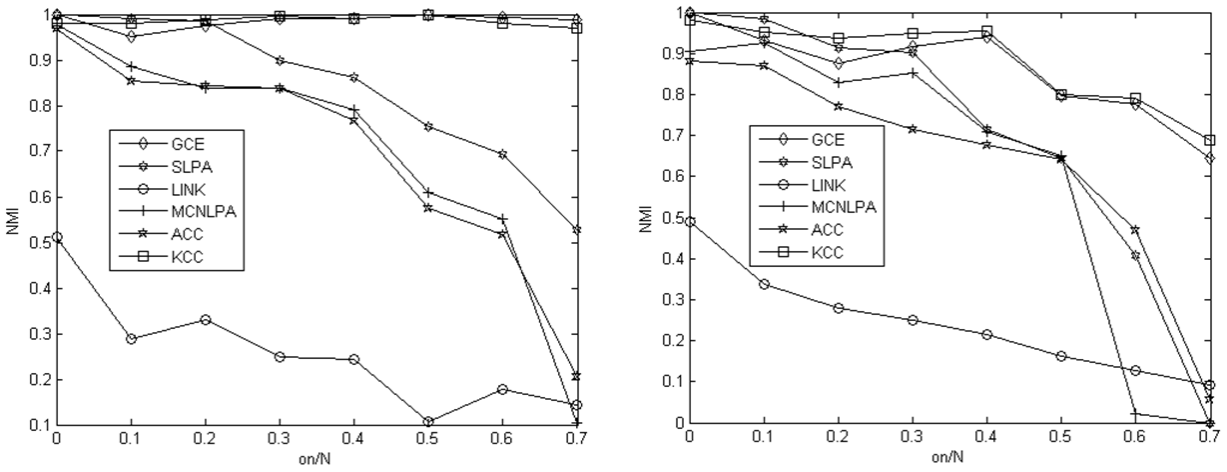


Figure 3. NMI for different algorithms of the benchmark network

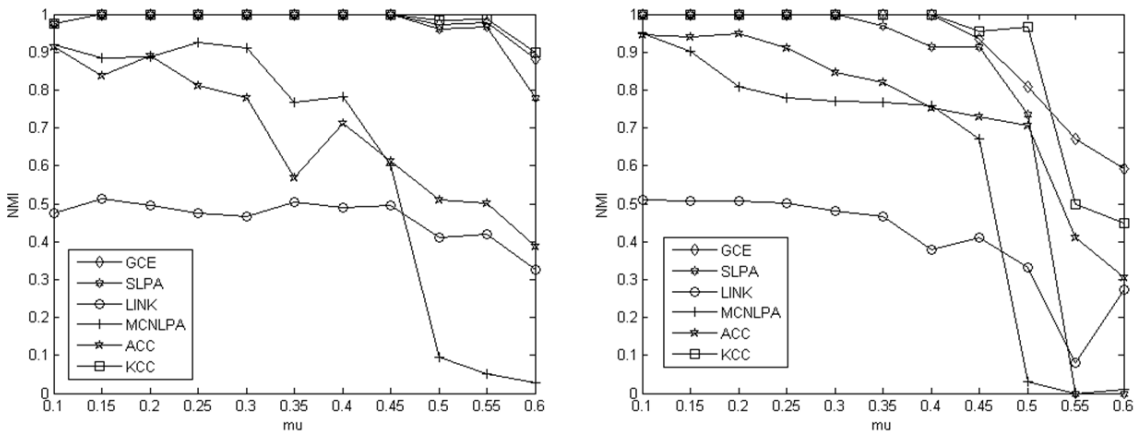
Beyond network parameters LFR (N = 1000, k = 20, $maxk$ = 50, $minc$ = 10, $maxc$ = 50, om = 2, on = 0-700, $t1$ = 2, $t2$ = 1) and mixed parameters mu = 0.1 and 0.3, we chose five other overlapping community detection algorithms to compare with the KCC algorithm, with an increase number of overlapping nodes in the total number of nodes in the network. The NMI value of the algorithm is shown in Figures 3(a) and (b). As seen from the figure, when the network structures become highly overlapped, the KCC algorithm achieves higher community detection accuracy with different mu values.

The community size distribution interval [$minc$, $maxc$] also has an impact on the community detection algorithm and network parameters. As the percentage of overlapping nodes in the total number of nodes in the network increases, the NMI value changes, as shown in Figures 3(c) and (d). With an increase in network distribution interval, the proposed algorithm has higher detection accuracy when compared to the other five algorithms.

The experiment also takes into account the influence of different network sizes and mixed parameters on the algorithm. The non-overlapping community detection accuracy results of the six algorithms in the experiment are shown in Figures 4(e), (f), (g) and (h). As can be seen from the figures, when the network size is small, the KCC algorithm can acquire higher community detection accuracy. With an increase of network size and an increase of mixed parameters, the KCC algorithm can also acquire higher community detection precision. The algorithm is similar to the GCE algorithm with high community detection accuracy.

3.3.2. Real World Network

We tested the algorithms on several real-world complex networks. The data used in the experiment are: Karate Club Network by Newman [18], Dolphin social network [19], American College football, Books about US politics, Jazz musicians network, and Email network of human interactions. Table 3 shows the actual network parameters used in the experiment. We used EQ as evaluation metric of real-world network.



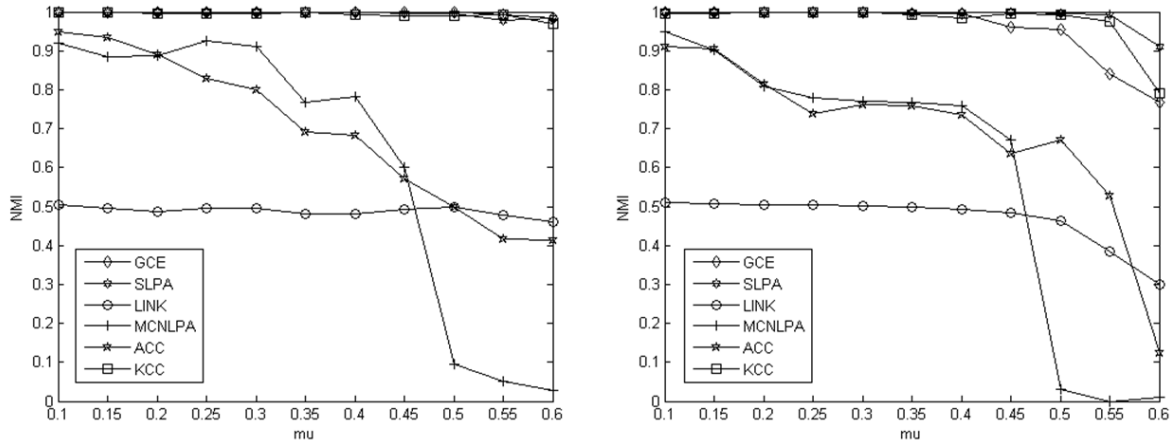


Figure 4. NMI for different algorithms of the benchmark network

The EQ function is an evaluation function of complex network overlapping communities [17]. A higher EQ represents a stronger overlapping community structure

$$EQ = \frac{1}{2m}\sum_{i}\sum_{v,w\in Ci}\frac{1}{O_v O_w}\left[A_{vw} - \frac{d_v d_w}{2m}\right] \tag{4}$$

Where d_v is the degree of the node v, O_v is the number of communities to which the node v belongs, and A is the adjacency matrix of the network.

$$m = \frac{1}{2}\sum_{vw} A_{vw}$$

Table 3. Real network for experiment

No	Network	Nodes	Edges	Communities
1	Zachary	34	78	2
2	Dolphins	62	159	2
3	Football	115	613	12
4	Political books	105	441	3
5	jazz	198	5484	
6	email	1133	5451	

As the real community structures of real networks is unknown, the EQ function is used in this paper as an evaluation metric of network clustering results. Table 4 shows the EQ of six algorithms on six real networks. It can be seen from the table that the algorithm in this paper achieves better overlapping community detection results on these kinds of real networks.

Table 4. EQ comparison of real world network algorithm results

Q-value	KCC	GCE	SLPA	LINK	MCNLPA	ACC
Zachary	0.3390	0.3771	0.3572	0.1336	0.3710	0.3836
Dolphins	0.3605	0.4661	0.4710	0.1111	0.1957	0.4881
Football	0.5908	0.5890	0.6005	0.0672	0.5094	0.6102
Political books	0.4973	0.4817	0.4652	0.0751	0.4978	0.5013
jazz	0.2921	0.2893	0.2815	0.047	0.2906	0.3015
email	0.4171	0.3841	0.4128	0.0338	0.4078	0.4247

The algorithm divides karate into two communities, where nodes 3, 9 and 10 are overlapping nodes. Figure 5 shows the results of the partitioning of the Karate network using the KCC algorithm. The algorithm can correctly identify overlapping community structures and overlapping nodes.

The algorithm divides Dolphins into two communities, where nodes Mus, Number and Osar are overlapping nodes. Figure 6 shows the results of the partitioning of the dolphin network using the KCC algorithm. Among them, only the node sn89 is divided into the wrong community.

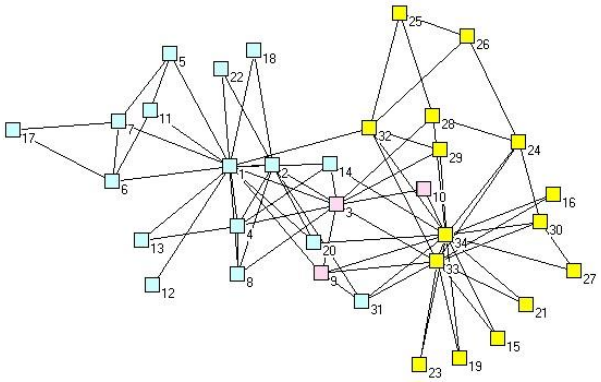


Figure 5. Communities structure of Karate obtained by GAC

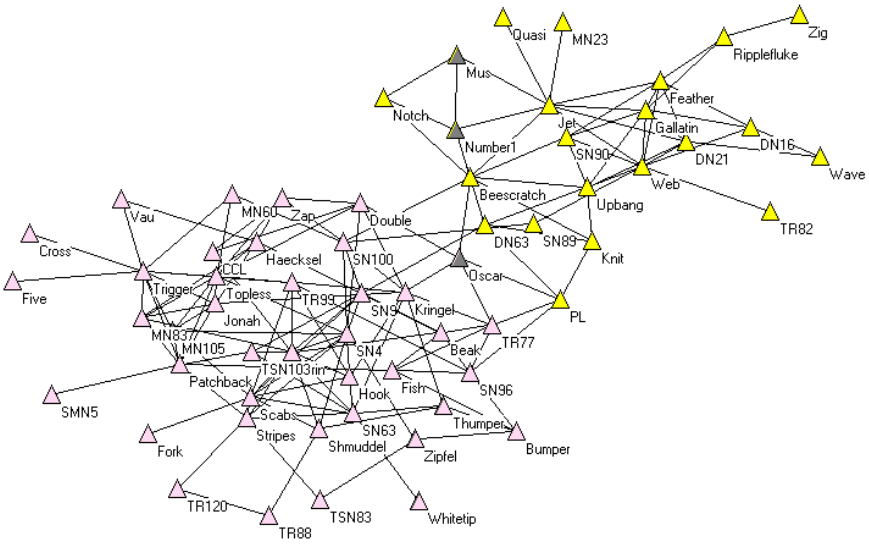


Figure 6. Communities structure of Dolphins obtained by KCC

The KCC algorithm has a high NMI value, which indicates that its community detection results are close to the real community structures. The algorithm is able to identify cliques in the network, which can be united cliques that are closely connected. The nodes located at the boundary that do not belong to the community are likely to be removed through the fitness function. The results are close to the real community structure and the time efficiency and complexity of the algorithm are also within the acceptable range.

The algorithm time efficiency and algorithm complexity are also within acceptable limits as seen in Figure 7. The computer-generated used LFR (N = 1000-5000, k = 20, $maxk$ = 50, $minc$ = 10, $maxc$ = 50, om = 2, on = 100, $t1$ = 2, $t2$ = 1).

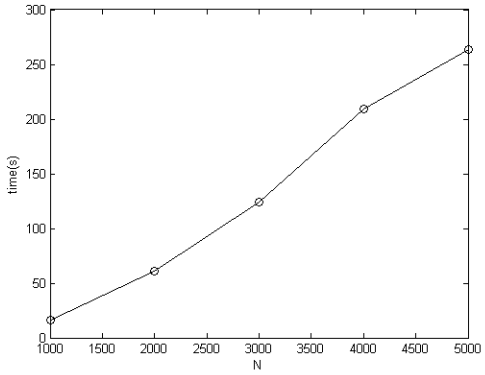


Figure 7. KCC algorithm runtime

4. Conclusions

This paper presents an algorithm for discovering overlapping communities in complex networks. The algorithm draws on the idea of cliques as the core of the initial community, and proposes to treat the overlapping community network as a collection of all k-cliques. The algorithm takes into account the topology of the network community and adds a series of k-cliques when the natural community expands. The overlapping nodes that are not part of the local community are removed from the community, namely, nodes with negative fitness values. Then, the division of the overlapping network community structures is executed. In computer-generated networks and real networks, the experiments show that based on the idea, the algorithm can achieve good results. In addition time efficiency and time complexity of the algorithm are all in acceptable ranges, also illustrating that the idea is feasible.

Acknowledgements

This work is supported by the National Key R&D Program of China under grant No.2016YFB1200100.

References

1. G. Palla, I. Derényi, I. Farkas, and T. Vicesk, "Uncovering the Overlapping Community Structures of Complex Networks in Nature and Society," *Nature*, Vol. 435, No. 7043, pp. 814-818, June 2005
2. A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the Overlapping and Hierarchical Community Structure in Complex Network," *New Journal of Physics*, Vol. 11, No. 3, pp. 033015, March 2009
3. E. Gregori, L. Lenzini, and S. Mainardi, "Parallel *k*-Clique Community Detection on Large-Scale Networks," *IEEE Transaction on Parallel and Distributed System*, Vol. 24, No. 8, pp. 1651-1660, August 2013
4. J. Yang, J. McAuley, and J. Leskovec, "Community Detection in Networks with Node Attributes," in *Proceedings of the 13th IEEE International Conference on Data Mining*, pp. 1151-1156, Dallas, USA, December 2013
5. F. Hao, S. S. Yau, G. Min, and L. T. Yang, "Detecting k-Balanced Trusted Cliques in Signed Social Networks," *IEEE Internet Computing*, Vol. 18, No. 2, pp. 24-31, March 2014
6. X. Wen, W. Chen, Y. Lin, T. Gu, H. Zhang, Y. Li, et al., "A Maximal Clique based Multiobjective Evolutionary Algorithm for Overlapping Community Detection," *IEEE Transactions on Evolutionary Computation*, Vol. 21, No. 3, pp. 363-377, September 2016
7. D. Jin, B. Gabrys, and J. Dang, "Combined Node and Link Partitions Method for Finding Overlapping Communities in Complex Networks," *Scientific Reports*, Vol. 5, pp. 8600, February 2015
8. C. Lee, F. Reid, A. McDaid, and N. Hurley, "Detecting Highly Overlapping Community Structure by Greedy Clique Expansion," (https://arxiv.org/abs/1002.1827)
9. J. Xie, B. K. Szymanski, and X. Liu, "SLPA: Uncovering Overlapping Communities in Social Networks via a Speaker-Listener Interaction Dynamic Process," in *Proceedings of the 11th International Conference on Data Mining Workshops*, pp. 344-349, Vancouver, CA, November 2011
10. Y. Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link Communities Reveal Multi-Scale Complexity in Networks," *Nature*, Vol. 466, No. 7307, pp. 761-764, October 2010
11. T. Zhang and B. Wu, "A Method for Local Community Detection by Finding Core Nodes," in *Proceedings of 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 1171-1176, Istanbul, Turkey, August 2012
12. X. Zhang, W. Zheng, C. Wang, Z. Ding, and Y. Su, "An Overlapping Community Detection Algorithm based on Addtion of a Clique at Each Step," *Journal of South China University of Technology* (*Natural Science Edition*), Vol. 44, No. 9, pp. 24-31, September 2016
13. Z. Huang, Z. Wang, and Z. Zhang, "Detecting Overlapping and Hierarchical Communities in Complex Network based on Maximal Cliques," in *Proceedings of the 6th Chinese National Conference on Social Media Processing*, pp. 184-191, Guangzhou, CHN, November 2015
14. P. Wu and L. Pan, "Detecting Highly Overlapping Community Structure based on Maximal Clique Networks," in *Proceeding of 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 196-199, Beijing, CHN, August 2014
15. Y. Cui, X. Wang, and J. Li, "Detecting Overlapping Communities in Networks using the Maximal Sub-graph and the Clustering Coefficient," *Physica A*: *Statistical Mechanics and its Applications*, Vol. 405, pp. 85-91, March 2014
16. V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the Definition of Modularity to Directed Graphs with Overlapping Communities," *Journal of Statistical Mechanics*: *Theory and Experiment*, Vol. 2009, No. 3, pp. 3166-3168, March 2009
17. H. Shen, X. Chen, K. Cai, and M. Hu, "Detect Overlapping and Hierarchical Community Structure in Networks," *Physica A*: *Statistical Mechanics and its Applications*, Vol. 388, No. 8, pp. 1706-1712, April 2009
18. W. W. Zachary, "An Information Flow Model for Conflict and Fission in Small Groups," *Journal of Anthropological Research*, Vol. 33, No. 4, pp. 452-473, 1977
19. D. Lusseau, "The Emergent Properties of a Dolphin Social Network," *Proceedings of the Royal Society B*: *Biological Sciences*, Vol. 270, No. Supl.2, pp. 186-188, 2003

Jian Ma is a PhD candidate at Beijing Jiaotong University. Her main research interests include data mining and complex network analysis.

Jianping Fan is an Adjunct Professor and PhD supervisor at Beijing Jiaotong University. His main research interests include computing, cloud computing, and parallel and distributed computing.