

# An improved two-stage label propagation algorithm based on LeaderRank

Miaomiao Liu<sup>Equal first author, 1, 2</sup>, Jinyun Yang<sup>Corresp., Equal first author, 1</sup>, Jingfeng Guo<sup>3</sup>, Jing Chen<sup>3</sup>, Yongsheng Zhang<sup>1</sup>

<sup>1</sup> School of Computer and Information Technology, Northeast Petroleum University, Daqing, Heilongjiang, China

<sup>2</sup> Key Laboratory of Petroleum Big Data and Intelligent Analysis of Heilongjiang Province, Northeast Petroleum University, Daqing, Heilongjiang, China

<sup>3</sup> College of Information Science and Engineering, Yanshan University, Qinhuangdao, Hebei, China

Corresponding Author: Jinyun Yang

Email address: yjy1129@foxmail.com

**Abstract.** To solve the problems of poor stability and low modularity ( $Q$ ) of community division results caused by the randomness of node selection and label update in the traditional label propagation algorithm, an improved two-stage label propagation algorithm based on LeaderRank was proposed in this study. In the first stage, the order of node updating is determined by the participation coefficient (PC). Then, a new similarity measure is defined to improve the label selection mechanism so as to solve the problem of label oscillation caused by multiple labels of the node with the most similarity to the node. Moreover, the influence of the nodes is comprehensively used to find the initial community structure. In the second stage, the rough communities obtained in the first stage are regarded as nodes, and their merging sequence is determined by the PC. Next, the non-weak community and the community with the largest number of connected edges are combined. Finally, the community structure is further optimized to improve the modularity so as to obtain the final partition result. Experiments were performed on six real networks and fifteen artificial datasets with different scales, complexities, and densities. The modularity and normalized mutual information (NMI) were considered as evaluation indexes for comparing the improved algorithm with dozens of relevant classical algorithms. The results showed that the proposed algorithm yields superior performance, and the results of community partitioning obtained using the improved algorithm were stable and more accurate than those obtained using other algorithms. In those classical real networks, the modularity of community division results of the proposed algorithm was higher than that of other algorithms. And the NMI values were above 0.9943 on eight artificial datasets containing 1000 nodes, and 0.986 and 0.953 on two large-scale complex networks containing 5000 nodes. In addition, the proposed algorithm always performs well in community detection in five large-scale artificial data sets with 6000 to 10000 nodes.

# An improved two-stage label propagation algorithm based on LeaderRank

Miaomiao Liu<sup>1,2</sup>, Jinyun Yang<sup>1,\*</sup>, Jingfeng Guo<sup>3</sup>, Jing Chen<sup>3</sup>, Yongsheng Zhang<sup>1</sup>

<sup>1</sup> School of Computer and Information Technology, Northeast Petroleum University, Daqing Heilongjiang, China

<sup>2</sup> Key Laboratory of Petroleum Big Data and Intelligent Analysis of Heilongjiang Province, Daqing, Heilongjiang, China

<sup>3</sup> College of Information Science and Engineering, Yanshan University, Qinhuangdao, Hebei, China

Corresponding Author:

Jinyun Yang<sup>1,\*</sup>

Kaifaquliming Street, Daqing, Heilongjiang, 163318, China

Email address: yjy1129@foxmail.com

**Abstract.** To solve the problems of poor stability and low modularity ( $Q$ ) of community division results caused by the randomness of node selection and label update in the traditional label propagation algorithm, an improved two-stage label propagation algorithm based on LeaderRank was proposed in this study. In the first stage, the order of node updating is determined by the participation coefficient (PC). Then, a new similarity measure is defined to improve the label selection mechanism so as to solve the problem of label oscillation caused by multiple labels of the node with the most similarity to the node. Moreover, the influence of the nodes is comprehensively used to find the initial community structure. In the second stage, the rough communities obtained in the first stage are regarded as nodes, and their merging sequence is determined by the PC. Next, the non-weak community and the community with the largest number of connected edges are combined. Finally, the community structure is further optimized to improve the modularity so as to obtain the final partition result. Experiments were performed on six real networks and fifteen artificial datasets with different scales, complexities, and densities. The modularity and normalized mutual information (NMI) were considered as evaluation indexes for comparing the improved algorithm with dozens of relevant classical algorithms. The results showed that the proposed algorithm yields superior performance, and the results of community partitioning obtained using the improved algorithm were stable and more accurate than those obtained using other algorithms. In those classical real networks, the modularity of community division results of the proposed algorithm was higher than that of other algorithms. And the NMI values were above 0.9943 on eight artificial datasets containing 1000 nodes, and 0.986 and 0.953 on two large-scale complex networks containing 5000 nodes. In

addition, the proposed algorithm always performs well in community detection in five large-scale artificial data sets with 6000 to 10000 nodes.

**Keywords.** Label propagation; Community division; LeaderRank; Node influence; Weak community; Modularity

## Introduction

With the rapid development of the Internet and big data technology, research on complex networks has gradually penetrated into many fields, such as information science and biological science, and has thus become a very challenging research topic (5). In social networks, such as scientific research cooperation and transportation networks, objects are usually represented as nodes, and relationships between objects are represented as edges (40). Real-world networks have one important feature, community structure, that is, a network is usually composed of several communities, with relatively close node connections within the community and relatively sparse node connections between the communities. The discovery of community structure is an important basis for exploring the formation principle and function of complex network structures (22) and plays a vital role in many fields. For instance, in the field of biology (19), community detection is of great significance for understanding the specific organizational structure, functional analysis, and behavior prediction of biological systems. In the field of e-commerce, consumers with similar purchasing habits can be mined through community detection, thus creating greater business value through the establishment of efficient recommendation systems (11). In the field of infectious diseases, community detection can be used to analyze and identify the key population of infectious diseases so as to effectively control the spread of diseases (3). Therefore, the quick and effective discovery of the community structure of networks has become the primary task and an important branch of social network research.

With the extensive research on social network analysis, many community detection algorithms have emerged, but most of them suffer from limitations such as high complexity, low accuracy of community division, and unstable results. Label propagation algorithm (LPA) has attracted attention due to its advantages of low time complexity, no prior conditions, and suitability for community detection in large-scale networks (10). However, the traditional LPA has the following disadvantages: 1) LPA adopts a random strategy in the updating sequence of nodes, resulting in randomness in the community partition results; 2) LPA treats every node as equally important and does not distinguish the importance of each node; 3) LPA assigns a unique label to each node and fails to identify overlapping communities (17).

In view of the abovementioned shortcomings, numerous improved algorithms have been proposed. (4) proposed an improved LPA algorithm based on label propagation ability, developed a calculation method based on a k-shell decomposition algorithm for determining the importance of individual nodes (21), and formulated a label update strategy through the importance ranking of nodes and label propagation ability. (30) proposed a community detection algorithm based on node influence and similarity (NIS-LPA), wherein the selected seed nodes are used to expand into seed regions, and then the similarity between nodes is calculated based

on the network topology and real attributes of nodes, thus improving the stability and accuracy of the algorithm. (34) proposed a community detection algorithm integrating LeaderRank and label propagation (LLPA) wherein the three aspects of node label initialization, node update sequence, and label propagation selection process are improved. The LeaderRank algorithm is adopted to select key nodes, and labels are assigned to them by calculating the influence of the nodes. Thereafter, the nodes are updated according to the influence of the nodes, and the propagation ability between nodes is considered in the process of label propagation. (7) proposed a community detection algorithm based on boundary nodes and label propagation (LBN), which determines core nodes and boundary nodes, respectively, and then determines the community to which they belonged according to the weight of the boundary nodes, thus improving the stability of the algorithm. However, the values of  $Q$  (33) and NMI (35) are still unsatisfactory. (38) proposed label importance–based label propagation algorithm (LILPA) for community detection for application in core drug detection. In LILPA, when labels are transmitted to other nodes, the label updating process based on node importance, node attractiveness, and label importance is used to improve the label instability and the accuracy and efficiency of community division. For overlapping communities, (14) proposed an efficient community detection algorithm based on label propagation with community kernel (CK-LPA), which assigns a corresponding weight to each node according to the importance of the node in the network and updates node labels according to the weight order. They also discussed the composition of weights, label updating, propagation strategies, and convergence conditions. (18) improved the label update order and propagation threshold. They proposed an overlapping community detection algorithm based on label propagation by using the PageRank and node clustering coefficients algorithms (COPRAPC), wherein nodes with low influence are selected for label propagation, and the node clustering coefficient is used to control the maximum number of communities that nodes belong to. (25) proposed an overlapping community detection algorithm integrating label preprocessing and node influence (FLPNI), thereby greatly reducing the randomness of label propagation. (29) proposed an improved label propagation algorithm for community detection based on two-level neighborhood similarity (TNS-LPA); defined a new two-level neighborhood similarity measurement, which selected the initial community center by considering the minimum distance and local centrality index; and optimized the algorithm by adopting the asynchronous updating label strategy according to the importance of nodes, thereby further improving the accuracy of community division. (12) proposed an improved label propagation algorithm based on modularity and node importance (LPA-MNI) wherein the initial community is identified based on the value of modularity, and then the remaining nodes that have not been assigned to the initial community are clustered through label propagation. Node importance is used to improve the label update sequence, and the label selection mechanism is used when the majority of nodes contain multiple labels. Experimental results showed that LPA-MNI is more robust than the traditional LPA algorithm. (8) proposed the node importance–based label propagation algorithm (NI-LPA) to identify overlapping communities to address the problem of instability in the LPA algorithm caused by random updating. NI-LPA uses information derived from node attributes to

simulate special propagation and filtering processes. Experiments on artificial and real networks of different sizes, complexities, and densities revealed the high efficiency of NI-LPAT for overlapping community detection. (24) proposed another label propagation algorithm based on node importance (NI-LPA) wherein the importance of nodes is defined by combining the signal propagation of nodes, the value of K-shell nodes themselves, and the Jaccard distance between adjacent nodes, which better avoids the instability caused by random selection of nodes in the traditional LPA algorithm. (15) encoded both semantic and geometric information of the environment in a weighted colored graph, in which the edges were partitioned into a finite set of ordered semantic classes (e.g., colors), and then incrementally searched for the shortest path among the set of paths with minimal inclusion of inferior classes, using information from the previous search by ideas similar to the LPA. (1) In the first and second iterations ( $t \leq 2$ ) of the propagation, if the number of maximum label frequencies in neighbor nodes was equal, the Adamic/Adar index was used to select the appropriate label. For the other iterations ( $t > 2$ ), a new criterion, known as label strength, was applied to select the label with the highest strength of a node. (35) proposed a new node similarity metric, and the label was updated according to the similarity between the current node and neighbor nodes.

The abovementioned algorithms focus on the calculation of the node importance and seed node selection and consider the randomness of node update order but ignore the importance of label update strategy, resulting in the unstable and less accurate community division. Therefore, this study focused on the updating strategy of nodes and labels to achieve efficient and accurate community division. The two-stage community detection algorithm based on the label propagation algorithm (39) (LPA-TS) has the following problems. 1) In the first stage, the algorithm determines the node update sequence from the descending participation coefficient (PC) and then updates the node label to that with the largest similarity so as to obtain the initial partition result. However, only the number and degree of common neighbors are considered in the definition of similarity. There may be multiple nodes with maximum similarity with the same number and degree of common neighbors. If one node is randomly selected for label update, the result of community division will be unstable. 2) In the second stage, the algorithm first regards the initial community as nodes and then determines the order of community mergers from the PC. Then, the algorithm performs merging according to the conditions of a weak community and finally obtains the community structure. However, in some classical networks, the community division results are not ideal, and the modularity is low because LPA-TS has some shortcomings in the updating strategy of nodes and labels and the definition of initial community merge conditions. To solve these problems, an improved two-stage label propagation algorithm (LPA-ITSLR) was proposed in this paper. The contributions and innovations of this paper are as follows.

(1) To solve the problem of unstable and inaccurate community division results yielded by the LPA-TS algorithm, a new similarity measurement index between nodes was proposed to optimize the node label updating strategy. In the initial stage of community division, the number and degree of common neighbors of nodes and the similarity of structural information between

common neighbors are considered comprehensively. In view of the situation that multiple nodes may have the maximum similarity value, the importance of nodes is sorted by calculating the LeaderRank value so as to avoid the randomness of node label update order and ensure the stability of the initial community division result.

(2) To address the problem of low modularity in LPA-TS, the optimal parameter value was determined by improving the definition of weak community in the original algorithm, and the evaluation function based on complementary entropy was changed to the objective function based on modularity optimization in the community merging stage so as to further improve the quality of community division and the accuracy of the final division result.

(3) Experiments were conducted on six real networks and 15 artificial datasets with different scales and complexities. The  $Q$  and  $NMI$  values were used as evaluation indexes to compare the proposed algorithm with several recent label propagation algorithms. Experimental results showed that the improved algorithm has higher quality and stability in the community division.

## Theoretical Basis

### Community division

A complex network is generally represented by  $G(V, E)$ , where  $V = \{v_1, v_2, \dots, v_n\}$  is the node set,  $E = \{e_1, e_2, \dots, e_m\}$  represents the set of edges, and  $\Omega = \{\Omega_1, \Omega_2, \dots, \Omega_k\}$  is one of the division of  $G$  if and only if:

- 1)  $\forall \Omega_r \in \Omega, \Omega_r \neq \emptyset$ ;
- 2) For all  $\Omega_r \in \Omega, \bigcup_{\Omega_r \in \Omega} \Omega_r = V$ ;
- 3)  $\forall \Omega_p, \Omega_q \in \Omega, \Omega_p \cap \Omega_q = \emptyset$ ;

### Participation coefficient

The PC of node  $v_i$   $PC_i$ (39) is used to describe the distribution of nodes with different communities in the network edge; it is defined as Eq. (1), where  $k$  is the number of communities, and  $d_i$  is the degree of node  $v_i$  and  $d_i(\Omega_r) = |\{v_j | (v_i, v_j) \in E \wedge v_j \in \Omega_r\}|$ . A high PC value indicates that the node is connected with more communities and that the node has a low degree of belonging to each community. In contrast, a low PC value indicates that the node is connected to a fewer number of communities and that the node has a high degree of belonging to each community. When community detection is performed, nodes with low PC and obvious community affiliation are selected to start traversal, which is more conducive for finding the correct community structure.

$$PC_i = 1 - \sum_{r=1}^k \left( \frac{d_i(\Omega_r)}{d_i} \right)^2 \quad (1)$$

### Strong and weak communities

Community structures can be strong or weak(39). A strong community means that the number of links between any node in the community and the inside of the community is greater than the number of links between the node and the outside of the community. It can be defined as Eq. (2). A weak community means that the sum of the edges of all nodes in the community and the nodes inside the community is greater than the sum of the edges of all nodes outside the community. It

can be defined as Eq. (3). In general, a community should satisfy at least the character of weak community.

$$\alpha * d_i^{in}(\Omega_r) > d_i^{out}(\Omega_r), \forall i \in \Omega_r \quad (2)$$

$$\alpha * \sum_{i \in \Omega_r} d_i^{in}(\Omega_r) > \sum_{i \in \Omega_r} d_i^{out}(\Omega_r) \quad (3)$$

In Eqs. (2) and (3),  $d_i^{in}(\Omega_r)$  represents the number of connected edges between the node  $v_i$  in the community  $\Omega_r$  and the internal nodes of  $\Omega_r$ ,  $d_i^{out}(\Omega_r)$  represents the number of connected edges between node  $v_i$  in  $\Omega_r$  and other nodes except  $\Omega_r$ . In general,  $\alpha = 2$ .

### LeaderRank algorithm

The LeaderRank algorithm(34) is used to calculate the influence of nodes in the network. A background node  $v_g$  is added to the network and connected with all the nodes in the network to form a new network. The algorithm assigns 1 unit of LeaderRank (LR) value to all nodes except the background node in advance, assigns 0 unit of LR value to node  $v_g$ , uses Eq. (4) to calculate the LR value of node  $v_i$  in each iteration, and iterates repeatedly until  $v_i$  reaches a steady state and then uses Eq. (5) to divide the background nodes evenly among all the nodes.

$$LR_i(t) = \sum_{j=0}^N \frac{a_{ij}}{d_j} LR_j(t-1) \quad (4)$$

Where  $t$  is the number of iterations and  $N$  is the number of nodes in the network. If there is an edge between nodes  $v_i$  and  $v_j$ , then  $a_{ij} = 1$ , otherwise,  $a_{ij} = 0$ ;  $d_i$  represents the degree of node  $v_i$ , and  $a_{ij}/d_j$  represents the probability of node  $v_i$  walking to node  $v_j$  randomly.

$$LR_i = LR_i(t_c) + \frac{LR_g(t_c)}{N} \quad (5)$$

Where  $t_c$  is the number of iterations when it reaches stability and  $LR_i(t_c)$  is the LR value when node  $v_i$  reaches stability. Similarly,  $LR_g(t_c)$  is the LR when node  $v_g$  reaches stability.

### Evaluation Indicators

**Modularity.** Modularity ( $Q$ ), proposed by Newman *et al.*, is commonly used for measuring the strength of community structures. The closer its value is to 1, the higher the strength of community structures, that is, the better the quality of community division(33).  $Q$  can be calculated as follows:

$$Q = \frac{1}{2m} \sum_{i,j} (a_{ij} - \frac{d_i d_j}{2m}) \delta_{i,j} \quad (6)$$

Where  $m$  is the total number of edges in the network,  $a_{ij}$  is an element in the adjacency matrix  $A$  of network  $G$ , and  $d_i$  is the degree of node  $v_i$ . When  $v_i$  and  $v_j$  belong to the same community,  $\delta_{i,j} = 1$ ; otherwise  $\delta_{i,j} = 0$ .

**Normalized mutual information.** For networks with a known community structure, NMI is generally used to evaluate the community division effect. The higher the NMI value, the more similar the result is to the real community structure. A value of 1 indicates that the partition result is completely consistent with the actual community structure (35). Assuming that  $A = \{A_1$

,  $A_2, \dots, A_k$  and  $B = \{B_1, B_2, \dots, B_{k'}\}$  are the real community structure of the network and the community division result of the network by an algorithm, respectively,  $k$  and  $k'$  are the number of communities under the two divisions, NMI can be defined as follows:

$$NMI = \frac{2 \sum_{i=1}^k \sum_{j=1}^{k'} T_{ij} \log \frac{n T_{ij}}{T_i T_j}}{- \sum_{i=1}^k T_i \log \frac{T_i}{n} - \sum_{j=1}^{k'} T_j \log \frac{T_j}{n}} \quad (7)$$

Where  $n$  is the total number of nodes in the network,  $T$  is the confusion matrix,  $T_{ij}$  is the number of common nodes included in the real divided communities  $A_i$  and  $B_j$ , and  $T_i$  is the sum of the elements in the  $i$ -th row of the confusion matrix.

## The Proposed Algorithm

### Question-posing

In the first stage of the LPA-TS algorithm, when there are two or more nodes with the largest similarity with the current node, the algorithm randomly selects one node for label update; this may lead to unstable partition results. LPA-TS algorithm expresses the similarity of nodes as  $CN$ , which can be expressed as Eq. (8), where  $N_i$  represents the neighbor nodes of node  $v_i$  and  $d_i$  represents the degree of node  $v_i$ .

$$CN(v_i, v_j) = |N_i \cap N_j| + \frac{1}{d_i d_j} \quad (8)$$

In the network diagram shown in Figure 1, when the LPA-TS algorithm is used for the first stage of community division, two initial rough communities are obtained:  $\Omega_1 = \{v_7, v_8, v_9, v_2\}$  and  $\Omega_2 = \{v_1, v_3, v_4, v_5, v_6\}$ . The nodes in different communities are represented by different shapes and colors in Figure 1. At this time, node  $v_0$  has not yet been merged into any community. As shown in Figure 1,  $N_0 = \{v_2, v_8, v_1, v_5\} \wedge (v_2, v_8 \in \Omega_1) \wedge (v_1, v_5 \in \Omega_2)$ . According to the similarity calculation formula of the LPA-TS algorithm, the node  $v_0$  has a large similarity with  $v_9$  in the community  $\Omega_1$ ; in addition,  $v_0$  has a large similarity with  $v_3$  in the community  $\Omega_2$ . Moreover, both  $v_9$  and  $v_3$  have the same neighbor attributes as node  $v_0$ , that is,  $CN(v_9, v_0) = CN(v_3, v_0)$ . At this time, the LPA-TS algorithm randomly selects a community to merge  $v_0$  into it and finally yields two community division results,  $\Omega = \{\{v_7, v_8, v_9, v_2\}, \{v_0, v_1, v_3, v_4, v_5, v_6\}\}$  and  $\Omega' = \{\{v_7, v_8, v_9, v_2, v_0\}, \{v_1, v_3, v_4, v_5, v_6\}\}$ , resulting in instability.

**Figure 1 A network instance with two communities**

The traditional LPA algorithm and the LPA-TS algorithm were used to conduct 100 experiments on classic Karate and Football networks. The corresponding module degree  $Q$  (33) of the community division results is shown in Figure 2. Both algorithms exhibited obvious oscillations, indicating that the community division results of the algorithms are unstable.

(a) (b)

**Figure 2 Results of 100 experiments of the two algorithms on two networks**

The LPA-TS algorithm only yielded an initial community division result in the first stage, and there were still many small communities. In the Karate network shown in Figure 3, some nodes



with higher degrees have greater similarities with many nodes. For example, node  $v_{34}$  can easily pass its label to neighboring nodes, while those at the edge of the network have low similarity to central nodes higher degrees. For example, nodes  $v_{25}$  and  $v_{26}$  can easily form small-scale communities, such as triangle nodes and diamond nodes in Figure 3. To merge these small communities, LPA-TS uses the definition of weak communities and the evaluation function based on complementary entropy in the second stage. However, in the definition of weak communities  $\alpha$  is set as 2, which leads to unstable division results in some networks, that is, the final community division results are not ideal, and the degree of modularity is low. Therefore, in this study, the parameters and the objective function in the second-stage community merger strategy were improved and a new community division method, LPA-ITSLR, was developed to achieve stable and more accurate community division results.

**Figure 3 LPA-TS algorithm partitioning results for Karate network in the first stage**

### Node similarity definition

To solve the abovementioned problems, a new similarity index was proposed, which comprehensively considers the common neighbors, degrees of nodes, and the structural relationship between common neighbors. The more common neighbors two nodes have, the more similar they are. The higher the degree of a node, the more the number of nodes it shares its edges with, that is, the similarity of two nodes is inversely proportional to the degree of the node itself. The number of connected edges between neighboring nodes is combined to avoid multiple nodes with the same similarity value as the original node. The improved similarity index can be expressed as follows:

$$S_{(i,j)}^{ICN} = \frac{|N_i \cap N_j| + 1}{|N_i \cup N_j|} + \frac{1}{d_i d_j} + \frac{C(v_i, v_j)}{|N_i \cap N_j|} \quad (9)$$

Where  $C(v_i, v_j)$  represents the number of edges between the common neighbor nodes of nodes  $v_i$  and  $v_j$ . The numerator of the first term of the equation is increased by 1 so that the improved similarity index is not 0 when there is no public neighbor. According to the definition of similarity in Eq. (9), the similarity between nodes  $v_0$  and  $v_9$  in Figure 1 is greater; thus, node  $v_0$  is merged into the community  $\Omega_1$ , which solves the problem of unstable community division results that may occur in the LPA-TS algorithm.

### Algorithm description

**Stage 1: Initial community detection.** The first stage of the proposed LPA-ITSLR algorithm is shown in Table 1. First, each node is assigned a unique label, and the similarity between nodes is calculated according to Eq. (9). Then, the PC value of each node is calculated and sorted in ascending order. Next, the labels are updated according to the sorted nodes. In the label updating strategy, the similarity between the current node and other nodes is compared. If the node with the largest similarity is not unique, the LR is further compared; if not, one is randomly selected to obtain the rough initial community structure in the first stage.

**Table 1 First stage of the LPA-ITSLR algorithm**

**Stage 2: Community merge.** The second stage of the proposed LPA-ITSLR algorithm is shown in Table 2. In view of the problem that small communities may cause low network

modularity in the first stage, whether the initial community meets the weak community condition is judged first. If the condition is not met, the community with the largest number of connected edges is selected for merging; this process is repeated until the entire network meets the weak community condition. The research of LPA-TS (39) algorithm shows that  $\alpha$  is generally set as 2 in Eq. (3). In order to achieve more accurate community detection results,  $\alpha$  is set as 0.5, 1, 1.2, 1.5 and 2, respectively for the 8 real data sets used in this paper. Through repeated experiments, it can be known that when  $\alpha$  is set as 1.5, good division results are achieved on the 8 networks. Moreover, experiments are also carried out on 15 artificial data sets to verify the rationality of the value of  $\alpha$ . Therefore, in our study,  $\alpha$  is determined to be 1.5 to achieve better performance in all networks. Each community is regarded as a node, and its PC value is calculated using Eq. (1) to determine the community with the largest PC value; then, the community with the most links is determined for merging. If the modularity increases after the merge, the merge will be selected; otherwise, it will not be merged, thus ensuring that the community structure after the second stage merge will have a higher modularity and be closer to the real community structure.

**Table 2 Second stage of the LPA-ITSLR algorithm**

## Experiment and Analysis

In this study, numerous experiments are conducted on real networks and artificial datasets with different structural parameters. The classical LPA algorithm, LPA-TS algorithm, and several community detection algorithms based on label propagation were compared; moreover, the effectiveness, correctness, stability, and accuracy of the proposed algorithm were verified.

### Analysis of experimental results on real networks

**Real dataset.** In total, six classic real social network datasets were used in the experiment; their attribute characteristics are presented in Table 3. In Table 3,  $|V|$  represents the total number of nodes in the network,  $|E|$  represents the total number of edges,  $|\Omega|$  represents the number of communities included in the network,  $\max(k)$  represents the maximum node degree,  $\langle k \rangle$  represents the average node degree,  $\langle L \rangle$  represents the average path length, and  $\langle c \rangle$  represents the clustering coefficient.

**Table 3 Basic structural parameters of real datasets**

**Community division results.** The proposed LPA-ITSLR algorithm was used to divide communities in the six abovementioned real datasets. The results are illustrated in Figure 4, where nodes in different communities are represented by different color.

**Figure 4 Community detection results of real networks**

**Stability analysis of LPA-ITSLR.** The proposed LPA-ITSLR algorithm and the LPA and LPA-TS algorithms were compared and analyzed in the six abovementioned real datasets. Each dataset was run independently for 10 times, and the average value of the three algorithms on the six datasets was obtained (denoted as  $\langle Q \rangle$ ), as shown in Table 4. The independent experimental results for each time are shown in Figure 5.

As can be seen from the experimental results presented in Table 4 and Figure 5, LPA-ITSLR performed well in all datasets, with the exception that the average module degree on the NetScience network was slightly lower than that obtained using the other two algorithms.

Moreover, LPA-ITSLR yielded more stable community partitioning results and a higher module degree than the other two algorithms. NetScience is a weighted network; however, in the experiment, the weight was ignored, and it was transformed into a powerless network for community division. Therefore, the quality of community division on this network obtained using LPA-ITSLR was slightly lower than that obtained using the other two algorithms. However, in the 10 independent experiments, the results of the LPA and LPA-TS algorithms exhibited fluctuations, indicating that the two algorithms are unstable due to the randomness of node and label update. The module-degree value of the proposed LPA-ITSLR algorithm always remained stable for every network, indicating that LPA-ITSLR effectively solves the oscillation problem in the process of label propagation and has higher accuracy and stability. To further verify the robustness of LPA-ITSLR, 100 independent experiments were conducted on the Karate, Dolphin, and Football networks; the results are presented in Figure 6. The community division results obtained using the LPA algorithm exhibited the most serious fluctuations in the modularity value, followed by the LPA-TS algorithm. In contrast, LPA-ITSLR maintained the same community division results in 100 experiments, and the modularity was higher than that of LPA and LPA-TS.

**Table 4 Average modularity values of 10 experiments for the three algorithms on real datasets**

**Figure 5 Comparison of algorithm stability**

(a) (b) (c)

**Figure 6 Comparison of modularity of LPA, LPA-TS, and LPA-ITSLR**

To further evaluate the performance of the LPA-ITSLR algorithm, it was compared with four recent community detection algorithms based on label propagation. Among them, the COPRA algorithm (6) realizes community division by assigning multiple labels with attribution coefficients to a node. The WLPA algorithm (23) first selects the label with a larger weight for propagation during the label propagation process. The LINSIA (26) algorithm is based on node importance and employs label importance to complete the community division. The LILPA (38) algorithm uses a fixed label update sequence based on the ascending order of node importance for discovering communities. The modularity of the results obtained using the five algorithms on the four real datasets is presented in Table 5.

**Table 5 Modularity comparison of five algorithms**

From Table 5, it can be seen LPA-ITSLR yielded the highest modularity and most stable in community division results. Thus, instability caused by label oscillation is avoided effectively by using LPA-ITSLR.

**Performance comparison of LPA-ITSLR with other algorithms.** For further analysis of the effectiveness of the proposed algorithm for community partition and correctness, three classic datasets of Karate, Dolphins, and Football were used, and the LPA-ITSLR algorithm and seven classic community detection algorithms were employed for obtaining the division results for correlation analysis in terms of the number of communities  $|Q|$  and module  $Q$  as evaluation indicators. The results are presented in Table 6.

**Table 6 Results of eight algorithms on classical networks**

From Table 6, it can be seen that the number of communities and modularity of the partition results of the eight algorithms on the three classical networks were different, but LPA-ITSLR exhibited good performance on these datasets; moreover, the partition results and the number of communities were consistent with the real network structure, and the modularity was higher than that obtained using other algorithms.

### Analysis of experimental results of artificial datasets

**Artificial datasets.** Ten artificial networks were generated using the LFR benchmark<sup>(31)</sup>; the basic information is presented in Table 7. The number of nodes  $|V|$  in the top eight artificial networks is 1000, and the community size  $|\Omega|$  is 10–50, that is,  $\min|\Omega| = 10$ ,  $\max|\Omega| = 50$ . The average degree of nodes  $\langle k \rangle$  is 20, and the maximum degree  $\max(k)$  is 50. The values of 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, and 0.45 were employed as the mixing parameter  $\mu$ , and the eight networks were denoted as LFR-1–LFR-8. The latter two artificial networks are more complicated. The number of nodes is 5000, the community size is 50, the average degree of nodes is 10, and the maximum degree is 50. The mixing parameter  $\mu$  was 0.1 and 0.3, respectively, and these two networks were denoted as LFR-9 and LFR-10.

**Table 7 Description of synthetic networks**

**Comparative analysis of algorithm performance.** For the first eight artificial datasets, the proposed LPA-ITSLR algorithm was compared with the LPA and LPA-TS algorithms in terms of the community division results. The average modularity  $\langle Q \rangle$  and NMI were used as evaluation indicators. The experimental results are shown in Figure 7. As the value of  $\mu$  increased, the network became more complex. The modularity of the community division results of the three algorithms on the corresponding network decreased by varying degrees, but LPA-ITSLR yielded higher modularity than the other algorithms. Moreover, the NMI value of LPA-ITSLR on the first seven networks was 1, and the NMI value of the network with a  $\mu$  value of 0.45 was 0.9943, showing extremely strong stability and higher quality of community division.

**Figure 7 Comparison of modularity and NMI on eight synthetic datasets**

For large-scale artificial networks LFR-9 and LFR-10 with high complexity, the proposed LPA-ITSLR algorithm was compared with seven recent label propagation algorithms for community division.  $Q$  and  $NMI$  were considered as evaluation parameters. The results are presented in Table 8.

**Table 8 Results for LFR9 and LFR10**

From Table 8, it can be seen that the community division results obtained using the seven algorithms were not stable, and the algorithm proposed in this paper maintains stable community division results on the two complex artificial data sets. Although the NMI value was slightly lower than that for other algorithms, the modularity was far higher. In the community merge phase optimization strategy based on modularity, LPA-ITSLR is superior as it can yield stable and high-quality community division results.

For the above 10 artificial networks, the experimental results show that the proposed algorithm is superior to other algorithms in both  $Q$  and  $NMI$ . In order to further verify the superiority of the proposed algorithm, we compare the number of communities detected by LPA-ITSLR algorithm with the actual number of communities of the ten networks, and the results are shown in Table 9.

**Table 9 Actual number of communities and the number of communities detected by LPA-ITSLR**

It can be seen from table 9 that the number of communities detected by the algorithm proposed in this paper is basically consistent with the actual number of communities. In general, good results are obtained except for small deviations in some networks.

**Performance analysis of the algorithm for large data sets.** In order to further verify the effectiveness of the algorithm, several large-scale data sets were used for experiments in this paper. The number of nodes of these artificial networks were 6000, 7000, 8000, 9000 and 10000, respectively, and these networks were denoted as LFR-11 to LFR-15, respectively. Table 10 shows the experimental results of the algorithm on these five large-scale networks, including the actual number of communities, the number of communities detected by the LPA-ITSLR algorithm,  $Q$  and NMI.

**Table 10 Community detection results of five large-scale artificial networks**

It can be seen from table 10 that, the algorithm performs well on these large data sets. With the increase of the number of nodes, the network scale and complexity continues to expand, and the modularity and NMI all showed a trend of decline. But the  $Q$  is always above 0.86, and the NMI is more than 0.95 by and large. In addition, the number of communities detected by the algorithm proposed is basically consistent with the actual number of communities, which further verifies the effectiveness and superiority of the LPA-ITSLR algorithm.

## Conclusions

To solve the problem of unstable results and low modularity of the LPA-TS algorithm in community detection on some networks, an improved LeaderRank-based two-stage label propagation algorithm named LPA-ITSLR was proposed in this study. In the first stage, the order of node updating is determined by descending order of the PC values. In the label propagation strategy, the improved similarity index is used, and then the influence of the nodes is compared so as to obtain the initial community division. In the second stage, the community is regarded as a node, and the PC is calculated again and sorted in ascending order. For determining the optimal parameter value in the weak community condition, the community is merged. Finally, the community structure is further improved based on the modularity optimization, and the final community division result is obtained. The proposed LPA-ITSLR algorithm solves the problem that the randomness of LPA-TS algorithm may yield unstable community partition results. Moreover, LPA-ITSLR yielded higher modularity than other algorithms on six real networks and 15 artificial datasets and achieved a more stable community division. However it has a higher time complexity in the case of certain networks with special community structures such as when the network community structure is complex, when there are many small communities and less contact between communities, and for nonequilibrium size distribution networks, a community detection method based on label propagation integrated deep learning and optimization could be employed to determine the node similarities and label influence. In the future, community detection in large-scale networks will be further studied to reduce the time complexity of the algorithm, and to achieve more accurate and efficient community detection results.

## Acknowledgements

This work was supported by the National Natural Science Foundation of China (42002138, 62172352), Natural Science Foundation of Heilongjiang Province (LH2019F042), postdoctoral scientific research development fund of Heilongjiang Province (No.LBH-Q20073) and Excellent Young and Middle-aged Innovative Team Cultivation Foundation of Northeast Petroleum University (KYCXTDQ202101).

## References

- (1) Aghaalizadeh S, Afshord S T, Bouyer A, et al. Improving the stability of label propagation algorithm by propagating from low-significance nodes for community detection in social networks[J]. Computing, 2021: 1-22. DOI: <https://doi.org/10.1007/s00607-021-01027-4>
- (2) Arinik N, Labatut V, Figueiredo R. 2021. Characterizing and comparing external measures for the assessment of cluster analysis and community detection. IEEE Access 9: 20255-20276 DOI: [10.1109/ACCESS.2021.3054621](https://doi.org/10.1109/ACCESS.2021.3054621).
- (3) Chakraborty T, Ghosh S, Park N. 2019. Ensemble-based overlapping community detection using disjoint community structures. Knowledge-Based Systems 163: 241–251 DOI: [10.1016/j.knosys.2018.08.033](https://doi.org/10.1016/j.knosys.2018.08.033).
- (4) DENG Kaixuan, CHEN Hongchang, HUANG Ruiyang. 2018. Improved LPA Algorithm Based on Label Propagation Ability. Computer Engineering 44(3):60–64.
- (5) Dey A K, Tian Y, Gel Y R. 2021. Community detection in complex networks: From statistical foundations to data science applications. Wiley Interdisciplinary Reviews: Computational Statistics (1) DOI: [10.1002/wics.1566](https://doi.org/10.1002/wics.1566).
- (6) Gregory S. 2010. Finding overlapping communities in networks by label propagation. New journal of Physics 12(10): 103018.
- (7) Gui Q et al. 2018. A community discovery algorithm based on boundary nodes and label propagation. Pattern Recognition Letters 109: 103–109 DOI: [10.1016/j.patrec.2017.12.018](https://doi.org/10.1016/j.patrec.2017.12.018).
- (8) Kouni IBE, Karoui W, Romdhane L B. 2020. Node Importance based Label Propagation Algorithm for overlapping community detection in networks. Expert Systems with Applications 162: 113020 DOI: [10.1016/j.eswa.2019.113020](https://doi.org/10.1016/j.eswa.2019.113020).
- (9) Kumar R et al. 2015. Fast greedy algorithms in mapreduce and streaming [J]. ACM Transactions on Parallel Computing (TOPC) 2(3): 1–22 DOI: [10.1145/2809814](https://doi.org/10.1145/2809814).
- (10) Li C, Chen H, Li T, et al. 2021. A stable community detection approach for complex network based on density peak clustering and label propagation. Applied Intelligence : 1–21.
- (11) Li C, Zhang Y. 2020. A personalized recommendation algorithm based on large-scale real micro-blog data. Neural Computing and Applications 32(1).
- (12) Li H et al. 2021. LPA-MNI: An Improved Label Propagation Algorithm Based on Modularity and Node Importance for Community Detection. Entropy 23(5): 497 DOI: [10.1142/S0217979215500290](https://doi.org/10.1142/S0217979215500290).
- (13) Li P Z et al. 2019. Edmot: An edge enhancement approach for motif-aware community detection [C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019: 479–487 DOI: [10.1145/3292500.3330882](https://doi.org/10.1145/3292500.3330882).
- (14) Li W et al. 2017. Stepping community detection algorithm based on label propagation and similarity. Physica A: Statistical Mechanics and its Applications 472: 145–155 DOI: [10.1016/j.physa.2017.01.030](https://doi.org/10.1016/j.physa.2017.01.030).



- (15) Lim J, Salzman O, Tsiotras P. An Incremental-Search Algorithm for Weighted Colored Graphs[C]//2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 6907-6913. DOI: [10.1109/IROS51168.2021.9636736](https://doi.org/10.1109/IROS51168.2021.9636736)
- (16) Liu K et al. 2015. Label propagation based evolutionary clustering for detecting overlapping and non-overlapping communities in dynamic networks. Knowledge-Based Systems 89: 487–496 DOI: [10.1016/j.knosys.2015.08.015](https://doi.org/10.1016/j.knosys.2015.08.015).
- (17) Lu M, Zhang Z, Qu Z, et al. 2018. LPANNI: Overlapping community detection using label propagation in large-scale complex networks. IEEE Transactions on Knowledge and Data Engineering 31(9): 1736–1749 DOI: [10.1109/TKDE.2018.2866424](https://doi.org/10.1109/TKDE.2018.2866424).
- (18) MA J et al. 2019. Overlapping community detection algorithm by label propagation using PageRank and node clustering coefficients. Journal of National University of Defense Technology 41(1): 183–190.
- (19) Martinet L E et al. 2020. Robust dynamic community detection with applications to human brain functional networks. Nature communications 11(1): 1–13.
- (20) Newman M E J. 2006. Finding community structure in networks using the eigenvectors of matrices. Physical review E 74(3): 036104 DOI: [10.1103/PhysRevE.74.036104](https://doi.org/10.1103/PhysRevE.74.036104).
- (21) Sun P G, Miao Q, Staab S. 2021. Community-based k-shell decomposition for identifying influential spreaders. Pattern Recognition 120: 108130 DOI: [10.1016/j.patcog.2021.108130](https://doi.org/10.1016/j.patcog.2021.108130).
- (22) Tommasel A, Godoy D. 2018. Multi-view community detection with heterogeneous information from social media data. Neurocomputing 289: 195–219 DOI: [10.1016/j.neucom.2018.02.023](https://doi.org/10.1016/j.neucom.2018.02.023).
- (23) Tong C et al. 2015. Weighted label propagation algorithm for overlapping community detection [C]//2015 IEEE International Conference on Communications (ICC). IEEE, 2015: 1238–1243 DOI: [10.1109/ICC.2015.7248492](https://doi.org/10.1109/ICC.2015.7248492).
- (24) Wang T et al. 2020. Label propagation algorithm based on node importance. Physica A: Statistical Mechanics and its Applications 551: 124137 DOI: [10.1016/j.physa.2020.124137](https://doi.org/10.1016/j.physa.2020.124137).
- (25) WU QS et al. 2020. Overlapping community detection algorithm integrating label preprocessing and node influence. Journal of Computer Applications 40(12): 3578–3585.
- (26) Wu T et al. 2016. Integrated structure investigation in complex networks by label propagation. Physica A: Statistical Mechanics and its Applications 448:68–80 DOI: [10.1016/j.physa.2015.12.073](https://doi.org/10.1016/j.physa.2015.12.073).
- (27) Xie J, Szymanski B K, Liu X. 2011. Slpa: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process [C]//2011 IEEE 11th international conference on data mining workshops. IEEE 2021: 344–349 DOI: [10.1109/ICDMW.2011.154](https://doi.org/10.1109/ICDMW.2011.154).
- (28) Xing Y et al. 2014. A node influence based label propagation algorithm for community detection in networks. The Scientific World Journal 2014 DOI: [10.1155/2014/627581](https://doi.org/10.1155/2014/627581).
- (29) Xu G, Guo J, Yang P. 2020. TNS-LPA: An Improved Label Propagation Algorithm for Community Detection Based on Two-Level Neighbourhood Similarity. IEEE Access 9: 23526–23536 DOI: [10.1109/ACCESS.2020.3045085](https://doi.org/10.1109/ACCESS.2020.3045085).
- (30) YANG Xiaojing. 2020. Community detection algorithm based on node influence and similarity D. Fujian: Quanzhou. Thesis, Huaqiao University.
- (31) Yang Z, Perotti J I, Tessone C J. 2017. Hierarchical benchmark graphs for testing community detection algorithms. Physical review E 96(5): 052311 DOI: [10.1103/PhysRevE.96.052311](https://doi.org/10.1103/PhysRevE.96.052311).
- (32) Yin F et al. 2018. Mining and Prediction of the Core Group in the Communication Network Based on Walk Trap and ARIMA Algorithm [C]//2018 11th International Symposium on Computational Intelligence and Design (ISCID). IEEE 1: 88–92 DOI: [10.1109/ISCID.2018.00027](https://doi.org/10.1109/ISCID.2018.00027).

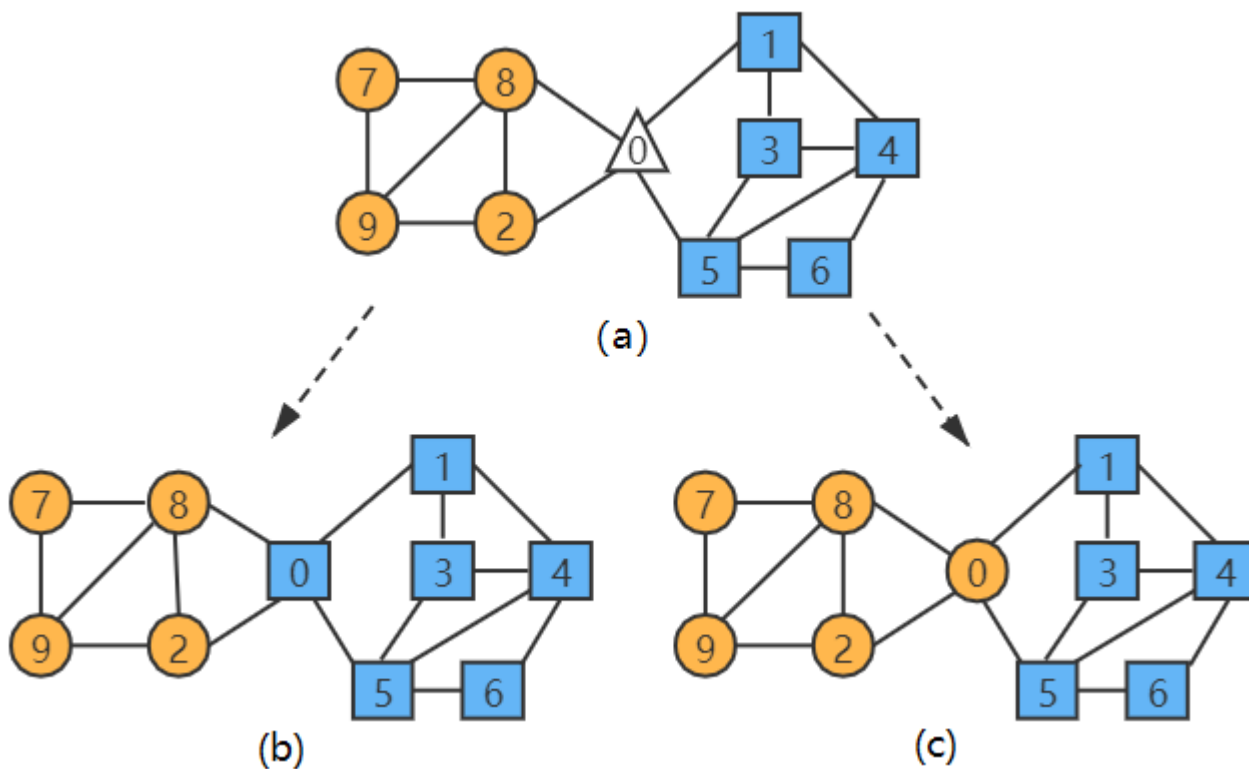
- (33) Yuan Q, Liu B. 2021. Community detection via an efficient nonconvex optimization approach based on modularity. *Computational Statistics & Data Analysis* 157: 107163 DOI: [10.1016/j.patrec.2017.12.018](https://doi.org/10.1016/j.patrec.2017.12.018).
- (34) ZHAI Zhenxin, YU Yuecheng, GU Yu. 2021. Community discovery algorithm combining leaderRank and tag propagation. *Computer and Digital Engineering* 49(5): 942–946.
- (35) Zhang J, Xia Z. A New Novel Label Propagation Algorithm[C]//International Conference on Web Information Systems and Applications. Springer, Cham, 2021: 39-46. DOI: [https://doi.org/10.1007/978-3-030-87571-8\\_4](https://doi.org/10.1007/978-3-030-87571-8_4)
- (36) Zhang P. 2015. A revisit to evaluating accuracy of community detection using the normalized mutual information. *arXiv preprint arXiv:1501.03844*.
- (37) Zhang X K et al. 2017. Label propagation algorithm for community detection based on node importance and label influence. *Physics Letters A* 381(33): 2691–2698 DOI: [10.1016/j.physleta.2017.06.018](https://doi.org/10.1016/j.physleta.2017.06.018).
- (38) Zhang Y et al. 2020. Lilpa: A label importance based label propagation algorithm for community detection with application to core drug discovery. *Neurocomputing* 413: 107–133 DOI: [10.1016/j.neucom.2020.06.088](https://doi.org/10.1016/j.neucom.2020.06.088).
- (39) ZHENG WP et al. 2018. A two-stage community detection algorithm based on label propagation [J]. *Computer research and Development* 55(09):1959–1971 DOI: [10.7544/issn1000-1239.2018.20180277](https://doi.org/10.7544/issn1000-1239.2018.20180277).
- (40) Zhou X et al. 2019. A novel modularity-based discrete state transition algorithm for community detection in networks. *Neurocomputing* 334: 89–99 DOI: [10.1016/j.neucom.2019.01.009](https://doi.org/10.1016/j.neucom.2019.01.009).



# Figure 1

A network instance with two communities

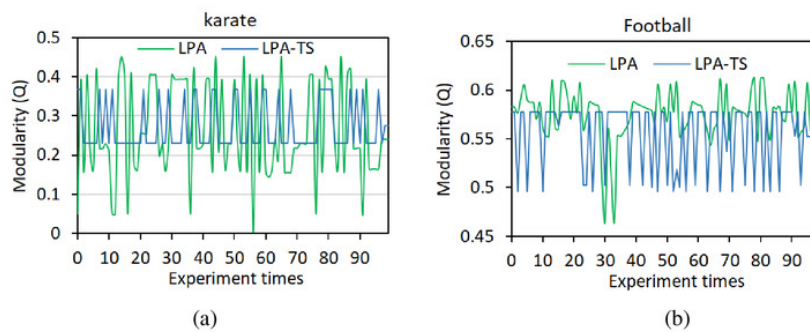
The nodes in different communities are represented by different shapes and colors in Figure 1.



## Figure 2

Results of 100 experiments of the two algorithms on two networks

LPA and LPA-TS were tested 100 times on two different networks

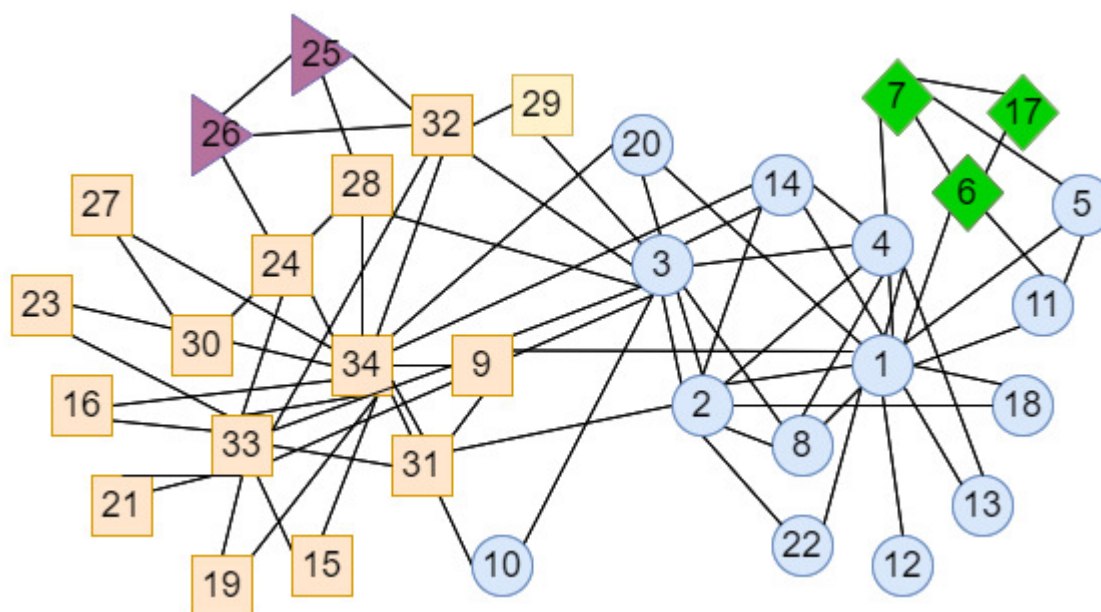


**Fig. 2** Results of 100 experiments of the two algorithms on two networks

# Figure 3

LPA-TS algorithm partitioning results for Karate network in the first stage

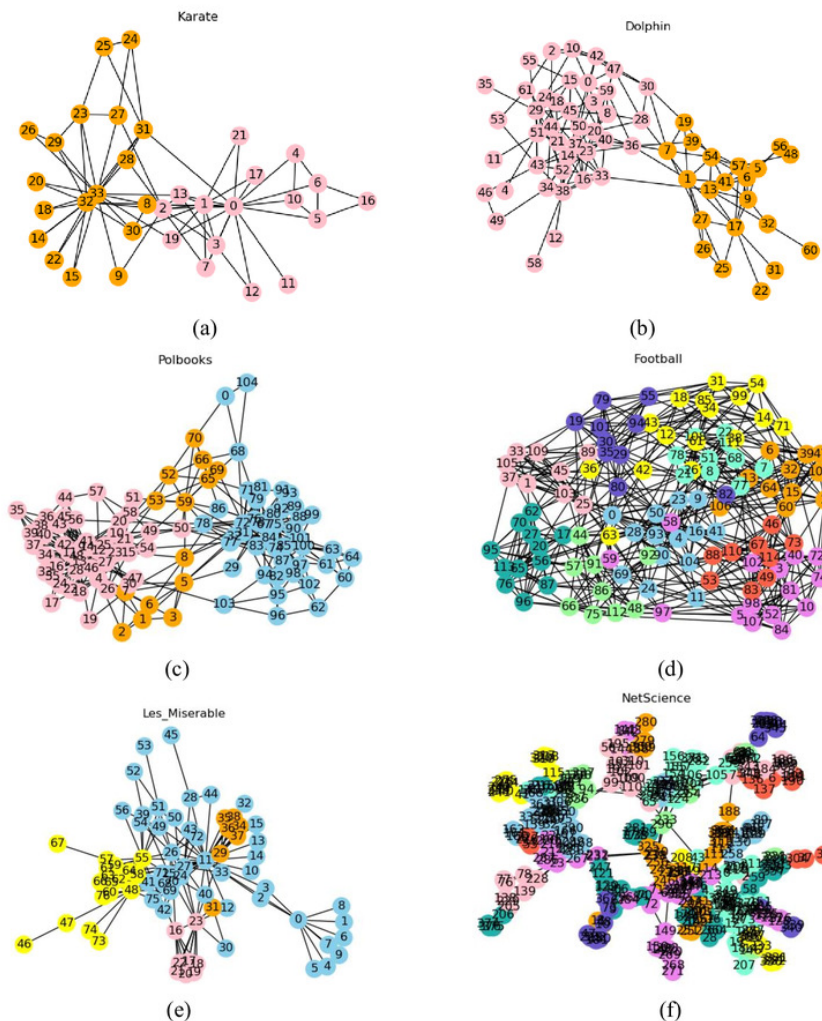
Different colors represent different neighborhoods. The LPA-TS algorithm only yielded an initial community division result in the first stage, and there were still many small communities. In the Karate network shown in Figure 3,



# Figure 4

Community partition results of real networks

The partition results of this algorithm on six real data sets are shown

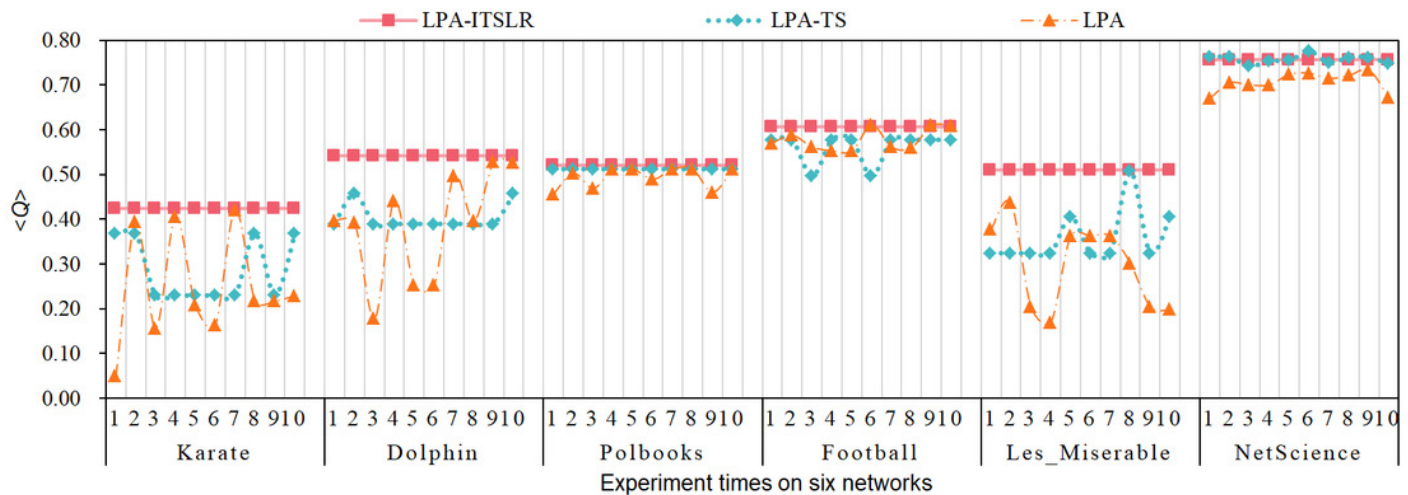


**Fig. 4** Community partition results of real networks

# Figure 5

Comparison of algorithm stability

The modularity comparison of three algorithms on six networks

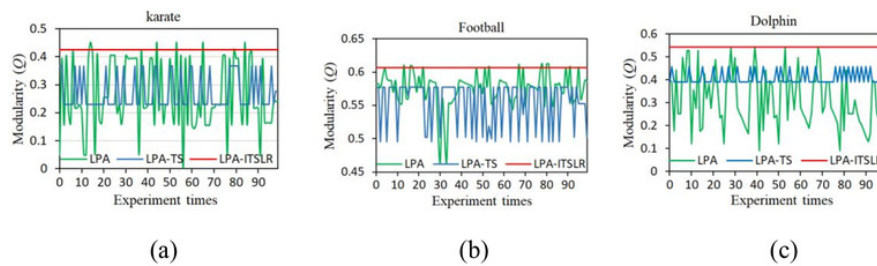


## Figure 6

Comparison of modularity of LPA, LPA-TS, and LPA-ITSLR

The results of 100 experiments were compared on Karate, Football and Dolphin networks





**Fig. 6** Comparison of modularity of LPA, LPA-TS, and LPA-ITSLR

# Figure 7

Comparison of modularity and NMI on eight synthetic datasets

The modularity and NMI results of the three algorithms under different  $\mu$  values were compared

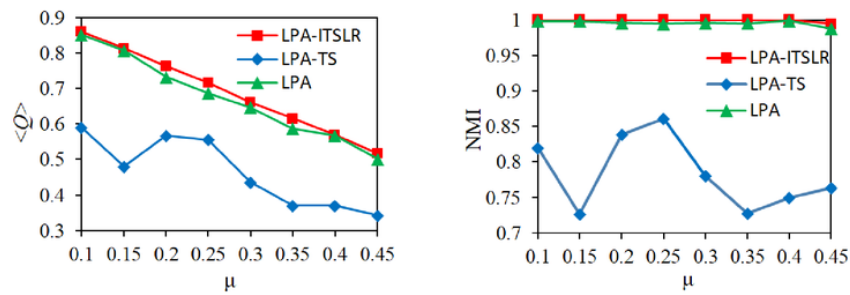


Fig. 7 Comparison of modularity and NMI on eight synthetic datasets

**Table 1** (on next page)

First stage of the LPA-ITSLR algorithm

**Table 1 First stage of the LPA-ITS LR algorithm**

---

**Step 1:** Initial community discovery

---

**Input:**  $G(V, E)$

**Output:** Label of each node  $v_i \in V, L(V) = \{l_1, l_2, \dots, l_q\}$

- 1: assign a unique label to each node in the  $G$ ;
  - 2: **for** each  $v_i, v_j \in V$ , calculate  $S_{(i,j)}^{ICN}$  according to Formula (9), **end for**;
  - 3:  $U \leftarrow$  Calculate the  $PC_i$  for each node according to Formula (1);
  - 4: **Sort** ( $U$ ) by  $PC_i$  in ascending order;
  - 5: **for** each  $v_i \in U$  :
    - 6:  $S \leftarrow \arg \max S_{(i,j)}^{ICN}$ , where  $v_j \in N_i$ ;
    - 7: **if**  $|S| \neq 1$ :
      - 8:  $L \leftarrow \arg \max LR(v_j)$ , where  $v_j \in S$
      - 9: **if**  $|L| \neq 1$ :  $k = \text{random}(|L|)$ ,  $l_i \leftarrow l_k$ ;
      - 10: **else**:  $l_i \leftarrow l_L$ ;
      - 11: **else**:  $l_i \leftarrow l_S$ ;
    - 12: **end for**
-

## Table 2 (on next page)

Second stage of the LPA-ITSLR algorithm

**Table 2 Second stage of the LPA-ITSLR algorithm**

---

**Step 2:** Community merge

---

**Input:**  $L(V) = \{l_1, l_2, \dots, l_q\}$

**Output:**  $\Omega = \{\Omega_1, \Omega_2, \dots, \Omega_k\}$

```

1:  for  $l_i$  in  $L(V)$  :
2:      if  $(\alpha * \sum_{j \in l_i} d_j^{in}(l_i) > \sum_{j \in l_i} d_j^{out}(l_i)) == false$  :
3:           $l_i = l_j$ ; (where  $l_j = \max \sum_{v_i \in l_i, v_j \in l_j} a_{ij}$ )
4:      end for
5:  get the community set  $\Omega' = \{\Omega_1^{(0)}, \Omega_2^{(0)}, \dots, \Omega_k^{(0)}\}$ ,  $n' = |\Omega'|$ ,  $t = 0$  ;
6:  take each  $\Omega_i^{(t)}$  as a node  $s_i$  ;
7:  take the number of edges between nodes as weight ;
8:  get the new network  $G' = (V', E')$ , where  $V' = (s_1, s_2, \dots, s_{n'})$  ;
9:   $t = 0$  ; Calculate  $Q(t)$  ;
10: do:
11:      $s_m = \arg \max PC(s_i)$  ;
12:      $j = \operatorname{argmax} \omega_{mj}$ ; (where  $\omega_{mj} = \sum_{v_i \in s_m, v_j \in s_j} a_{ij}$ ,  $s_j \in V'$ ,  $j \neq i$ )
13:     try merge  $\Omega_m^{(t)}$  and  $\Omega_j^{(t)}$  and Calculate  $Q(t+1)$  ;
14:     if  $(Q(t+1) > Q(t))$  :
15:         merge  $\Omega_m^{(t)}$  and  $\Omega_j^{(t)}$ ;  $t = t + 1$ ,  $n' = n' + 1$ ;
16: while  $(Q(t+1) > Q(t))$ 
17: return  $\Omega^{t+1}$ 

```

---

### **Table 3**(on next page)

Basic structural parameters of real datasets

six classic real social network datasets were used in the experiment; their attribute characteristics are presented in Table 3.



**Table 3 Basic structural parameters of real datasets**

Dataset	$ V $	$ E $	$ \Omega $	$\max(k)$	$\langle k \rangle$	$\langle d \rangle$	$\langle c \rangle$
Karate	34	78	2	17	4.588	2.408	0.588
Dolphin	62	159	2	12	5.129	—	0.309
Polbooks	105	441	3	25	8.4	3.079	0.448
Football	115	613	12	12	10.661	2.508	0.403
Les_Miserable	77	254	—	36	6.597	2.641	0.736
NetScience	379	914	16	34	3.451	6.042	0.798

# **Table 4**(on next page)

Average modularity values of 10 experiments for the three algorithms on real datasets

**Table 4 Average modularity values of 10 experiments for the three algorithms on real datasets**

Dataset/ $\langle Q \rangle$	LPA	LPA-TS	LPA-ITSLR
Karate	0.3174	0.3716	<b>0.4242</b>
Dolphin	0.4920	0.3759	<b>0.5418</b>
Polbooks	0.3801	0.4569	<b>0.5207</b>
Football	0.5819	0.6010	<b>0.6068</b>
Les_Miserable	0.2719	0.5007	<b>0.5102</b>
NetScience	<b>0.7769</b>	0.7573	0.7567

# **Table 5**(on next page)

Modularity comparison of five algorithms

**Table 5 Modularity comparison of five algorithms**

Network	Karate	Dolphin	Polbooks	Football
COPRA	0.2348 $\pm$ 0.10187	0.3741 $\pm$ 0.03946	0.4884 $\pm$ 0.03215	0.5972 $\pm$ 0.02115
WLPA	0.3682 $\pm$ 0.08176	0.3695 $\pm$ 0.02517	0.5070 $\pm$ 0.00622	0.5981 $\pm$ 0.01374
LINSIA	0.3989 $\pm$ 0.00004	0.3878 $\pm$ 0.00005	0.4521 $\pm$ 0.00007	0.5853 $\pm$ 0.00007
LILPA	0.4213 $\pm$ 0.0029	0.4003 $\pm$ 0.00214	0.4635 $\pm$ 0.00646	0.6061 $\pm$ 0.00151
LPA-ITSLR	<b>0.4242</b>	<b>0.5418</b>	<b>0.5207</b>	<b>0.6068</b>

# **Table 6**(on next page)

Results of eight algorithms on classical networks

1

**Table 6 Results of eight algorithms on classical networks**

Network Criteria	Karate		Dolphin		Football	
	$ \Omega $	$Q$	$ \Omega $	$Q$	$ \Omega $	$Q$
Fastgreedy	3	0.38	4	0.495	6	0.549
LPA	2	0.292	3	0.492	9	0.576
Leading Eigenvector	4	0.393	5	0.491	8	0.492
Walktrap	5	0.353	4	0.489	10	0.602
NIBLPA	3	0.352	5	0.452	9	0.542
EdMot	3	0.412	4	0.518	9	0.604
LPA-MNI	2	0.372	4	0.527	11	0.582
LPA-ITSLR	<b>2</b>	<b>0.4242</b>	<b>2</b>	<b>0.5418</b>	<b>10</b>	<b>0.6068</b>

2

**Table 7** (on next page)

Description of synthetic networks



**Table 7 Description of synthetic networks**

Network	$ V $	$\langle k \rangle$	$\max(k)$	$\min \Omega $	$\max \Omega $	$\mu$
LFR-1 ~ LFR-8	1000	20	50	10	50	0.1~0.45
LFR-9	5000	10	50	50	50	0.1
LFR-10	5000	10	50	50	50	0.3

## Table 8 (on next page)

### Results for LFR9 and LFR10

For large-scale artificial networks LFR-9 and LFR-10 with high complexity, the proposed LPA-ITSLR algorithm was compared with seven recent label propagation algorithms for community division.  $Q$  and  $NMI$  were considered as evaluation parameters.

1

**Table 8 Results for LFR9 and LFR10**

Criteria Network	NMI		$\mathcal{Q}$	
	LFR-9	LFR-10	LFR-9	LFR-10
COPRA	0.9853 $\pm$ 0.00466	0.9859 $\pm$ 0.00413	0.4259 $\pm$ 0.00786	0.3353 $\pm$ 0.00286
SLPA	<b>0.9994 <math>\pm</math> 0.00081</b>	0.9931 $\pm$ 0.00352	0.4467 $\pm$ 0.00122	0.3437 $\pm$ 0.00193
LINSIA	0.8813 $\pm$ 0.00000	0.8267 $\pm$ 0.00007	0.3221 $\pm$ 0.00007	0.3107 $\pm$ 0.00007
DLPA+	0.9887 $\pm$ 0.00135	0.9414 $\pm$ 0.00156	0.4423 $\pm$ 0.00164	0.3381 $\pm$ 0.00074
WLPA	0.9980 $\pm$ 0.00113	<b>0.9979 <math>\pm</math> 0.00111</b>	0.4443 $\pm$ 0.00174	0.3366 $\pm$ 0.00145
LPA_NI	0.9987 $\pm$ 0.00082	0.9847 $\pm$ 0.00124	0.4467 $\pm$ 0.00024	0.3437 $\pm$ 0.00112
LILPA	0.9955 $\pm$ 0.00084	0.9692 $\pm$ 0.00115	0.4472 $\pm$ 0.00011	0.3453 $\pm$ 0.00041
LPA-ITSLR	0.9862	0.9531	<b>0.8782</b>	<b>0.8091</b>

2

**Table 9**(on next page)

Actual number of communities and the number of communities detected by LPA-ITSLR

The number of real communities and algorithm division

**Table 9 Actual number of communities and the number of communities detected by LPA-ITSLR**

LFR networks	Actual number of communities	Number of communities divided by LPA- ITSLR
LFR-1	35	40
LFR-2	35	35
LFR-3	38	38
LFR-4	45	45
LFR-5	39	39
LFR-6	42	42
LFR-7	42	42
LFR-8	42	40
LFR-9	85	81
LFR-10	98	69

**Table 10**(on next page)

Table 10 Community detection results of five large-scale artificial networks

The parameters and experimental results of 5 large data sets are presented

1

**Table 10 Community detection results of five large-scale artificial networks**

Dataset	$ V $	$\langle k \rangle$	$\max(k)$	$\min \Omega $	$\max \Omega $	$\mu$	actual number of communities	number of communities found	$\langle Q \rangle$	NMI
LFR-11	6000	10	50	30	60	0.1	125	128	0.8730	0.9762
LFR-12	7000	10	50	30	60	0.1	130	133	0.8686	0.9510
LFR-13	8000	10	50	30	60	0.1	176	176	0.8828	0.9805
LFR-14	9000	10	50	30	60	0.1	175	178	0.8722	0.9629
LFR-15	10000	10	50	30	60	0.1	175	180	0.8645	0.9478