

Received December 7, 2019, accepted December 21, 2019, date of publication December 27, 2019, date of current version January 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962751

Local Optimization for Clique-Based Overlapping Community Detection in Complex Networks

JIAN MA¹ AND JIANPING FAN²

¹School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China

²Research Center for High-Speed Railway Network Management of Ministry of Education, Beijing Jiaotong University, Beijing 100044, China

Corresponding author: Jian Ma (13112083@bjtu.edu.cn)

This work was supported by the National Key R&D Program of China under Grant 2016YFB1200100.

ABSTRACT Detecting communities in complex networks has been one of the most popular research areas in recent years. There have been many community detection algorithms proposed to date. However, the local information (cliques) of communities and the search efficiency of algorithm have not been considered both in previous studies. In this paper, we propose a novel local expansion algorithm for detecting overlapping communities based on cliques. The algorithm draws on the assumption that cliques are the core of communities, as the clique takes into account the local characteristics of the community. The proposed algorithm adopts a single node with the maximum density as an initial community to prevent the formation of a large number of near-duplicate community structures, which improves the search efficiency of the algorithm. In many experiments using computer-generated and real-world networks, the proposed algorithm based on this idea verifies that the algorithm is able to detect overlapping communities effectively. The experiment yields better community uncover results, and the time efficiency and the complexity of algorithm are also satisfactory.

INDEX TERMS Complex network, community detection, overlapping community detection, local optimization, k-clique.

I. INTRODUCTION

Researchers are increasingly interested in the study of complex networks [1]. They are typically used to represent complex systems, such as in society, biology, computer and other fields. The World Wide Web, the Internet, the actor collaboration network, and communication networks are representative examples of complex networks [2], [3].

Community structure is an important topological property of complex networks. In a network, nodes are always clustered into different communities (clusters or groups), there is a widely accepted definition of the community structure, nodes in the same community are densely connected and links between nodes in different communities are sparse [4], [5]. There is another generally accepted view of a community, the community in a network is a group of similar nodes that differ from other nodes in the network. Nodes belonging to the same community can be classified according to their relationship in clusters. Nodes belonging to the same community may have more common characteristics than different

communities [6]–[8]. Research on the clustering of complex networks is not only great significance for understanding the functions of complex networks, but also for predicting the behavior in complex networks [9]. Clustering networks are widely used in personalized recommendation, protein function prediction and epidemic spreading [10].

A complex network consists of relatively independent communities that influence on each other. For example, a group of mutual acquaintances in a social network, a subset of web pages on the same topic, a compartment in a food network, a functional module in a protein interaction network, a biochemical pathway in a metabolic network, and so on [11]. The algorithms are: modularity-based optimization clustering [12], [13], label propagation partitioning [14], [15], partition-based algorithms [16], [17], and spectral methods [18], [19].

For community structure is one of the most important topological features of complex networks, in networks, which is characterized by densely linked internal nodes and sparse connection of different community nodes. Further studies have found that networks have many overlapping communities [20]–[22], for example, humans have a variety of social

The associate editor coordinating the review of this manuscript and approving it for publication was Longxiang Gao¹.

attributes, therefore, humans belong to different groups. Since then, many community detection algorithms have been developed, and the algorithms can find overlapping communities in which one node possibly appears in more than one community [23]. Communities can overlap in bipartite [24], [25] and weighted networks. The algorithms are: the clique percolation method [26], [27], link-based partitioning [28] and local extension based clustering [29]–[31].

In this paper, we propose a local optimization algorithm based on cliques (LOC) for overlapping community detection in complex networks. The proposed algorithm is an improvement to the Local Fitness Method (LFM) [29] and Clique Percolation Method (CPM) [26] algorithm. The algorithm picks a node with the highest density and focuses on a clique-based expansion strategy, and it is also based on local optimization function. The progress of clique-based locally optimized expansion can avoid repeated calculation of local optimization function and also fully considered the local characteristics of a community when acquiring natural communities, initializing the community with a node rather than cliques avoids excessively near-duplicate. Different from previous work, the proposed algorithm greatly improves the efficiency of the algorithm and the quality of the community division. The proposed algorithm is more suitable for large-scale and dynamic networks. The LOC allows communities to overlap, each node may appear in more than one community. Many nodes may belong to multiple cliques, thus, the LOC algorithm is able to detect overlapping community structures.

The remainder of this paper is organized as follows. Section II discusses related work, and Section III describes the proposed LOC algorithm. Section IV presents the experimental results of the LOC algorithm obtained on both computer-generated and real-world data sets, compares the clustering accuracy of the proposed algorithm to several overlapping community detection algorithms, and analyzed the proposed algorithm's time complexity. Conclusions and suggestions for future work are presented in Section V.

II. RELATED WORK

The number of communities in a network and their size are unknown in advance, which will be calculated by community detection algorithms [32].

The Speaker-Listener Label Propagation Algorithm (SLPA) [33], it is a speaker-listener-based information propagation process, in which a node acted as an information provider or consumer to hold labels in the stochastic process. A previous study [34] proposed the Label Propagation Algorithm based on the Maximal Clique Network detection (MCNLPA) to uncover highly overlapping communities. This algorithm regarded maximal cliques as the nodes of another network, and applied the LPA algorithm to this new matrix. The label propagation algorithm [33], [34] showed some uncertainty in label clustering. Another study [28] revealed pervasive overlapping communities and hierarchies by links rather than nodes. However, this algorithm emphasized community external links rather internal links, which results in

many small communities. In addition, the performance of this algorithm was not superior to other algorithms. A previous study [35]–[37] attempted to solve community detection problems using a matrix, however, the matrix operations were difficult for large-scale and dynamics networks.

In 2005, Palla *et al.* [26] proposed the CPM algorithm. The key idea of the CPM is that communities comprise overlapping complete sub-graphs. In the CPM, communities were interconnected through a series of adjacent k -cliques. Li *et al.* [38] introduced maximal cliques to uncover overlapping community structures. These algorithms [26], [27], [38] were more suitable for edge-intensive networks, spare parts would be inefficient, and the algorithm had no ability to assign nodes that were not part of any cliques. Cui *et al.* [39] also proposed an algorithm based on the Clustering Coefficient of two neighboring maximal sub-graphs (ACC) to detect overlapping communities, maximal cliques were extracted from a network and merged with the clustering coefficients of neighboring cliques. However, the built-in parameters of the methods [39]–[41] were difficult to determine. One study [42] put forward an algorithm based on Optimization over Maximal Cliques (OMC). The algorithm uncovered all maximum cliques in a network as initial communities, and it repeatedly joined communities in each step when chose the greatest increase in value of EQ function, The “dendrogram” corresponding to the largest modularity score is the most accurate partition. Unfortunately, the algorithm consumes a lot of time.

In 2009, Lancichinetti *et al.* [29] proposed the LFM algorithm. It presented a fitness function based on random seed and local optimization. However, The LFM might be inefficient because it calculated only one node at a time, and nodes were likely to be computed repeatedly. Eustace *et al.* introduced a neighborhood ratio to identify community size [43]–[45], Wang and Li [46] proposed the core-vertex to expand a community according to intimate extent, and Wang *et al.* proposed an overlapping communities method in dynamic social networks [47].

In 2010, Lee *et al.* [30] proposed the Greedy Clique Expansion algorithm (GCE). The algorithm found the largest clique as an initial seed, around these seeds, a fitness function was optimized locally using greedy search strategies. Cliques were near-duplicate in structure, so it was ultimately necessary to eliminate similar communities. Another [31] proposed a local community approach to perform community expansion. The method was feed with the local information of neighboring nodes as input, however, the nodes that did not satisfy the optimization function were not promptly handled.

III. LOC ALGORITHM

A. LOCAL DENSITY

Given an undirected and unweighted graph $G = (V, E)$, where $V = \{v_1, v_2, v_3, \dots, v_n\}$ is a node set, and $E = \{e_1, e_2, e_3, \dots, e_m\}$ is an edge set.

The main idea of the Density Peaks Clustering Algorithm (DPCA) [28] is to locate high-density regions separated

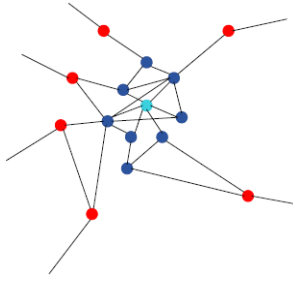


FIGURE 1. Example of fitness function [29].

by low-density areas. We reference the local density of the DPCA algorithm, the local density ρ_i of node i is defined as Equation (1):

$$\rho_i = \sum_j \chi(d_{ij} - d_c) \tag{1}$$

where, $\chi(x) = 1$ if $x \leq 0$, otherwise, $\chi(x) = 0$. d_c is a cutoff distance, and d_{ij} is the distance between nodes i and j .

Here, we take the shortest path length between nodes as distance d_{ij} , which signifies that the result is robust in terms of the selection of cutoff distance d_{ij} . The magnitude of cutoff distance d_c changes in different measurements. According to the literature [48], for simplicity, the cutoff distance d_{ij} is primarily set to constant $d_c = 1$ mostly in this study.

B. FITNESS FUNCTION

Lancichinetti *et al.* [29] proposed a fitness function in the LFM algorithm to identify local communities by maximizing the following function. The paper adopt the expression in Equation (2) as the optimization function.

Where respectively, k_{in}^G is the inner degrees of community G , which is equal to double the number of edges in community G . k_{out}^G is the external degrees of community G , it is the number of edges linking each community node with the rest of the modules, 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 is.

$$f_G = \frac{k_{in}^G}{(k_{in}^G + k_{out}^G)^\alpha} \tag{2}$$

In Equation (3), given a module fitness function of node A , the node fitness function in terms of a sub-graph f_G^A is defined as the difference of sub-graph G with and without A , where $G + \{A\}$ and $G - \{A\}$ represent whether node A is or is not included in sub-graph G , respectively [38].

$$f_G^A = f_{G+\{A\}} - f_{G-\{A\}} \tag{3}$$

In Figure 1, for the natural community extension of the sky blue node, the fitness values of all blue nodes are positive, $f(v) > 0$, and the fitness values of all red nodes are negative, that is, $f(v) < 0$.

C. LOC ALGORITHM DESCRIPTION

Palla *et al.* [26] argued that communities consists of a collection of completely connected sub-graphs (cliques). In the

CPM, communities are interconnected through a series of adjacent k-cliques. Since a node may belong to multiple k-cliques simultaneously, thus the LOC algorithm has the ability to uncover overlapping community structures.

The proposed LOC algorithm can be divided into two phases, k-clique recognition and natural community expansion. The first stage is to extract all k-cliques in a network. A k-clique is the complete sub-graph which the numbers of nodes are equal to k. The algorithm adopts a recursive algorithm to uncover k-cliques. Let nodes be the sequence $\{v_1, v_2, v_3, \dots, v_n\}$ of a network, and the procedure begins with set $\{v_1\}$. Then, detects the first backward node v_i linking to node v_1 in the preset order, then recursive this process from $\{v_1, v_i\}$, until the recursive termination condition is satisfied that there are k nodes in the collection. Then, the recursion is stepped back to check other possible k-cliques.

In the second phase, the algorithm selects a maximum local density node and takes the node as an initial community. The node with the highest node density is always located in the center area of the community, and other nodes in the network are easily involved. It also addresses unstable partition caused by the random seed selection strategy. The algorithm uses local optimization search based on the fitness function, and k-cliques are added each time when the natural community expands, rather than including only one single node. For expansion, the LOC algorithm finds the local maxima fitness value of neighboring nodes in a community, and the algorithm involves all k-cliques that include this node in the community. Note that the negative fitness value nodes are removed afterwards. This process is repeated until the fitness function value of the community no longer increases, the community is the ultimate one, select the maximum local density one that has never been traversed, until there is no node in the network that has not been visited.

The LOC algorithm considers a community as a collection of fully connected sub-graphs (cliques). First, a node is picked as the initial community, rather than a clique, because some cliques structures are near-duplicate, thus, it is eventually necessary to eliminate these similarly structured communities. The seed selection strategy can effectively simplify the community detection procedure. Adding cliques rather than a single node in the process of local optimization search can speed up convergence to an optimal process. Moreover, the community is extended by only a single node, which involves a huge number of the fitness function calculations. Obviously, the practice reduce the calculation time of the fitness function, which improves the efficiency of the proposed LOC algorithm.

$$\begin{pmatrix} 0 & 1 & 2 & 1 & 1 & 2 & 2 & 2 \\ 1 & 0 & 1 & 1 & 2 & 3 & 3 & 3 \\ 2 & 1 & 0 & 1 & 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 0 & 2 & 3 & 3 & 3 \\ 1 & 2 & 1 & 2 & 0 & 1 & 1 & 1 \\ 2 & 3 & 2 & 3 & 1 & 0 & 1 & 1 \\ 2 & 3 & 2 & 3 & 1 & 1 & 0 & 1 \\ 2 & 3 & 2 & 3 & 1 & 1 & 1 & 0 \end{pmatrix}$$

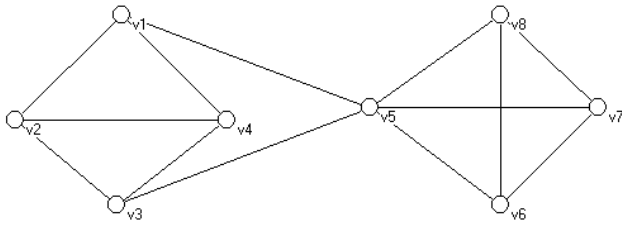


FIGURE 2. Example of community structures.

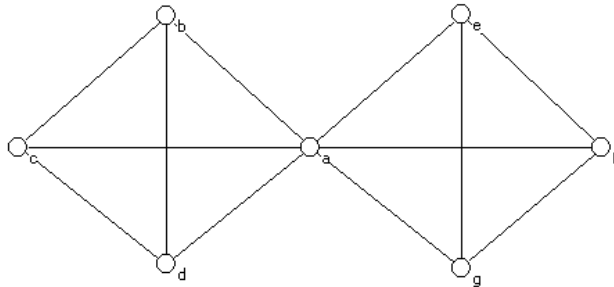


FIGURE 3. Example of overlapping community structures.

In Figure 2, the network has eight nodes, and the network consists of nodes $\{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$. $\{4, 4, 4, 4, 6, 4, 4, 4\}$ is the local density of the corresponding nodes. The above matrix is the shortest path length matrix for the network shown in Figure 2. The algorithm first finds all 3-cliques, and then the algorithm selects node v_5 with the maximum value as the initial community, the fitness value of nodes v_1, v_3, v_6 and v_8 is the maxima among neighboring nodes, pick v_1 for instance, all cliques which include node v_1 are $\{v_1, v_2, v_4\}$ with respect to the choice of v_1 , the community $\{v_1, v_2, v_4\}$ has no node with negative fitness value. They are involved in the community to form the natural community $\{v_1, v_2, v_4, v_5\}$. The fitness value of node v_3 is the maxima among adjacent nodes. The fitness value corresponding to $\{v_1, v_2, v_3, v_4, v_5\}$ is 0.824. However, the community fitness value composed of nodes v_1, v_2, v_3 and v_4 is 0.833, and node v_5 has a negative fitness value (-0.009), at this time, the community $\{v_1, v_2, v_3, v_4\}$ has no node with a negative fitness value. The algorithm excludes node v_5 from the community to get the correct community $\{v_1, v_2, v_3, v_4\}$. Node v_6 is randomly selected from the maximum node set which have not yet been accessed. Repeat the above steps to get another community $\{v_5, v_6, v_7, v_8\}$. The algorithm quickly uncover all proper communities through 3-cliques.

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 2 & 2 & 2 \\ 1 & 1 & 0 & 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 0 & 2 & 2 & 2 \\ 1 & 2 & 2 & 2 & 0 & 1 & 1 \\ 2 & 2 & 2 & 2 & 1 & 0 & 1 \\ 1 & 2 & 2 & 2 & 1 & 1 & 0 \end{pmatrix}$$

In Figure 3, the network has six nodes, and the network comprises nodes $\{a, b, c, d, e, f, g\}$, and $\{7, 4, 4, 4, 4, 4, 4\}$ is the corresponding node local density. The above matrix is the

shortest path length matrix for the network shown in Figure 3. The algorithm first detects all 3-cliques. The algorithm selects node a, which has the maximum value, and the fitness value of nodes b, d, e or f is the maxima among neighboring nodes. As random seed b, all cliques that include node b are involved in the community to form natural community $\{a, b, c, d\}$. Then the value of fitness function of the community is 0.8, simultaneously, there is no node with a negative fitness value. The fitness function of community $\{a, b, c, d, e\}$ or $\{a, b, c, d, g\}$ is 0.78. In other words, $f(e)$ or $f(g)$ are less than 0. Finally, node f is selected randomly from the nodes that have not been visited. The above steps are repeated to obtain the right community $\{a, e, f, g\}$. The algorithm also discovers overlapping communities, and the overlapping node is node a.

Nodes with the maxima fitness value are selected iteratively. All k-cliques with this node are included to the same community. Subsequently, nodes with negative fitness values are excluded to partition network community structures and detect overlapping nodes. The algorithm emphasizes the built-in parameters as k-cliques of k, and avoids random seed strategy. Since many nodes belong to more than one clique, the algorithm has the ability to uncover natural overlapping groups.

D. FLOW OF LOC ALGORITHM

The flow of the proposed LOC algorithm is summarized as follow:

1. Extract all k-cliques of network G;
2. Rank nodes by the local density in descending order;
3. Pick sub-graph G' including node A with the maximum local density value;
4. The fitness function value of each neighboring node around sub-graph G' is calculated using Equation (3);
5. If the maxima fitness of a neighboring node is greater than or equal to 0, all k-cliques including the node are added to sub-graph G', yielding a larger sub-graph G''. If such a k-clique does not exist, only the target node is added;
6. Recalculate the fitness function of node in sub-graph G'';
7. If there is a node with a negative fitness, it is removed from sub-graph G''. Generate a new sub-graph G''';
8. If step 7 occurs, go to step 6, else, go to step 4;
9. Pick a node with the maximum fitness that has not been assigned, and go to step 3. This process is looped until all nodes in the graph are assigned to at least one community.

Figure 4 shows the flowchart of the proposed LOC algorithm. The algorithm flow is vividly shown in this figure.

E. TIME COMPLEXITY

It is difficult to uniformly represent the time complexity of the algorithm, because it depends on the extent of community overlap and the community size.

Let n be the number of network nodes, and \bar{k} max be the average degree of the nodes in a network. The k-clique

Algorithm 1 Main Procedures of LOC Algorithm

Input: Graph G

Output: communities of the Graph G

```

1.  $G' \leftarrow A$ ; /* node  $A$  is the maximum value of local density*/
2.  $v \leftarrow$  the node with the maxima fitness value of all neighbors of  $G'$ ; /*calculate the fitness values of all neighbor nodes in  $G'$ , and find the neighboring node  $v$  with the maxima fitness value*/
3. if  $f(v) \geq 0$  /* 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.      $G'.add(k\text{-cliques})$ ; /*add all these  $k$ -cliques into sub-graph  $G'$ */
6.   else
7.      $G'.add(v)$ ; /*add node  $v$  to  $G'$ */
8.   end if
9. while
10.   recalculate the fitness of the node in  $G'$ ;
11.   if  $f(v_0) < 0$  /*there is node  $v_0$  with a negative fitness value*/
12.      $G'.delete(v_0)$ ; /*delete  $v_0$  from sub-graph  $G'$ */
13.     continue;
14.   else
15.     go to step 2; /*go to step (2) to loop*/
16.   end if
17. end while
18. else
19.   return  $G$ ; /*return  $G$  as the outcome*/
20. end if

```

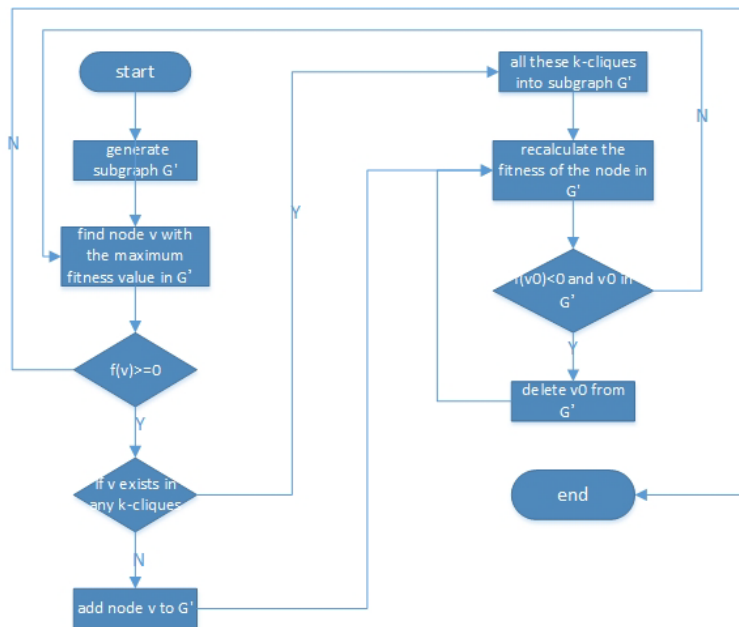


FIGURE 4. Flow of LOC algorithm.

recognition phase uses a recursive algorithm, and the worst-case time complexity of computing k -cliques scales as $O(n^2)$.

For sparse community structures, the second procedure of the LOC algorithm, the algorithm has a worst-case computational complexity of $O(n^2 \log n)$. For dense communities with high overlap, let the number of communities be c , and the time complexity is approximately as $O(c^2 \log c)$.

The time complexity of other comparison algorithms for comparison, the SLPA algorithm, the time complexity scales as $O(T^*m)$, where T is the defined maximum number of iterations, and m is the total number of edges. The time complexity of the GCE algorithm depends on community size and the degree of community overlap. For the MCNLPA algorithm, the maximal cliques stage is an NP-complete problem, and

TABLE 1. The meaning of parameters.

| Parameter | Meaning |
|-----------|---|
| N | the number of nodes |
| k | the average degree of nodes |
| $maxk$ | nodes' maximum degree |
| $minc$ | nodes' minimum cluster sizes |
| $maxc$ | nodes' maximum cluster sizes |
| t_1 | the exponent of nodes degree distribution |
| t_2 | the exponent of cluster sizes |
| mu | the mixed parameter |
| om | the number of communities to which overlapping nodes belong |
| on | the number of nodes which belongs to multiple communities |

the remaining time complexity is nearly linear. In addition, the complexity time of the ACC algorithm, the computing time is at most $O(n^2)$.

IV. EXPERIMENTAL ANALYSIS

The CPU is Intel(R) i7 @3.4GHz with 8GB Memory running Windows 10 operating system; Python version is 2.7.13 and MATLAB version is 7.0.

A. LFR BENCHMARK NETWORK

LFR benchmark networks proposed by Lancichinetti *et al.* [49] are a type of computer-generated networks with predefined tunable parameters. The networks possess real-world characteristics and are widely applied in various overlapping network community detection algorithms. The network parameters are illustrated in Table 1. Table 2 and 3 shows the parameters settings for the LFR benchmark networks in this experiment.

Figure 5 shows the heatmaps of the LFR computer-generated networks. The parameters are LFR: the network size $N = 128$, $k = 16$, maximum degree of nodes $maxk = 16$, $[minc, maxc] = 32$, mu value ranges from 0 to 0.8, and t_1, t_2 equals to 2 and 1. With the increase of mixed parameters mu . The figures from obvious clustering to impossibly distinguishable partitions. The network has 128 nodes and it is divided evenly into four communities. Here, a node has z_{in} links connecting to nodes in the same community and locates an average z_{out} links to nodes in other communities. The total degree of a node is $z = z_{in} + z_{out} = 16$. The value of z_{out} is increased from 0 to 8. Figure 5 shows the heatmaps of z_{out} from 0 to 8. Figure 5(a) is heatmap for $z_{out} = 0$. Figure 5(i) presents $z_{out} = 8$.

Lancichinetti *et al.* [29] extended the Normalized Mutual Information (NMI) as an evaluation index for overlapping community detection. The NMI can be used to effectively evaluate the accuracy of community partition. NMI values ranges from 0 to 1; the greater the NMI value, the better the partition result.

$$NMI(x|y) = 1 - \frac{1}{2} [H(x|y)_{norm} + H(y|x)_{norm}] \quad (4)$$

We analyze the proposed LOC algorithms experimentally on various datasets of artificial networks and real-world networks, and we empirically compared with several

TABLE 2. LFR benchmark of non-overlapping network parameters.

| Parameter | Network(a) | Network(b) | Network(c) | Network(d) |
|-----------|------------|------------|------------|------------|
| N | 1000 | 1000 | 1000 | 1000 |
| k | 20 | 20 | 20 | 20 |
| $maxk$ | 50 | 50 | 50 | 50 |
| $t1$ | 2 | 2 | 2 | 2 |
| $t2$ | 1 | 1 | 1 | 1 |
| $minc$ | 10 | 10 | 20 | 20 |
| $maxc$ | 50 | 50 | 100 | 100 |
| om | 2 | 2 | 2 | 2 |
| on | 0-700 | 0-700 | 0-700 | 0-700 |
| mu | 0.1 | 0.3 | 0.1 | 0.3 |

TABLE 3. LFR benchmark of non-overlapping network parameters.

| Parameter | Network(a) | Network(b) | Network(c) | Network(d) |
|-----------|------------|------------|------------|------------|
| N | 1000 | 1000 | 5000 | 5000 |
| k | 20 | 20 | 20 | 20 |
| $maxk$ | 50 | 50 | 50 | 50 |
| $t1$ | 2 | 2 | 2 | 2 |
| $t2$ | 1 | 1 | 1 | 1 |
| $minc$ | 10 | 20 | 10 | 20 |
| $maxc$ | 50 | 100 | 50 | 100 |
| om | 1 | 1 | 1 | 1 |
| on | 0 | 0 | 0 | 0 |
| mu | 0.1-0.6 | 0.1-0.6 | 0.1-0.6 | 0.1-0.6 |

state-of-the-art algorithms: LINK [28], GCE [30], SLPA [33], MCNLPA [34], MC [38] and ACC [39].

For the GCE algorithm, the key parameter k is 3, the built-in parameter α is 1 and ε is 0.6. For the SLPA algorithm, we record the optimal threshold ranges 0.1-0.45, and an interval is 0.05. For the LINK algorithm, the built-in parameter varies from 0.2-0.7, and with an interval value 0.1.

Figure 6 and 7 are the comparisons of experimental outcomes of the proposed LOC algorithm. The X axis represents on/N or mu , the Y axis represents NMI. In the experiment, the resolution parameter α in Equation (2) is tuned to 1 in the proposed algorithm, the k value in the k-clique is assigned to 3 or 4, the algorithm get comparative performance.

In Figure 6(a) and (b), the parameters of networks are LFR: the network size $N = 1000$, the average degree $k = 20$, $maxk = 50$, and the exponent of nodes degree distribution t_1 and the exponent of cluster sizes t_2 equals to 2 and 1. $om = 2$, on ranges from 0 to 700, $[minc, maxc] = [10, 50]$, and the mixed parameter $mu = 0.1$ and 0.3, Experiment evaluates the effect of the LOC algorithm and compares it with state-of-the-art overlapping community detection algorithms in terms of NMI, as the number of overlapping nodes in the network increases. The performance of the algorithm is shown in Figure 6(a) and (b). As can be seen from the figures, the proposed LOC algorithm demonstrates higher community detection accuracy with different mu values. The LOC is close to the real community structures and the performance obviously exceeds its counterparts except for the GCE algorithm,

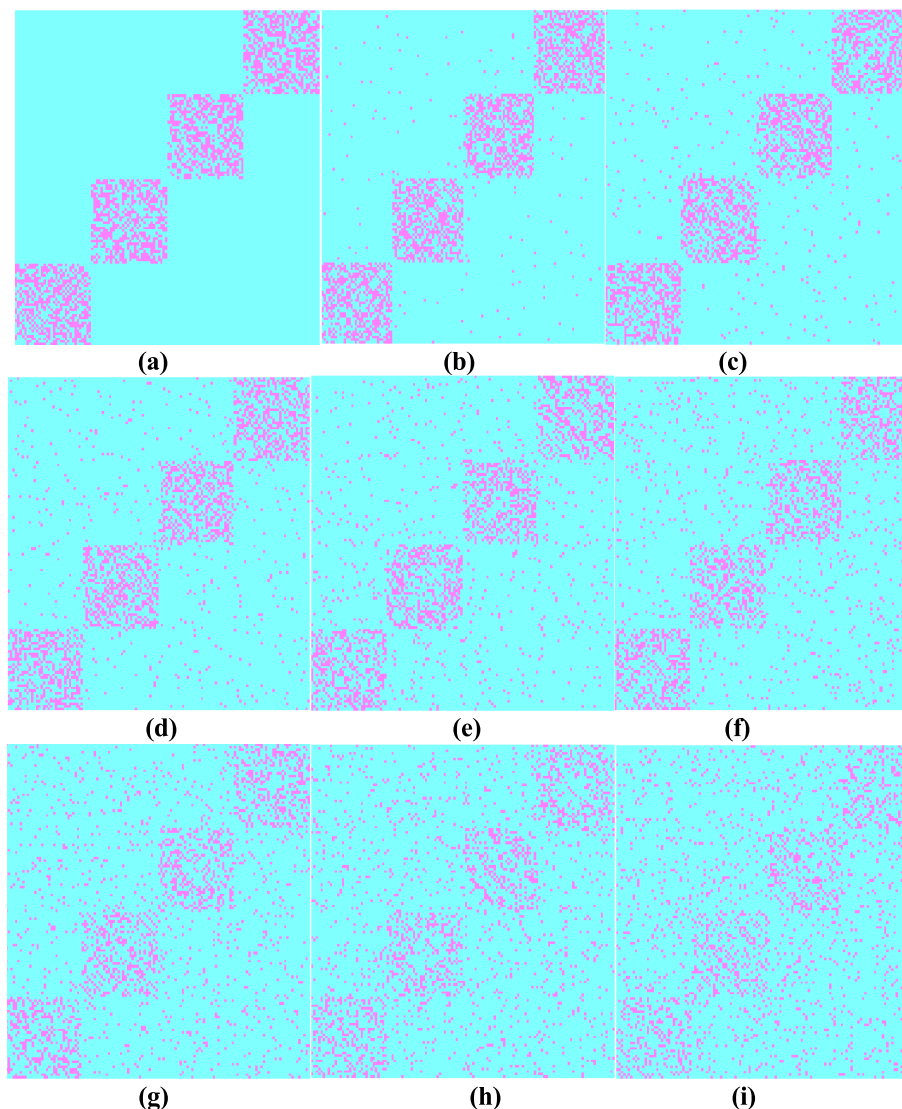


FIGURE 5. Heatmaps of LFR ($N = 128, k = 16, maxk = 16, minc = 32, maxc = 32, \mu = 0 - 0.8, t_1 = 2, t_2 = 1$).

as the network structures become highly overlapped, the other algorithm has significantly decreased scores as overlapping nodes increase in the network. In Figure 6(a), the algorithm correctly identified almost the nodes, the fraction of correction is 98.55% when on is 0, an average of 98.2%, of nodes be revealed when on equals 700.

The community size distribution interval $[minc, maxc]$ also has an influence on various community detection algorithms, the network parameters are LFR: the network size $N = 1000$, the average degree $k = 20$, $maxk = 50$, and the exponent of nodes degree distribution t_1 and the exponent of cluster sizes t_2 equals to 2 and 1, on ranges from 0 to 700, $om = 2$, $[minc, maxc]$ is $[20, 100]$, and mixed parameter μ is 0.1 and 0.3, the percentage of overlapping nodes in the total number of nodes of the networks increases, the outcome scores of the algorithm are presented in Figure 6(c) and (d), As the network distribution interval increases. Compared to the other algorithms, the proposed algorithm shows slight fluctuation

as the number of overlapping nodes increases. A large community size distribution interval means a great community magnitude. Others performances has less ability to uncover the accurate community besides the GCE algorithm. In Figure 6(c), the correct proportion is 97% when the value of on is 700, the fraction fail to only 69% in Figure 6(d) counterpart.

The experiment also considers the impact of different network scales and mixed parameter on the algorithm's performance. The non-overlapping community detection accuracy results of the algorithms along with the increasing mixed parameter and network scale in the algorithms are illustrated in Figure 7(a), (b), (c) and (d), a larger mixed parameter value implies that the algorithm become more different to identify the actual community whatever the algorithm is. It can be concluded from the figures that when network sizes are small and the mixed parameters are increasing, the proposed LOC algorithm demonstrates better community

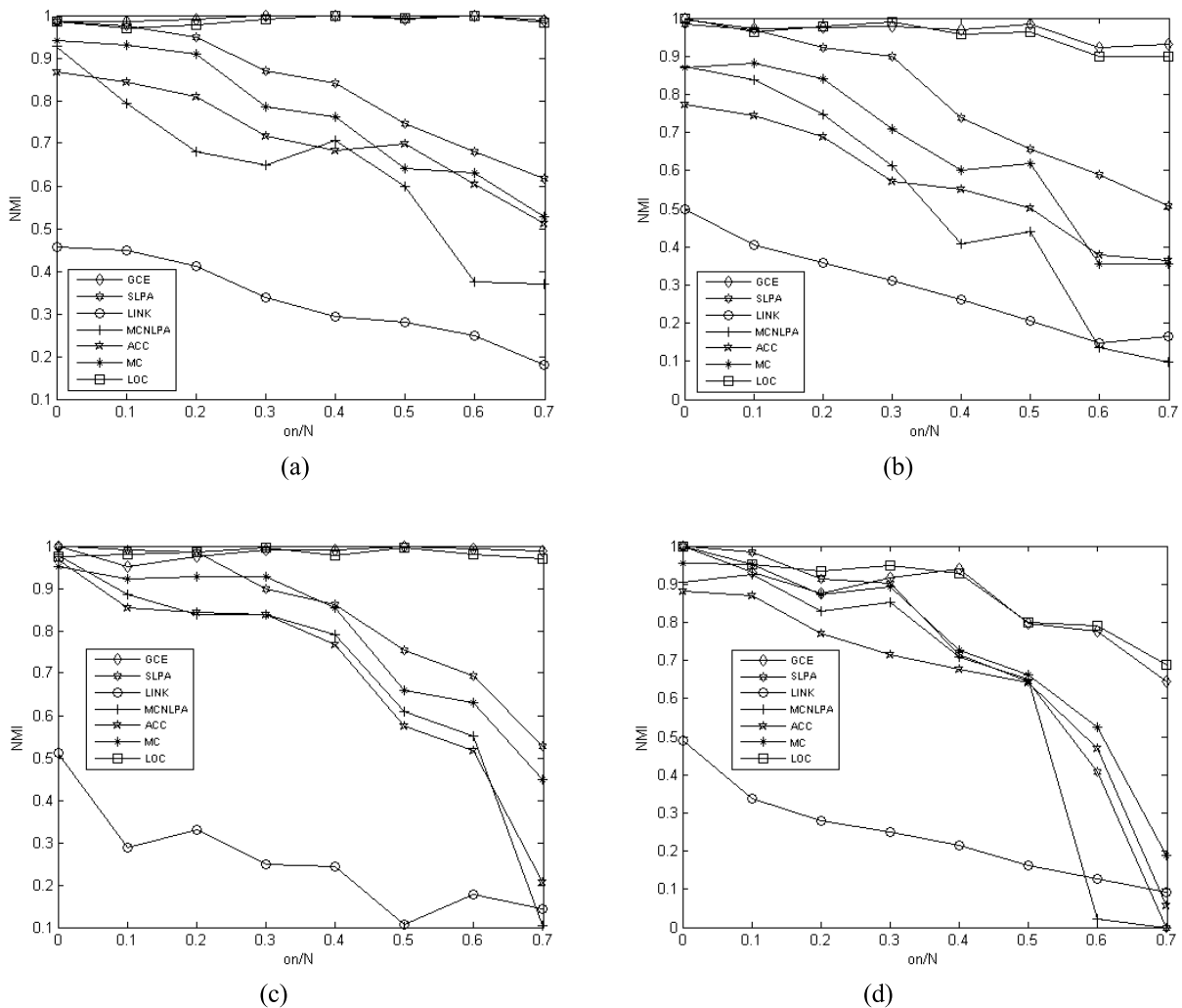


FIGURE 6. NMI for different algorithms of overlapping LFR benchmark networks. (a) LFR ($N = 1000, k = 20, maxk = 50, minc = 10, maxc = 50, \mu = 0.1, t_1 = 2, t_2 = 1, om = 2, on = 0 - 700$). (b) LFR ($N = 1000, k = 20, maxk = 50, minc = 10, maxc = 50, \mu = 0.3, t_1 = 2, t_2 = 1, om = 2, on = 0 - 700$). (c) LFR ($N = 1000, k = 20, maxk = 50, minc = 20, maxc = 100, \mu = 0.1, t_1 = 2, t_2 = 1, om = 2, on = 0 - 700$) (d) LFR ($N = 1000, k = 20, maxk = 50, minc = 20, maxc = 100, \mu = 0.3, t_1 = 2, t_2 = 1, om = 2, on = 0 - 700$).

detection performance. In addition, with the increasing network scale, the proposed LOC algorithm also demonstrates better community detection precision. The outcomes of the LOC algorithm are similar to those of the GCE algorithm, both of them has higher partition quality. Obviously, Experiment other results investigate the NMI scores drop quickly than the LOC algorithm as the community become more obscure.

B. REAR-WORLD NETWORK

The first data set is the Karate Club network, which has 34 vertices and 78 edges. This dataset represents the community structure of a Karate Club at an American university. After two years, the club was divided into two factions due to disputes among its members. Figure 8 shows the outcome of the LOC algorithm. The second case is Dolphin data. Lassran observed 62 bottlenose dolphins between 1994 and 2001. During this time, as a dolphin left, the dolphin society was

divided into two parts. The third one we consider is college football net for a regular season of 2000, where each node represents a football team, and if two teams play, there is a connection between them. Each team belongs to a conference (community). A team always plays more games in its own meetings and fewer games between meetings.

We evaluated the algorithms on 10 real-world networks from different fields. The data adopted in the experiment are: Karate Club Network by Newman [50], Dolphin social network [51], American College football, US politics books, Jazz musicians, human interactions about email, Power, Astrophysics collaborations, High-energy theory collaborations and Internet [52]. Table 4 gives the network parameters in the experiment. The metrics helps to measure complex network clustering are described in the literatures [53]–[55].

EQ [53] is the metric for evaluation of complex communities that extends Newman’s definition of modularity, and we

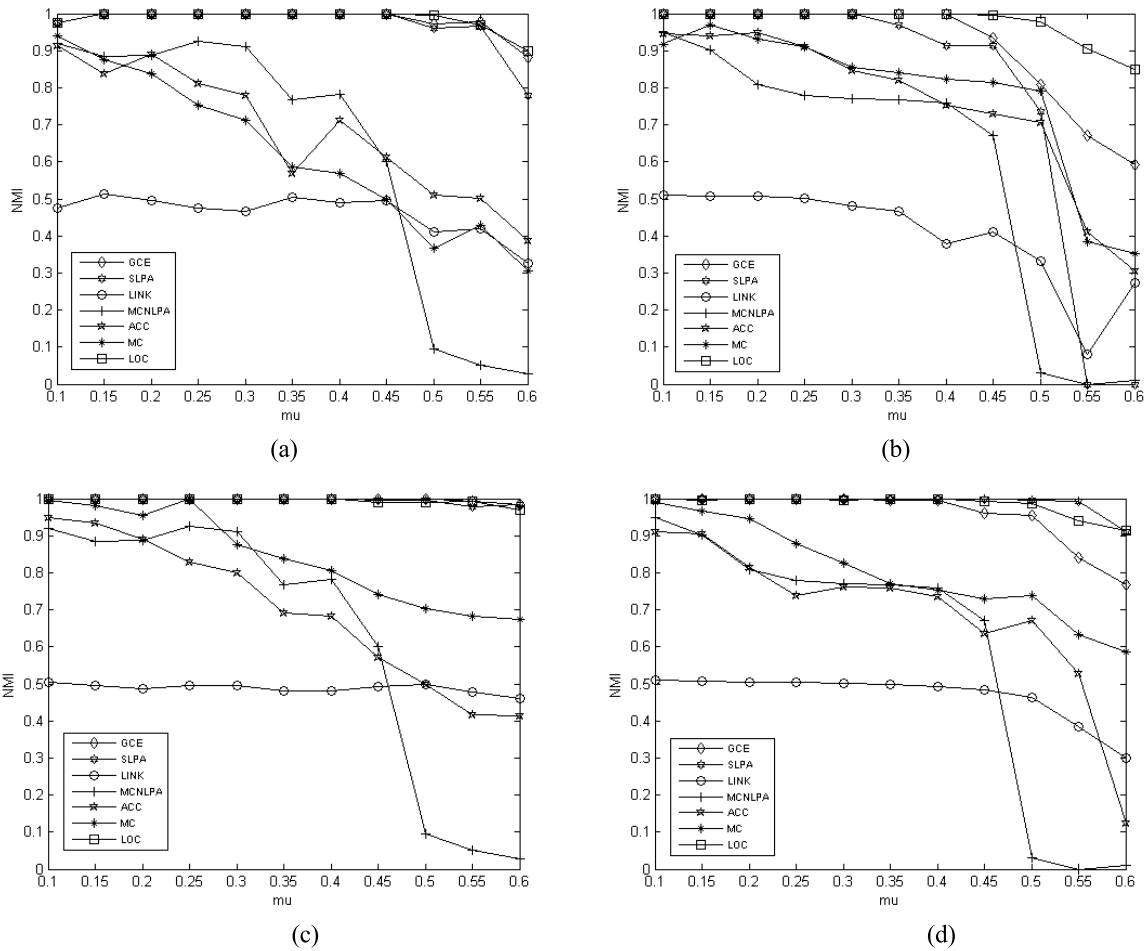


FIGURE 7. NMI for different algorithms of non-overlapping LFR benchmark networks. (a) LFR ($N = 1000, k = 20, maxk = 50, minc = 10, maxc = 50, \mu = 0.1 - 0.6, t_1 = 2, t_2 = 1, om = 1, on = 0$). (b) LFR ($N = 1000, k = 20, maxk = 50, minc = 20, maxc = 100, \mu = 0.1 - 0.6, t_1 = 2, t_2 = 1, om = 1, on = 0$). (c) LFR ($N = 5000, k = 20, maxk = 50, minc = 10, maxc = 50, \mu = 0.1 - 0.6, t_1 = 2, t_2 = 1, om = 1, on = 0$). (d) LFR ($N = 5000, k = 20, maxk = 50, minc = 20, maxc = 100, \mu = 0.1 - 0.6, t_1 = 2, t_2 = 1, om = 1, on = 0$).

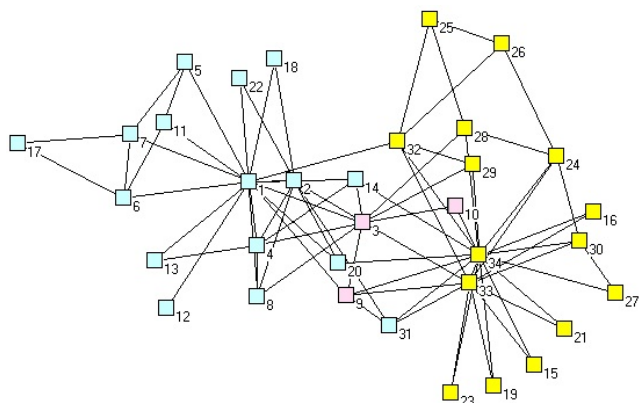


FIGURE 8. Community structures of the Karate obtained by LOC algorithm.

express the modularity as follows: [32]

$$EQ = \frac{1}{2m} \sum_i \sum_{v,w \in C_i} \frac{1}{O_v O_w} \left[A_{vw} - \frac{d_v d_w}{2m} \right] \quad (5)$$

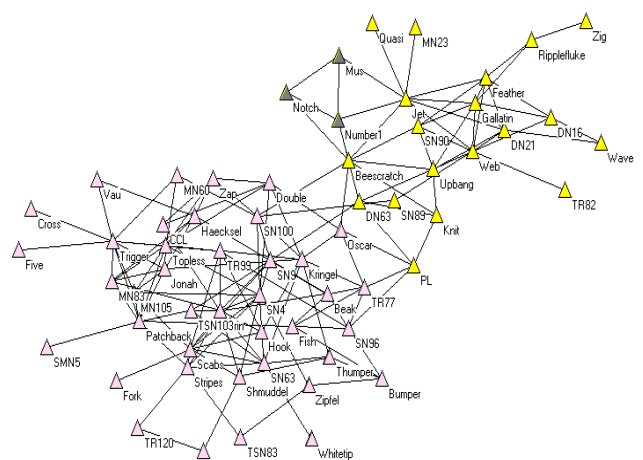


FIGURE 9. Community structures of the Dolphins obtained by LOC algorithm.

where O_v is the number of communities to which node v belongs, A is the adjacency matrix of a network, $m = \frac{1}{2} \sum_{v,w} A_{vw}$, and d_v is the degree of node v ,

TABLE 4. Real-world networks.

| No | Network | Nodes | Edges | $\langle k \rangle$ | Communities |
|----|---------------------|---------|---------|---------------------|-------------|
| 1 | Zachary's club [50] | 34 | 78 | 4.59 | 2 |
| 2 | Dolphins[51] | 62 | 159 | 5.13 | 2 |
| 3 | Football[52] | 115 | 613 | 10.66 | 12 |
| 4 | Political books | 105 | 441 | 8.40 | 3 |
| 5 | Jazz | 198 | 5484 | 27.7 | 4 |
| 6 | Email | 1133 | 5451 | 9.62 | 11 |
| 7 | Power | 4,941 | 6,594 | 2.67 | - |
| 8 | Astro-Ph | 14,845 | 119,652 | 16.12 | - |
| 9 | Hep-Th | 27,400 | 352,021 | 25.70 | - |
| 10 | Internet | 124,651 | 193,620 | 3.11 | - |

TABLE 5. EQ comparison of real-world network algorithm results.

| Q-value | LOC | GCE | SLPA | LINK | MCNLPA | ACC | MC |
|-----------------|-------------|-------------|-------------|-------------|-------------|--------------|-------------|
| Zachary's club | 0.3390/2 | 0.3771/4 | 0.3572/3 | 0.1336/22 | 0.3710/4 | 0.3836/3 | 0.3783/4 |
| Dolphins | 0.3605/3 | 0.4661/7 | 0.4710/5 | 0.1111/66 | 0.1957/2 | 0.4881/3 | 0.4744/4 |
| Football | 0.5908/10 | 0.5890/12 | 0.6005/10 | 0.0672/158 | 0.5094/5 | 0.6102/27 | 0.6013/28 |
| Political books | 0.4980/4 | 0.4817/4 | 0.4652/2 | 0.0751/126 | 0.4978/10 | 0.5013/5 | 0.5020/4 |
| Jazz | 0.3039/3 | 0.2893/2 | 0.2815/3 | 0.047/223 | 0.2906/8 | 0.3015/2 | 0.2908/2 |
| Email | 0.4281/13 | 0.3841/19 | 0.4128/16 | 0.0338/3132 | 0.4078/17 | 0.4247/13 | 0.4137/14 |
| Power | 0.5579/226 | 0.4590/226 | 0.5452/603 | 0.0875/1028 | 0.3079/326 | 0.3452/503 | 0.3875/528 |
| Astro-Ph | 0.5676871 | 0.5166/1200 | 0.5516/1010 | 0.1609/884 | 0.5166/1086 | 0.5516/1165 | 0.5609/1514 |
| Hep-Th | 0.5074/942 | 0.5034/5102 | 0.4719/1273 | 0.1631/783 | 0.4934/542 | 0.4719/673 | 0.4531/783 |
| Internet | 0.3648/3910 | 0.3648/3970 | 0.3606/4221 | 0.2729/4380 | 0.3148/4210 | 0.3606/43121 | 0.3629/5380 |

TABLE 6. Qa comparison of real-world network algorithm results.

| Q-value | LOC | GCE | SLPA | LINK | MCNLPA | ACC | MC |
|-----------------|-------------|-------------|--------------|-------------|-------------|--------------|-------------|
| Zachary's club | 0.4079/2 | 0.4215/4 | 0.4935/3 | 0.2103/22 | 0.4273/4 | 0.4124/3 | 0.4561/4 |
| Dolphins | 0.5038/3 | 0.5021/7 | 0.5327/5 | 0.3128/66 | 0.5180/2 | 0.5091/3 | 0.5017/4 |
| Football | 0.5930/10 | 0.5876/12 | 0.6004/10 | 0.159/158 | 0.5853/5 | 0.6126/27/ | 0.6085/28 |
| Political books | 0.5315/4 | 0.5328/4 | 0.5183/2 | 0.2505/126 | 0.5007/10 | 0.5217/5 | 0.5388/4 |
| Jazz | 0.3555/3 | 0.3406/2 | 0.3451/3 | 0.1966/223 | 0.3551/8 | 0.3448/2 | 0.3393/2 |
| Email | 0.4220/13 | 0.4239/19 | 0.4187/16 | 0.0388/3132 | 0.4006/17 | 0.4399/13 | 0.4048/14 |
| Power | 0.5397/226 | 0.5235/226 | 0.5143/603 | 0.1876/1028 | 0.5058/326 | 0.5004/503 | 0.53226/528 |
| Astro-Ph | 0.6077/871 | 0.5662/1200 | 0.5980/1010 | 0.2994/884 | 0.5555/1086 | 0.5664/1165 | 0.5677/1514 |
| Hep-Th | 0.5593/942 | 0.5494/5102 | 0.5582/1273 | 0.2418/783 | 0.5546/542 | 0.5332/673 | 0.5389/783 |
| Internet | 0.4708/3910 | 0.4631/3970 | 0.45736/4221 | 0.3338/4380 | 0.4276/4210 | 0.4402/43121 | 0.4688/5380 |

Liu *et al.* [54] defined the following metric for the quality of community division:

$$Q_a = \frac{1}{2m} \sum_u \sum_{i,j} \left[A_{ij} - \frac{V_i V_j}{2m} \right] A(C_s^x, i) A(C_s^x, j) \quad (6)$$

where A is the adjacency matrix for a network, V_i is the degree of node i for an unweight network, C_s^x is a community in the network, and $A(C_s^x, i)$ represents the ratio of the number of links of node i that belong to community C_s^x and the number of edges connected to node i .

Since the most real community structures of real networks are unknown, therefore EQ and the Qa function evaluate the partition criterion for network clustering outcomes in

this paper. Table 4 gives the parameters of the 10 real-world networks in this evaluation.

Table 5 and Table 6 are the EQ and Qa values of the algorithms on the 10 real networks. Therefore, we conclude that a greater modularity score indicates higher quality of community. The modularity value is always consistent with the compact division in a network. It can be derived from the table that the proposed algorithm in this paper achieves well overlapping community detection precision on such types of true-world datasets. Compared to the other overlapping algorithms, the results of the proposed algorithm are reasonable. For actual networks, the proportion of correctly classified nodes is more or less the same as other algorithms.

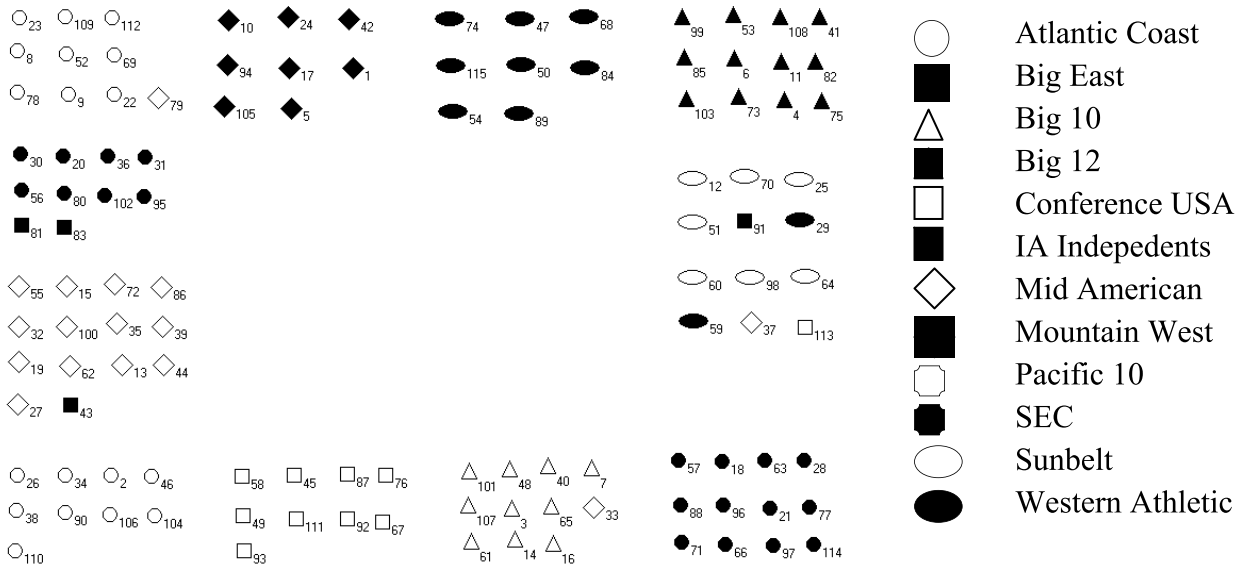


FIGURE 10. Applying LOC algorithm to the Football network.

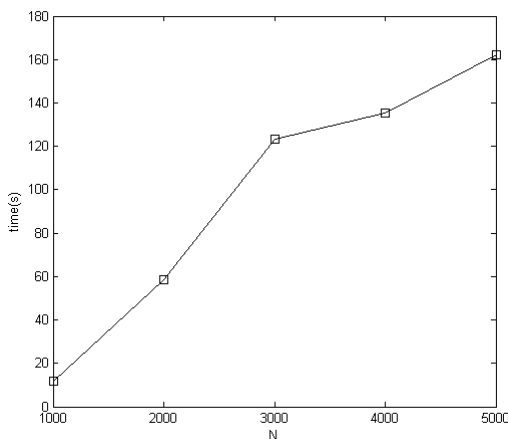


FIGURE 11. LOC algorithm runtime on LFR ($N = 1000 - 5000$, $k = 20$, $maxk = 50$, $minc = 10$, $maxc = 50$, $mu = 0.1$, $t_1 = 2$, $t_2 = 1$, $om = 2$, $on = 100$).

With the real-world networks, the detection accuracy of the proposed LOC algorithm in some fields is higher than that of the LINK and the MCNLPA algorithms, the scores of the LOC algorithm are superior to them. However, slightly less than the SLPA, the ACC and MC algorithms, the LOC algorithms generally perform the same as the GCE algorithms on the real-world data sets. High measure scores typically means compactable communities, and the LOC algorithm corresponds to the real partition, which is the reason why the value of a few metric perform weakness here. For instance, the GCE algorithm gets a great score with the Karate network and uncover four parts, but the genuine partition of this network is two. The conclusion is that the Qa function is consist with the EQ metric in general.

In Figs 8, 9 and 10, different communities are represented by different colors and sharps. The first dataset is the Karate Club network, which has 34 vertices and 78 edges.

The second case is the Dolphins data. The third case is the US college football network.

Figure 8 illustrates the results of partitioning of the Karate Club network using the LOC algorithm. The algorithm splits the Karate into two parts, and nodes 3, 9 and 10 are overlapping nodes. The result is identical with the correct overlapping nodes and overlapping community structures.

The LOC algorithm divides the Dolphins network into three communities, and nodes Mus, Number1 and Notch are overlapping nodes. Figure 9 denotes the visualization of the clustering of the Dolphin network applying the LOC algorithm. As can be seen, only node SN89 is incorrectly partitioned.

Another example from American football team network is the season schedule. It reveals that number of clusters is 12, and it uncover all clusters, approximately 87% of the teams are correctly identified into proper clusters. Figure 10 is the result of our algorithm.

The LOC algorithm has the higher NMI, EQ and Qa values, which indicates that its community detection outcomes are consilient with the genuine cluster structures. The algorithm is able to detect groups of networks, which the cliques can be tightly linked. Nodes on the clique boundary that are not part of the community may be deleted by the local optimization function. Consequently, the research results are the same as the true cluster structures.

The algorithm time efficiency and consumption are within acceptable limits, as Figure 11 reveals. The artificial-generated networks in the experiment are LFR benchmark, the network size N ranges from 1000 to 5000, the interval is 1000, the average degree $k = 20$, $maxk = 50$, and the exponent of nodes degree distribution t_1 and the exponent of cluster sizes t_2 equals to 2 and 1, on equals 100, om equals 2, $[minc, maxc]$ is [10, 50], the mixed parameter mu is 0.1.

The time efficiency and complexity of the algorithm are also within reasonable range.

V. CONCLUSION

This paper presents an algorithm to uncover overlapping communities in complex networks. The algorithm focuses on the assumption of k -cliques as the unit of communities, and proposes to identify community networks as a collection of overlapping linked cliques. Unlike other algorithms, a reasonable threshold should be set to determine whether there is an edge between cliques, or an isolated node should be considered separately. The algorithm looks upon as the seed selection strategy and topology of network community structures, a single node with the maximum density serves as the seed can eliminate similarities among k -cliques, and adds related-cliques as a natural community to expand, then, nodes with negative local optimization functions are cleared. The algorithm realizes the partition of overlapping network community structures. In both computer-generated networks and real-world networks, the experiment results demonstrate that the proposed algorithm can produce stable results. In addition, the time efficiency and time complexity of the algorithm are within acceptable limits, which also proves that this idea is practicable.

In the future, we plan to examine two points as the directions of the further work. First, we pointed out that the algorithm is directly used to weighted networks. Second, we will investigate applying to bipartite datasets for testing the algorithm.

ACKNOWLEDGMENT

The authors would like to express their great appreciation to the anonymous reviewers for their insightful comments.

REFERENCES

- [1] J. Huang, H. Sun, Q. Song, H. Deng, and J. Han, "Revealing density-based clustering structure from the core-connected tree of a network," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 8, pp. 1876–1889, Aug. 2013.
- [2] P.-Y. Chen and A. O. Hero, "Deep community detection," *IEEE Trans. Signal Process.*, vol. 63, no. 21, pp. 5706–5719, Nov. 2015.
- [3] G. J. Qi, C. C. Aggarwal, and T. Huang, "Community detection with edge content in social media networks," in *Proc. IEEE 28th Int. Conf. Data Eng.*, Washington, DC, USA, Apr. 2012, pp. 534–545.
- [4] Y. Cui, X. Wang, and J. Eustace, "Detecting community structure via the maximal sub-graphs and belonging degrees in complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 416, pp. 198–207, Dec. 2014.
- [5] J. Leskovec and R. Sosič, "SNAP: A general-purpose network analysis and graph-mining library," *TISTACM Trans. Intell. Syst. Technol.*, vol. 8, no. 1, pp. 1–20, Jul. 2016.
- [6] N. Salamanos, E. Voudigari, and E. J. Yannakoudakis, "Erratum to: Identifying influential spreaders by graph sampling," in *Proc. 5th Int. Workshop Complex Netw. Their Appl.*, Milan, Italy, 2017, pp. 111–122.
- [7] Y. Liu, M. Tang, T. Zhou, and Y. Do, "Core-like groups result in invalidation of identifying super-spreader by k -shell decomposition," *Sci. Rep.*, vol. 5, no. 1, Sep. 2015, Art. no. 9602.
- [8] M. E. J. Newman and A. Clauset, "Structure and inference in annotated networks," *Nature Commun.*, vol. 7, no. 1, p. 11863, Sep. 2016.
- [9] A. Grover and J. Leskovec, "NODE2VEC: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, 2016, pp. 855–864.
- [10] G. Ren and X. Wang, "Epidemic spreading in time-varying community networks," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 24, no. 2, Jun. 2014, Art. no. 023116.
- [11] M. Hosseini-Pozveh, K. Zamanifar, and A. R. Naghsh-Nilchi, "A community-based approach to identify the most influential nodes in social networks," *J. Inf. Sci.*, vol. 43, no. 2, pp. 204–220, Apr. 2017.
- [12] M. E. J. Newman and M. Girvan, "Community detection in networks: Modularity optimization and maximum likelihood are equivalent," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 94, p. 052315, Jun. 2016.
- [13] J. Xiang, J. Xiang, T. Hu, Y. Zhang K. Hu, J. Li, X. Xu, C. Liu, and S. Chen, "Local modularity for community detection in complex networks," *Phys. A, Stat. Mech. Appl.*, vol. 443, pp. 451–459, Feb. 2016.
- [14] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 76, no. 3, p. 036106, Oct. 2007.
- [15] G. Cordasco and L. Gargano, "Label propagation algorithm: A semi-synchronous approach," *Int. J. Social Netw. Mining*, vol. 1, no. 1, pp. 3–26, Dec. 2012.
- [16] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, nos. 3–5, pp. 75–174, Feb. 2010.
- [17] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 6, Jun. 2004, Art. no. 066133.
- [18] M. De Domenico and J. Biamonte, "Spectral entropies as information-theoretic tools for complex network comparison," *Phys. Rev. X*, vol. 6, no. 4, Dec. 2016, Art. no. 041062.
- [19] N. Meghanathan, "Spectral radius as a measure of variation in node degree for complex network graphs," in *Proc. 7th Int. Conf. u-e-Service, Sci. Technol.*, Dec. 2014.
- [20] J. Liu, Q. Xiong, W. Shi, X. Shi, and K. Wang, "Evaluating the importance of nodes in complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 452, pp. 209–219, Feb. 2016.
- [21] Y. Liu, Y. Deng, and B. Wei, "Local immunization strategy based on the scores of nodes," *Chaos*, vol. 26, no. 1, p. 013106, Jan. 2016.
- [22] G. Huang, B. Zhang, R. Ren, and J. Ren, "An algorithm to find critical execution paths of software based on complex network," *Int. J. Mod. Phys. C*, vol. 26, no. 09, Sep. 2015, Art. no. 1550101.
- [23] L. Fei, H. Mo, and Y. Deng, "A new method to identify influential nodes based on combining of existing centrality measures," *Mod. Phys. Lett. B*, vol. 31, no. 26, Sep. 2017, Art. no. 1750243.
- [24] X. Wang and X. Qin, "Asymmetric intimacy and algorithm for detecting communities in bipartite networks," *Phys. A, Statist. Mech. Appl.*, vol. 462, pp. 569–578, Nov. 2016.
- [25] Y. Cui and X. Wang, "Detecting one-mode communities in bipartite networks by bipartite clustering triangular," *Phys. A, Statist. Mech. Appl.*, vol. 457, pp. 307–315, Sep. 2016.
- [26] G. Palla, I. Derényi, I. J. Farkas, and T. Vicsok, "Uncovering the overlapping community structures of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, Jun. 2005.
- [27] E. Gregori, L. Lenzini, and S. Mainardi, "Parallel (k)-clique community detection on large-scale networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 8, pp. 1651–1660, Aug. 2013.
- [28] Y. Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, no. 7307, pp. 761–764, Oct. 2010.
- [29] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex network," *New J. Phys.*, vol. 11, no. 3, p. 033015, Mar. 2009.
- [30] C. Lee, F. Reid, A. McDaid, and N. Hurley, "Detecting highly overlapping community structure by greedy clique expansion," Jun. 2010, *arXiv:1002.1827*. [Online]. Available: <https://arxiv.org/abs/1002.1827>
- [31] X. Zhang, W. Zheng, C. Wang, Z. Zhuan, and Y. S. Sun, "An overlapping community detection algorithm based on addition of a clique at each step," *J. Cent. South Univ.*, vol. 44, no. 9, pp. 24–31, Sep. 2016.
- [32] X. Zhang, T. Martin, and M. E. J. Newman, "Identification of coreperiphery structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 91, p. 032803, Mar. 2015.
- [33] J. Xie, B. K. Szymanski, and X. Liu, "SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process," in *Proc. IEEE 11th Int. Conf. Data Mining Workshops*, Vancouver, BC, Canada, Dec. 2011, pp. 344–349.
- [34] P. Wu and L. Pan, "Detecting highly overlapping community structure based on maximal clique networks," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Beijing, China, Aug. 2014, pp. 196–199.

- [35] D. Jin, B. Gabrys, and J. Dang, "Combined node and link partitions method for finding overlapping communities in complex networks," *Sci. Rep.*, vol. 5, no. 1, p. 8600, Aug. 2015.
- [36] F. Hao, S. S. Yau, G. Min, and L. T. Yang, "Detecting k-balanced trusted cliques in signed social networks," *IEEE Internet Comput.*, vol. 18, no. 2, pp. 24–31, Mar. 2014.
- [37] X. Wen, W.-N. Chen, Y. Lin, T. Gu, H. Zhang, Y. Li, Y. Yin, and J. Zhang, "A maximal clique based multiobjective evolutionary algorithm for overlapping community detection," *IEEE Trans. Evol. Computat.*, vol. 21, no. 3, pp. 363–377, Sep. 2017.
- [38] J. Li, X. Wang, and Y. Cui, "Uncovering the overlapping community structure of complex networks by maximal cliques," *Phys. A, Statist. Mech. Appl.*, vol. 415, pp. 398–406, Mar. 2014.
- [39] Y. Cui, X. Wang, and J. Li, "Detecting overlapping communities in networks using the maximal sub-graph and the clustering coefficient," *Phys. A, Statist. Mech. Appl.*, vol. 405, pp. 85–91, Jul. 2014.
- [40] J. Yang, J. Mcauley, and J. Leskovec, "Community detection in networks with node attributes," in *Proc. IEEE 13th Int. Conf. Data Mining*, Dallas, TX, USA, Dec. 2013, pp. 1151–1156.
- [41] T. Zhang and B. Wu, "A method for local community detection by finding core nodes," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Istanbul, Turkey, Aug. 2012, pp. 1171–1176.
- [42] Z. Huang, Z. Wang, and Z. Zhang, "Detecting overlapping and hierarchical communities in complex network based on maximal cliques," in *Proc. Nat. Conf. Social Media Process.*, Guangzhou, China, 2015, pp. 184–191.
- [43] J. Eustace, X. Wang, and Y. Cui, "Community detection using local neighborhood in complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 436, pp. 665–677, Oct. 2015.
- [44] J. Eustace, X. Wang, and Y. Cui, "Overlapping community detection using neighborhood ratio matrix," *Phys. A, Statist. Mech. Appl.*, vol. 421, pp. 510–521, Mar. 2015.
- [45] J. Eustace, X. Wang, and J. Li, "Approximating Web communities using subspace decomposition," *Knowl.-Based Syst.*, vol. 70, pp. 118–127, Nov. 2014.
- [46] X. Wang and J. Li, "Detecting communities by the core-vertex and intimate degree in complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 392, no. 10, pp. 2555–2563, May 2013.
- [47] Z. Wang, Z. Li, and G. Yuan, "Tracking the evolution of overlapping communities in dynamic social networks," *Knowl-based Syst.*, vol. 157, pp. 81–97, Oct. 2018.
- [48] X. Wang, G. Liu, and J. Li, "Overlapping community detection based on structural centrality in complex networks," *IEEE Access*, vol. 5, pp. 25258–25269, Nov. 2017.
- [49] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 78, no. 4, Oct. 2008, Art. no. 046110.
- [50] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropol. Res.*, vol. 33, no. 4, pp. 452–473, 1977.
- [51] D. Lusseau, "The emergent properties of a dolphin social network," *Proc. Roy. Soc. London B, Biol. Sci.*, vol. 270, no. 2, pp. S186–S188, 2003.
- [52] M. E. J. Newman. *Network Data*. Accessed: Oct. 12, 2017. [Online]. Available: <http://www-personal.umich.edu/~mejn/netdata>
- [53] H. Shen, X. Chen, K. Cai, and M. Hu, "Detect overlapping and hierarchical community structure in networks," *Phys. A, Statist. Mech. Appl.*, vol. 388, no. 8, pp. 1706–1712, Apr. 2009.
- [54] J. Li, X. Wang, and J. Eustace, "Detecting overlapping communities by seed community in weighted complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 392, no. 23, pp. 6125–6134, Dec. 2013.
- [55] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the definition of modularity to directed graphs with overlapping communities," *J. Statist. Mech.*, vol. 2009, no. 3, pp. 3166–3168, Mar. 2008.



JIAN MA received the M.S degree in computer and information technology from Beijing Jiaotong University, in 2009, where she is currently pursuing the Ph.D. degree with the School of Computer and Information Technology. Her main research interests include complex network analysis and data mining.



JIANPING FAN was born in 1963. He received the Ph.D. degree. He is an Adjunct Professor and a Ph.D. Supervisor with Beijing Jiaotong University. His main research interests include computing, cloud computing, and parallel and distributed computing.

• • •