IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 June 2023, accepted 19 July 2023, date of publication 27 July 2023, date of current version 3 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3299289

RESEARCH ARTICLE

Multiscale Community Detection Using a Label Propagation-Based Clustering Method in Complex Networks

XUE ZHENG, DONGQIU XING, KEBIN CHEN, JING ZHAO, AND YUNJUN LU
College of Information and Communication, National University of Defense Technology, Wuhan 430019, China

Corresponding author: Kebin Chen (chenkebin17@nudt.edu.cn)

This work was supported by the National Social Science Foundation of China under Grant 2022-SKJJ-C-084.

ABSTRACT Multiscale community detection algorithms can reveal the hierarchy of complex networks. However, the existing algorithms are unable to realize full-resolution community detection, and the hierarchy structure of the obtained community is overidealized. Aiming at these problems, we propose an algorithm named label propagation algorithm with multiscale community detection (LPAMCD), which introduces a two-phase propagation process and a tunable parameter, called the belonging coefficient threshold, into the label propagation algorithm to realize full-resolution community detection. Moreover, LPAMCD has the ability to find the mechanism of dynamic confrontation between adjacent communities in absorbing boundary nodes, which implies that the community hierarchy of social networks is not an idealized dendrogram. The extensive experimental results with real networks show that LPAMCD can detect community structures at full resolution scales with high accuracy and stability. Furthermore, the novel finding of dynamic confrontation is demonstrated in the experiments.

INDEX TERMS Label propagation, multiscale community detection, full resolution, dynamic confrontation.

I. INTRODUCTION

Research on complex networks, such as social networks [1], [2], citation networks [1], biochemical networks [3], and scientific cooperation networks [4], [5], has revealed many critical laws and characteristics in scientific fields. Community detection is always a hot topic, especially in social complex networks that have modular structures. The nodes in a community are connected more densely than the nodes outside the network [6], inferring that the nodes within the same community usually have similar properties, for example, similar interests in social networks, the same discipline in citation networks and a common function in biochemical networks [7]. Therefore, identifying the community structure in complex networks is significant and helpful for analyzing modular characteristics [8].

In the real world, the community boundaries in networks, whether they are social, citation, biochemical or others, are usually elastic when considered from different perspectives. The community structure may be quite different when detected at multiple levels [2]. For example, in a scientific cooperation network, scientists belonging to the same community usually study similar scientific concepts or have common research interests [5]. Scholars can be divided into communities at different levels according to the first-class disciplines, second-class disciplines and research directions. In protein networks, nodes can be divided into functional modules at different levels according to the different functions of the organisms concerned. Similar situations are more common in social networks. Different detection scales will cluster people with different degrees of intimacy, interests, *etc.* [1], [2] Multiscale community detection can reveal the hierarchy of networks and the dynamic confrontation between adjacent communities, which enrich the concept of community structures [9]. Therefore, it is of practical significance to explore an efficient method to detect multiscale communities in complex networks.

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

However, multiscale community detection is still a challenging task, as the size and number of communities can vary simultaneously when viewed from different perspectives [10]. In recent years, many algorithms have been developed to detect multiscale community structures based on various approaches, *e.g.*, modularity optimization [3], [11], fitness functions [6], [12], [13], coalition formation game theory [14], density-based clustering [15] and DeepWalk [10], [12]. However, there are still three essential challenges for these algorithms. For example, some algorithms based on optimizing quality functions may suffer from the resolution limit problem and need prior knowledge on the sizes of communities, especially those using modularity [14]. Some algorithms cannot detect overlapping communities. Some algorithms using global optimization have high computational complexity and instability, resulting in low efficiency, especially when detecting large-scale networks. More importantly, in current studies, very little is known about the mechanism of dynamic confrontation between adjacent communities in absorbing boundary nodes, which means that the community boundary may vary with different detection resolutions, as adjacent communities have different levels of influence on the boundary nodes.

Motivated by epidemic spreading [16], the label propagation algorithm (LPA) is a good candidate for detecting overlapping communities at multiple scales quickly and effectively. By iteratively propagating different labels among nodes in a network until convergence, LPA can detect communities constructed by nodes with the same labels [17] without prior knowledge on the size and number of communities. This method has an obvious advantage of linear time complexity, leading to a high-speed and efficient community detection process in complex networks. Moreover, the label propagation process can fully reflect the mutual influence between nodes and the local characteristics of the network. Therefore, an increasing number of algorithms based on LPA have been used to detect nonoverlapping or overlapping communities [18], [19], [20]. Nevertheless, the original LPA has some shortcomings: (1) it only detects communities with fixed boundaries, which means it cannot be applied to multiscale community detection; and (2) the detection results may be unstable due to randomness during label propagation.

To fill these gaps, we propose an integrated framework named LPAMCD (label propagation algorithm with multiscale community detection) to detect multiscale communities in complex networks. The main contributions of the paper are as follows:

(1) We define the Node Importance (*NI*), Degree Influence (*Dinf*) and Propagation Indicator (*PI*) measures to comprehensively consider the influence of neighboring nodes in label propagation. By fully utilizing these three new measures, the multiscale community detection accuracy can be enhanced.

(2) We propose a novel label propagation strategy to realize full-resolution community detection. Based on an introduced tunable parameter named the belonging coefficient, our algorithm has a unique capability to detect communities on any scale continuously and stably.

(3) Through a number of experiments, we establish a novel mechanism of dynamic confrontation between adjacent communities in absorbing boundary nodes in social networks. This reflects that the variations in the community boundary may be nonlinear with detection scale changes.

This paper is organized as follows: Section II reviews the literature. Section III describes several important terms, and a new label update method is defined. Section IV introduces the proposed LPAMCD approach in detail. Section V shows the experimental results. Finally, we summarize the conclusions of this paper.

II. RELATED WORK

A. MULTIRESOLUTION COMMUNITY DETECTION

Community structure has been found to be a significant feature in many networks. The nodes within the same community are densely connected with each other, while they are sparsely connected with the nodes in the rest of the networks. It has been proven that the community structure in a network is not unique when the network is viewed from different perspectives [3], [11], [21].

Various methods have been introduced to realize multiresolution community detection. Some algorithms are based on the relevance between dynamics and the multiscale structures of networks [2], [22], [23], [24]. Some algorithms optimize resolution based on the Potts model [25], [26]. Some algorithms are based on local optimization of the quality function [6], [27]. Some algorithms make use of novel strategies. For example, considering the computational efficiency, Lin introduced an approach to detect hierarchical communities based on integer programming [28]. Felfli presented a density-based clustering strategy using the hill-climbing procedure to unveil community structures [15]. Li proposed a framework to detect hierarchical communities with the focused crawler algorithm [29].

Most of these studies are unable to achieve full resolution community detection, and the community structure obtained is overidealized, especially in social networks. However, our algorithm has the capability to detect communities at any scale continuously and stably. In reality, the community size may vary nonlinearly with a change in the detection scale, and the community cannot be considered a simple combination of subgroups. Our algorithm reveals a novel mechanism of the dynamic confrontation between adjacent communities in absorbing boundary nodes, which indicates that the community hierarchy of social networks is not an idealized dendrogram.

B. LABEL PROPAGATION ALGORITHM

The label propagation algorithm (LPA) is a graph-based semi-supervised learning algorithm [17]. Due to its high speed and efficiency, many researchers have applied it to community detection. In this algorithm, different labels are

assigned to nodes, and then the labels iteratively spread from labeled nodes to unlabeled nodes. After the iterations end, the communities constructed by the nodes with the same labels are detected. However, the original LPA can only detect nonoverlapping communities. It also has the problem of instability, as the initial label assignment and propagation process are random. Some advanced versions of LPA have been proposed to not only detect overlapping community structures but also ensure the stability and accuracy of the results [1], [19]. Nevertheless, no LPA algorithm capable of detecting multiscale communities has been found.

SLPA [16], [30] is a traditional variant of LPA that uses the speaker-listener rule during label propagation to detect overlapping communities. The labels of each node are stored in memory during each iteration. When iterations finish, the frequency of one label appearing in the memory determines the probability of this node belonging to the corresponding community. To detect overlapping communities, a tunable parameter γ should be predefined. However, when updating node labels, SLPA treats the importance of neighboring nodes the same, which may bring high randomness during label propagation and lead to low accuracy and stability. Furthermore, SLPA is not able to detect communities at multiple scales.

COPRA [31] is also a popular advanced version of LPA that can detect overlapping communities. It introduces a parameter v to limit the number of labels of each node. With the value of the belonging coefficient, the strength that a node belongs to a community is determined. During the iteration process of label propagation, the labels and the corresponding belonging coefficients of nodes are updated. However, similar to SLPA, COPRA also has the problem of instability and cannot detect multiscale communities.

To detect overlapping communities while removing the instability issue, Li proposed a parameter-free algorithm based on LPA, named SFLPA [1]. Initially, it identifies nodes' roles, among which the hub nodes are considered potential community centers. Then, taking hub nodes as the starting points, the labels propagate through nodes from high to low according to centrality scores. The community hierarchy is revealed after aggregating small communities. This algorithm eliminates instability and improves speed by integrating the theory of seed set expansion into label propagation, which is also adopted in our algorithm in this paper. However, when identifying hub nodes and propagating labels, only the centrality score is considered, and the connection and similarity between neighboring nodes is neglected. This may decrease the accuracy of the community detection results.

Among the existing LPA algorithms, there is no algorithm that can realize multiscale community detection, including the algorithms introduced above. In addition, although most label propagation algorithms hold the advantage of linear time complexity, assuring both stability and accuracy simultaneously is still a critical problem. Considering these issues, the main innovations of LPAMCD proposed in this paper are as follows: (1) it presents a new metric, the propagation indicator (*PI*), and a new function, the belong coefficient *(b(l, x))*, to legitimately propagate labels among nodes, thus simultaneously increasing both the stability and accuracy during overlapping community detection; and (2) it introduces a parameter b_{th} as the threshold of the belonging coefficient to realize multiscale community detection.

III. LABEL PROPAGATION UPDATE STRATEGY

In traditional multiscale community detection strategies, the community hierarchy is usually fixed and unique, such as the structure of a dendrogram. However, these algorithms have different problems, for example, they have trouble detecting overlapping communities, there are resolution limit issues for algorithms based on modularity, and these algorithms have high computational complexity. Therefore, this paper introduces an enhanced LPA algorithm to detect multiscale overlapping communities. Traditional LPA algorithms have instability problems, including SLPA and COPRA, and no LPA algorithm for multiscale community detection has been produced. Based on this, we propose LPAMCD to detect multiscale overlapping community structures and simultaneously improve the stability and efficiency. In particular, the community hierarchy detected by our algorithm may not be a dendrogram due to the nonlinear variation in the community size.

In this section, three items are identified first: Node Importance, Degree Influence and Propagation Indicator. Then, the proposed label update method based on the Propagation Indicator is described. The notations in the problem formulation and equations are described in Table 1.

A. DEFINITION OF ITEMS

*Definition 1 ((Node Importance (**NI**)): Node Importance indicates the possibility that one node acts as the potential community center.*

In a given network $G = (V, E)$, where V represents vertices (or nodes) in the network G and E represents edges in G, there is one node $x \in V$, and the neighbor nodes of x comprise a node set $Neg(x)$. Generally, node x is identified as a potential community center when it has the following two features: (1) $Neg(x)$has more node elements; and (2) the nodes in $Neg(x)$connect with each other densely [1], [19], [32]. That is, when a node has a high degree and a high clustering coefficient, it may be the center of a community. We propose that these kinds of nodes are more important in networks and can influence their neighboring nodes. Based on this, the degree and clustering coefficient of node $x$are used to measure its importance. The Node Importance equation is:

$$NI(x) = D(x)^{*}CC(x) \tag{1}$$

where $D(x)$ denotes the degree of node $x$and $CC(x)$ denotes the clustering coefficient of node x. The larger the NI of a node is, the more likely it is to become the center of a community.

TABLE 1. Meanings of the notations.

Notations	Meaning
G	A network
V	Set of vertices (or nodes) in G
E	Set of edges in G
$NI(x)$	Node Importance of node x
$D(x)$	Degree of node x
$CC(x)$	Clustering coefficient of node x
$Dinf(x, y)$	Degree Influence from node x to node y
$Sim(x, y)$	Similarity value between node x and node y
$Neg(x)$	Neighbor node-set of node x
$PI(x, y)$	Probability of a label propagating from node x to node y
l	Label l in G
$b(l, x)$	Probability of node x belonging to label l
b_{th}	Belonging coefficient threshold
lab_i	Set of the labels of node i
LAB	Set of lab_i
$N(l)$	Set of the nodes with label l in $Neg(x)$
$N(L_{negx})$	Set of the nodes with any label in $Neg(x)$
c_i	Name of community
C	Set of community in G
t	Timestamps of label propagation
$Source$	Set of hub nodes
P	Set of pending nodes
$pQueue$	Sort P in descending order according to NI
EQ	Quality function extension of modularity to evaluate the goodness of community partition quality
NC	Number of communities

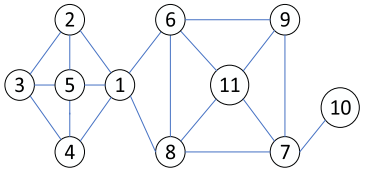


FIGURE 1. Sample network.

Taking the network shown in Fig. 1 as an example, the values of the degree, clustering coefficient and NI of each node are listed in Table 2. Node 5 and node 11 have the highest NI values compared with their neighbors, so these two nodes are potential community centers.

*Definition 2 ((Degree Influence (**Dinf**)): Degree Influence indicates the label propagation probability from one node to another under the influence of the degree difference between the two nodes.*

Nodes with a higher degree usually have a stronger influence on label propagation to their neighbors [1]. For example, in scientist cooperation networks, prominent scientists have a stronger effect on their neighbors; in friendship networks, such as Facebook and Twitter, internet celebrities have the most influence on their fans. Therefore, Degree Influence *(Dinf)* is used to assist the evaluation of label propagation, and we use equation (2) to calculate the degree influence from one node to another.

$$Dinf(x, y) = \frac{D(x)}{D(x) + D(y)} \tag{2}$$

where $Dinf(x, y)$ denotes the degree of influence from x to y. The larger the degree of node x is compared to that of node y, the higher the value of $Dinf(x, y)$ is, meaning that it is more likely for the label of node x to have an effect on node y. According to the sample network in Fig. 1, $Dinf(1,2)=0.625$ and $Dinf(2,1)=0.375$, so the ability of labels propagating from node 1 to node 2 is stronger than the reverse.

*Definition 3 ((Propagation Indicator (**PI**))): The propagation indicator reflects the capability of a label propagating from one node to another.*

We suggest that except for $Dinf(x, y)$, the similarity between nodes also affects the label propagation possibility. Nodes with higher similarity are more likely to belong to the same community. For example, in a scientist collaboration network, scientists with stronger professional similarity are more likely to be assigned to the same community; in a friendship network, people with more common interests are more likely to belong to the same community. Thus, the Propagation Indicator mainly consists of the similarity between nodes and the Degree Influence *(Dinf)* from one node to another. If a node has a higher similarity and *Dinf*, it should have a stronger label propagation capability.

We use the popular function named Jaccard Index [33], which is shown in (3), to evaluate the similarity between two nodes.

$$Sim(x, y) = \frac{Neg(x) \cap Neg(y)}{Neg(x) \cup Neg(y)} \tag{3}$$

where $Sim(x, y)$ is the similarity value between nodes x and y. $Neg(x)$ and $Neg(y)$ denote the neighbor node sets of nodes x and y, respectively. Therefore, the similarity of nodes x and y is the ratio of the number of their common neighbors to the number of all neighbors.

By combining the Jaccard Index and Degree Influence, the formula of the Propagation Indicator can be expressed as follows.

$$PI(x, y) = 0.5^*(\frac{Neg(x) \cap Neg(y)}{Neg(x) \cup Neg(y)} + \frac{D(x)}{D(x) + D(y)}) \tag{4}$$

where $PI(x, y)$ denotes the probability of a label propagating from node x to node y. As the value range of $Sim(x, y)$ and $Dinf(x, y)$ is 0 to 1, their summation is multiplied by 0.5 for normalization. In conclusion, $PI(x, y)$ evaluates the possibility of label propagation, and its value is in the range of 0 to 1.

B. LABEL UPDATE METHOD

Inspired by the idea in COPRA, we use the belonging coefficient $(b(l, x))$ to control the label propagation between nodes. $b(l, x)$ indicates the probability of node $x$belonging to label l. For a given network $G = (V, E)$, when the iteration of label propagation ends, there will be a set $LAB$to record the labels for each node $x \in V$.

TABLE 2. NI for the sample network in figure 1.

Node ID	1	2	3	4	***5***	6	7	8	9	10	***11***
Degree	5	3	3	3	***4***	4	4	4	4	2	***4***
Clustering	0.3	0.66	0.66	0.66	***0.66***	0.5	0.5	0.5	0.5	1	***0.66***
NI	1.5	2	2	2	***2.66***	2	2	2	2	2	***2.66***

In our algorithm, different labels propagate simultaneously, meaning that during each iteration of the propagation process, different labels spread from the newest updated nodes synchronously. Compared with the algorithms proposed in the literature [14], [19], [32], which propagate labels from all the sorted nodes in the network in turn, our method improves the propagation efficiency. Propagating labels from the potential community centers can make the labels spread far enough to form integrated communities, thus improving the accuracy of the detected community structure.

To realize multiscale community detection, we introduce a parameter named the belonging coefficient threshold (b_{th}). Only when $b(l, x)$ is larger than b_{th}, can label l be propagated to node x successfully. By adjusting the value of b_{th}, the community structures can be detected at different scales. Specifically, when the b_{th} value is low, the $b(l, x)$ value of each node more easily meets the requirement of label propagation, which is greater than the threshold b_{th}. Thus, the formed communities are increasingly larger. In contrast, when the b_{th} value is high, the $b(l, x)$ value of each node has difficulty exceeding b_{th}. At this time, only densely connected nodes, that is, only nodes with sufficiently large $b(l, x)$ values, can form communities. Thus, the community size in this moment is small, and its number is relatively large. Therefore, nodes with different closeness values can be distinguished by changing the value of b_{th} and further realizing multiscale community detection.

To record the labels of each node, the array $lab_i = [l_i]$ is used to represent that node i belongs to label l_i. During every iteration of the label propagation process, the tuple $LAB=\{lab_i\}$ is updated to record the labels that each node receives. The overlapping community appears when two or more labels are propagated to the same node, that is, the corresponding lab_i has more than one element. Nodes of this kind form overlapping regions.

Suppose that at the beginning of an iteration, the label to be spread is l and the node to be updated is x ($x \in V$). The algorithm to update label l for node x is defined as follows:

(1) The neighbors of node x comprise node set $Neg(x)$.

(2) In $Neg(x)$, the nodes with label l form node set $N(l)$:

$$N(l) = \{u \in Neg(x) | l \in lab_u\} \tag{5}$$

where lab_u represents the label set to which node u belongs.

(3) In $Neg(x)$, the nodes with labels form node set $N(L_{negx})$:

$$N(L_{negx}) = \{u \in Neg(x) | lab_u \neq \emptyset\} \tag{6}$$

(4) Based on (4)-(6), calculate the belonging coefficient $b(l, x)$ of node x to label l:

$$b(l, x) = \frac{\sum_{u \in N(l)} PI(u, x)}{\sum_{v \in N(L_{negx})} PI(v, x)} \tag{7}$$

where $N(l)$ denotes the node set consisting of all the nodes with label l in $Neg(x)$, so the numerator $\sum_{u \in N(l)} PI(u, x)$ represents the sum of the influence of the nodes with label l in $Neg(x)$ on node x. The denominator $\sum_{v \in N(L_{negx})} PI(v, x)$ represents the sum of the influence of all kinds of labels in $Neg(x)$ on node x. In addition, the nodes with no label do not belong to $N(L_{negx})$, so this kind of node has no effect on node x during label propagation. From (7), it can be seen that different labels will influence node x with different strengths. The closer $\sum_{u \in N(l)} PI(u, x)$ is to $\sum_{v \in N(L_{negx})} PI(v, x)$, the higher the probability node x has of obtaining label l. Thus, the node sets with different labels in $Neg(x)$, namely, adjacent communities, will all have an effect on the label propagation results, which indicates that whether node x can acquire label l is the consequence of the dynamic confrontation between adjacent communities.

As a result, the community structure detected at different scales by our algorithm is different from the traditional dendrogram [15], [21], [28], [34], meaning that in some small areas, when the belonging coefficient threshold (b_{th}) increases, the community size may become large and the number may decrease. However, this novel phenomenon is consistent with the reality of social networks. For example, when a person decides to participate in one or more communities, the influence of each community on the person will be considered, including the closeness (alienation or friendliness), impact (positive or negative), environmental similarity and other realistic factors. Therefore, the decision may not always be the same, especially when the internal community structure changes. That is, the person may choose to join a community depending on whether their friends are members of the community. The mechanism of the dynamic confrontation between adjacent communities is illustrated in detail in Section V-C2 in combination with the experimental results.

(5) Judging the relationship between $b(l, x)$ and b_{th}.

If $b(l, x) > b_{th}$, node x can obtain label l, meaning that label l is appended to lab_x. Otherwise, label l cannot be propagated to node x.

Here, we propose a label update strategy that can overcome the randomness of original label propagation strategy and increase the efficiency by propagating multiple labels simultaneously. The belonging coefficient threshold b_{th} is

introduced to realize multiscale community detection. That is, we first calculate $b(l, x)$ to evaluate the probability for node x to obtain label l and then compare $b(l, x)$ and b_{th} to determine whether l can be propagated to x. Thus, adjusting b_{th} can change the size and quantity of the detected communities.

IV. ALGORITHM DESCRIPTION

According to the definitions and label propagation method described in Section III, we propose a new multiscale community detection algorithm using label propagation, LPAMCD. It improves the stability and accuracy of traditional label propagation algorithms in two ways. First, it finds the hub nodes, which are the potential community centers and act as the initial points of label propagation. Second, we think the updating node is influenced by its neighbors, and the influence strength is determined by the sum of Propagation Indicator (*PI*). In other words, a label will have a stronger capability to propagate to the updating node when it affects more neighboring nodes, and each neighbor has a larger value of *PI*.

After completing the iteration process of label propagation, the nodes are assigned multiple labels form overlapping parts. Thus, LPAMCD can detect overlapped community structures. By identifying hub nodes and updating labels based on (7), the accuracy, stability and efficiency can be observably improved by LPAMCD.

A. HUB NODE IDENTIFICATION

To detect the overlapped or nonoverlapped community structures, we try to find the potential community central nodes that are recorded as hub nodes by adopting the idea of seed set expansion [1]. We think that the nodes with a larger *NI* have a higher possibility of being potential community centers, and this kind of node is recorded as a hub. In our algorithm, the hub node is screened based on the local peak of *NI*, i.e., the *NI* value is no less than that of all its neighbors. In the rare case, when the NI values of adjacent nodes that are also local peaks are the same, we choose one of the nodes as the hub node. Taking Fig. 1 as an example, it can be seen from Table 2 that the *NI* values of node 5 and node 11 are local peaks. Therefore, the set of hub nodes for this network is {5, 11}.

The hub node set, recorded as *Source*, contains the community center nodes. At the beginning of the label propagation process, each hub is assigned a unique label and acts as the starting point for label spreading. To lower the community detection instability caused by label randomness in the original label propagation algorithm, we will propagate labels from the screened hub nodes or according to the sequence of nodes sorted in descending order of *NI*, ensuring the uniqueness of the label propagation result. Furthermore, different labels are propagated simultaneously from each hub, which improves the spreading efficiency.

B. LABEL PROPAGATION PROCESS

In the label propagation algorithm, nodes with the same label belong to the same community. As mentioned above, we use array lab_x=[l] to indicate that node x has label l. By extending the elements in lab_x, all the labels obtained by node x can be recorded. lab_x=∅ indicates that node x has not received any label and belongs to no community, while $|\ lab_x\ | \geq 2$ means that node x belongs to the overlapping part. In addition, the array c_i={(x, t)} denotes that node x joins community c_i at time t, and c_i can be appended when other nodes join community c_i. The tuple C={c_i} denotes the overall community structure detected in the network and is updated during the label propagation process. Note that the timestamps of label infection are recorded by t; therefore, the synchronization of label propagation in our algorithm is guaranteed.

1) LABEL PROPAGATION FROM HUB NODES

There are two phases in the label propagation process of LPAMCD. In the first phase, the hub nodes construct the set *Source*, and each hub is assigned a unique label, that is, lab_u=[s] ($u \in$ *Source*),where s is the label allocated to hub node u. Other nodes except the hub nodes are not assigned any label, so lab_x=∅ ($x \notin$ *Source*). Initially, taking each hub node as the starting point, the labels of all the hub nodes propagate simultaneously. Then, we regard the newly affected nodes as new starting points in the next iteration of label propagation. It should be noted that the starting points only affect their immediate neighbors during each iteration, and the detailed label propagation rule is as follows: During the t-*th*round of the propagation process, for node x, suppose the label set of its neighbor nodes is {$l_1, l_2, \ldots, l_m$}. Then, calculate the belonging coefficient $b(l_i, x)$ according to (5)-(7) and compare the result with the threshold b_{th}. As noted earlier, only when $b(l_i, x) > b_{th}$ can node x obtain label l_i and join community c_i at time t, meaning that c_i and lab_x are updated. After several such iterations, the labels propagate from hub nodes to their surroundings. The infection of a certain label will stop when $b(l, x)$ is not larger than b_{th} or when the updating node has receives the label. When the tuple C for the whole network is no longer changed or the iteration time t reaches the upper limit T, the label propagation iteration ends.

Specifically, if the label of one hub is propagated to other hubs, the communities developed by these hub nodes will be merged into one, meaning that the nodes in these communities will have the same label. We integrate this merging process into the label propagation algorithm, and the process is as follows: Before label propagation from hub node i, we check the element number in lab_i, that is, $|lab_i|$. Only when $|lab_i|$=1 can label propagation from node i proceed normally. If there are two or more elements in lab_i, it indicates that the corresponding community, which is c_i, should be merged into the others. That is, the original community c_i is a subset of another community to which

node i belongs. Therefore, the label propagation process starting from this hub node i is skipped by setting the corresponding community c_i to an empty community ($\emptyset$). Through experimental validations, this operation effectively avoids repeated detection for communities that need to be merged. Thus, the efficiency is greatly improved, and the detection time is reduced. The label propagation process from hub nodes is detailed in Algorithm 1.

Algorithm 1 Label propagation process from hub nodes

Input: Network $G = (V, E)$, Maximum iteration number T, The set of hub nodes *Source*, The set of *Source*labels $LAB = \{lab_i \mid lab_i = [l_i],\ i \in Source\}$, The set of community $C = \{c_i \mid c_i = \{(i, 0)\},\ i \in Source\}$
Output: Community set C and label set LAB
for all nodes $x \in V$ **do**
  for $y \in Neg(x)$ **do**
  $DI(x, y) \leftarrow$ Calculate *Degree Influence* according to (2)
  $Sim(x, y) \leftarrow$ Calculate *Similarity* according to (3)
  $PI(x, y) \leftarrow$ Calculate *Propagation Indicator* according to (4)
  end for
end for
$t=0$
while $t<T$ **do**
  $t=t+1$
  for *node* $i \in Source$ **do**
    if $|lab_i| = 1$ **then**
      if $\forall$ node i s.t. $(i, t\text{-}1) \in c_i$**then**
        for node $x \in Neg(i)$ **do**
          $N(l_i) = \{u \in Neg(x) \mid l_i \in lab_u\}$
          $N(L_{negu}) = \{u \in Neg(x) \mid lab_u \neq \emptyset\}$
          $b(l_i, x) \leftarrow$ Update the belong coefficient label of l_i propagating to node x according to (7)
          if $b(l_i, x) > b_{th}$ **then**
            Add l_i to lab_x
            Add $\{x, t\}$ to c_i
          end if
        end for
      end if
      else $c_i = \emptyset$ /*Skip the community detection for the hub node i which have labels propagated from other hubs */
      end if
end for
if C does not change **then**
    break
  end if
end while
Output community set C and label set LAB

Taking the sample network in Fig. 1 as an example, the hub nodes identified are {5, 11}, as described in Section IV-A. To distinguish the number of nodes and labels, we use n to represent the node name and l_n to denote the label name in this paper. Therefore, the labels assigned to the hub nodes are $lab_5 = [l_5]$ and $lab_{11} = [l_{11}]$, meaning that node 5 is assigned label l_5 and node 11 is assigned label l_{11}. Then, we calculate *Dinf*, *Sim* and *PI* between different node pairs of the sample network, as shown in Table 3. Note that the horizontal and vertical node IDs in Table 3 represent node x and node y in (2)-(4), respectively. Table 3 and Table 3 show the values of *DI* and *Sim* from node x to node y, and Table 3 shows the value of *PI* to evaluate the probability of the label propagating from node x to node y.

The label propagation process of the sample network in Fig. 1 is illustrated in Fig. 2. As described above, only hub nodes {5, 11} have labels, which are marked in blue and red in Fig. 2(a). Then, labels l_5 and l_{11} start to propagate from node 5 and node 11 to their immediate neighbors synchronously. For node 1, $Neg(1)$={2, 4, 5, 6, 8}, $N(l_5)$ ={5} and $N(L_{neg1})$ = {5}. Then, we calculate $b(l_5, 1)$ according to (7) and obtain $b(l_5, 1)$=1. In the same way, $b(l_5, 2)$= $b(l_5, 3)$= $b(l_5, 4)$=1 and $b(l_{11}, 6)$= $b(l_{11}, 7)$= $b(l_{11}, 8)$= $b(l_{11}, 9)$=1. Therefore, labels l_5 and l_{11} are propagated to their neighbors in the first iteration, as shown in Fig. 2(b). In the next iterations, the labels keep propagating from the newly updated nodes to their neighbors, which either have no label or have not obtained the spreading ones. Taking node 6 as an example, $Neg(6)$={1, 8, 9, 11}, $N(l_5)$ ={1} and $N(L_{neg6})$ = {1,8, 9, 11}. Therefore, $b(l_5, 6)$ is computed as follows:

$$b(l_5, 6) = \frac{PI(1, 6)}{PI(1, 6) + PI(8, 6) + PI(9, 6) + PI(11, 6)} = 0.227 \quad (8)$$

In the same way, we calculate the new label belonging coefficient for each node, which is shown in Fig. 2(c). The belonging coefficient of label l_5 propagating to node 6 or 8 is 0.227, and that of label l_{11} propagating to node 1 is 0.39, so whether the label can affect the nodes successfully is related to the belonging coefficient threshold, that is, b_{th}.

Here, with the following two b_{th} values, we discuss the different community detection processes in detail. Case one, when b_{th} =0.45, as shown in Fig. 2(d1), label l_5 and label l_{11} cannot affect node {6, 8} and node {1}, respectively, as their belong coefficients, which are 0.227, 0.227 and 0.39, are smaller than 0.45. Therefore, label propagation stops, and the network is partitioned into two communities: {1, 2, 3, 4, 5}, which are marked in blue, and {6, 7, 8, 9, 10, 11}, which are marked in yellow.

Case two, when b_{th} =0.3, as shown in Fig. 2(d2), label l_{11} can be propagated to node 1, as its belonging coefficient 0.39 is larger than 0.3, which is marked by (l_{11}, 0.39, 2). Then, we calculate the belonging coefficients of label l_{11} on node {2, 4, 5}, which are 0.31, 0.31 and 0.75, respectively. The three nodes are all affected by label l_{11}, and the calculation results are all larger than 0.3, as shown in Fig. 2(d2). As hub node 5 obtains label l_{11}, community {1, 2, 3, 4, 5} will be merged into community {6, 7, 8, 9, 10, 11}. Thus, the whole network forms one community. Based on the above analysis, it is demonstrated that adjusting the value of b_{th} can change the community structure and realize multiscale community detection.

TABLE 3. (a) Degree Influence (Dinf) matrix of the sample network. (b) Jaccard similarity (Sim) matrix of the sample network. (c) Propagation indicator (PI) matrix of the sample network.

Node x / Node y	1	2	3	4	5	6	7	8	9	10	11
1	0	0.375	0	0.375	0.44	0.44	0	0.44	0	0	0
2	0.625	0	0.5	0	0.57	0	0	0	0	0	0
3	0	0.5	0	0.5	0.57	0	0	0	0	0	0
4	0.625	0	0.5	0	0.57	0	0	0	0	0	0
5	0.56	0.43	0.428	0.428	0	0	0	0	0	0	0
6	0.56	0	0	0	0	0	0	0.5	0.5	0	0.5
7	0	0	0	0	0	0	0	0.5	0.5	0.33	0.5
8	0.56	0	0	0	0	0.5	0.5	0	0	0	0.5
9	0	0	0	0	0	0.5	0.5	0	0	0.33	0.5
10	0	0	0	0	0	0	0.67	0	0.67	0	0
11	0	0	0	0	0	0.5	0.5	0.5	0.5	0	0

(a)

Node x / Node y	1	2	3	4	5	6	7	8	9	10	11
1	0	0.23	0	0.143	0.286	0.125	0	0.125	0	0	0
2	0.14	0	0.2	0	0.4	0	0	0	0	0	0
3	0	0.35	0	0.2	0.4	0	0	0	0	0	0
4	0.14	0	0.2	0	0.4	0	0	0	0	0	0
5	0.286	0.41	0.4	0.4	0	0	0	0	0	0	0
6	0.125	0	0	0	0	0	0	0.33	0.14	0	0.33
7	0	0	0	0	0	0	0	0.14	0.33	0.2	0.33
8	0.125	0	0	0	0	0.33	0.143	0	0	0	0.33
9	0	0	0	0	0	0.143	0.33	0	0	0.2	0.33
10	0	0	0	0	0	0	0.2	0	0.2	0	0
11	0	0	0	0	0	0	0.33	0.33	0.33	0	0

(b)

Node x / Node y	1	2	3	4	5	6	7	8	9	10	11
1	0	0.26	0	0.26	0.365	0.28	0	0.28	0	0	0
2	0.38	0	0.35	0	0.486	0	0	0	0	0	0
3	0	0.35	0	0.35	0.486	0	0	0	0	0	0
4	0.38	0	0.35	0	0.486	0	0	0	0	0	0
5	0.42	0.41	0.41	0.41	0	0	0	0	0	0	0
6	0.34	0	0	0	0	0	0	0.42	0.32	0	0.42
7	0	0	0	0	0	0	0	0.32	0.42	0.27	0.42
8	0.34	0	0	0	0	0.42	0.32	0	0	0	0.42
9	0	0	0	0	0	0.32	0.42	0	0	0.27	0.42
10	0	0	0	0	0	0	0.43	0	0.43	0	0
11	0	0	0	0	0	0.42	0.42	0.42	0.42	0	0

(c)

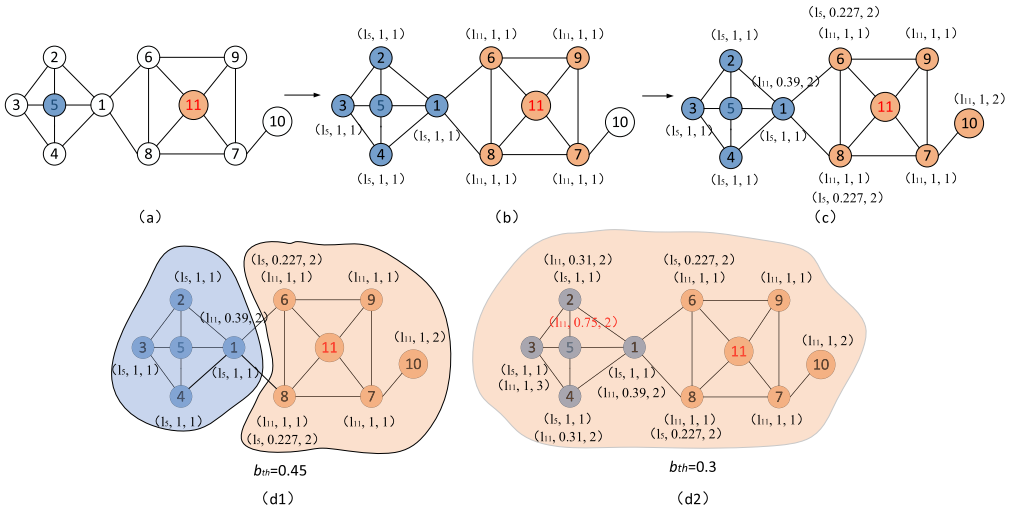


FIGURE 2. Label propagation process of LPAMCD on the sample network. The marker (l_a, b, c) next to each node represents (label, belonging coefficient, time). For example, (l_5, 1, 1) beside node 2 means that label l_5 is propagated to node 2 with b(l_5, 2)=1 at time 1.

2) LABEL PROPAGATION AMONG PENDING NODES

In some cases, especially in large networks, there are still some nodes that have no label after the process of label propagation from hub nodes in the first phase. These kinds of nodes are called pending nodes in our paper, and the node set is denoted by P. In the second label propagation phase, we focus on partitioning the pending nodes, and the propagation loops through the following steps:

1. Sort the pending nodes in descending order of NI to obtain the array $pQueue$. The node with the highest NI value in P ($pQueue[0]$) is treated as the center of a new community, so a new label is assigned to $pQueue[0]$, and the node is the starting point of label propagation. Moreover, the new element $\{(pQueue[0], t)\}$ is appended to the end of $C$to update the community structure.

2. As in phase one, calculate $b(l, x)$ of each updating node and compare the results with b_{th}. The label l can be propagated to node x only when $b(l, x) > b_{th,}$, and then the array c_i and lab_x are updated. Iterate the process several times until community detection from node $pQueue[0]$ is completed.

3. Similar to the merger cooperation in phase one, if the label of node a_1 ($a_1 \in P$) propagates to node a_2 ($a_2 \in P$), the communities corresponding to nodes a_1 and a_2 can be merged. We remove all the nodes like node a_2 from set P. In this way, the community detections for these nodes are skipped. This operation can greatly improve the efficiency of the detection process and decrease the complexity of the algorithm.

With the above three-step iteration process, the community partition for pending nodes, which are the remaining unsigned nodes, is realized until the iterations reach T or the set $pQueue$ becomes $\emptyset$.

It should be noted that in the first phase of the algorithm, the community detection for some nodes is skipped with the corresponding $c_i=\emptyset$ ($c_i \in C$). Therefore, the final accurate community structure can be acquired by removing the empty collections in C after the two-phase label propagation process is completed. The label propagation process among pending nodes is detailed in Algorithm 2.

Algorithm 2 Label propagation process among pending nodes

Input: Network $G = (V, E)$, Maximum iteration number T community set C and label set LAB
Output: Community structure C and label set LAB
$P = \{x \mid lab_x = \emptyset\}$ ← identify the set of pending nodes
$pQueue$ ← sort P in descending order according to NI
while $pQueue \neq \emptyset$ **do**
  Assign a label to $pQueue[0]$
  Append C to include $c_p = \{pQueue[0], t\}$
  $t=t+1$
  while $t < T$ **do**
    for ∀ node is.t. $(i, t\text{-}1) \in c_p$ **do**
      for node $x \in Neg(i)$**do**
        $N(l_i)=\{u \in Neg(x) \mid l_i \in lab_u\}$
        $N(L_{neg})=\{u \in Neg(x) \mid lab_u \neq \emptyset\}$
        $b(l_i, x)$ ← update belonging coefficient of label l_i propagating to node x according to (7)
        if $b(l_i, x) > b_{th}$ **then**
          Add l_i to lab_x
          Add $\{x, t\}$ to c_p
        end if
      end for
end for
if C does not change **then**
break
end if
end while
Remove nodes with labels from $pQueue$
end while
Remove $\emptyset$ from C
Output community structure C and label set LAB

C. COMPUTATIONAL COMPLEXITY

Consider a network with V nodes and E edges. In the first step of LPAMCD, the time complexity includes calculating NI, $Dinf$ and PI. The time complexity for computing the degree and clustering coefficient is $O(V)$ and $O(V^2)$,respectively, so the time complexity for NI is $O(V^2)$. According to (2), $Dinf$ is the ratio of the degree, so the time complexity for $Dinf$ is $O(V)$. The time complexity for the Jaccard Index is $O(V^2k)$, where k is the average degree of network nodes. According to (4), PI is linear to $Jaccard$ and $Dinf$, so the time complexity of PI is $O(V^2k+V) \simeq O(V^2k)$.

At step 2, hub nodes are identified based on the local peak of NI. This procedure is realized by comparing the NI of each node to all its neighbors, which incurs a time cost $O(E)$.

Step 3 is to perform the label propagation process, including labels propagating from hub nodes and pending nodes. Regardless of the value of b_{th}, after the last iteration of label propagation, every node is visited, and each visit accesses the neighbors of all the nodes. Intuitively, the propagation path from hub nodes or pending nodes is one-way. Therefore, the time complexity for this process is $O(V+E)$.

Thus, the overall time complexity of LPAMCD is $O(V^2+V+V^2k+E+(V+E))$. Commonly, $k \ll V$in most complex networks, so the final time complexity is $O(V^2+E)$. Compared with traditional LPA ($O(E)$), our algorithm has a certain increase in computational complexity. However, LPAMCD can realize full-resolution community detection with higher stability and efficiency, and it can eliminate randomness and accelerate the convergence rate.

V. EXPERIMENTS

In this section, we use three real-world networks to evaluate the accuracy, efficiency and stability of the algorithm. First, the qualitative analysis of LPAMCD in a small real network shows the effectiveness of multiscale community

detection. Second, to test the accuracy and efficiency of our algorithm, a quantitative analysis with three real-world networks is provided to illustrate the community structure variation with b_{th} and demonstrate the high accuracy and stability of LPAMCD. Specifically, the new finding about the dynamic confrontation between adjacent communities is also discussed in detail.

All the experiments are programmed in Python, simulated by PyCharm and conducted on a computer with a 2.4 GHz Intel Core i5 CPU and 16 GB RAM. The community structures of the networks are drawn in Gephi.

A. DATASETS

We test LPAMCD with three real-world networks, including the karate club network, dolphin network and Facebook network. To evaluate the quality of the community structure, the most popular metric is modularity, which was proposed by Newman and is based on the assumption that the links inside communities are denser than those outside the communities [35]. Although modularity is not a reasonable characterization for all communities of all real-world networks, it is still a widely accepted metric to evaluate the performance of community detection algorithms [36]. However, this metric is mainly focused on nonoverlapping communities. As the detected community structure in this paper may overlap, we choose *EQ*, proposed by Shen, to evaluate the community partition quality [21]. *EQ* is a quality function extension of modularity [35] to evaluate the goodness of overlapped and nonoverlapped community decomposition. It reduces to the traditional modularity (Q) [35] when the community has no overlap. The profiles of each network are shown in Table 4.

TABLE 4. Profiles of real networks.

Networks	Nodes	Edges	Average degree	Clustering coefficient	Assortative coefficient
Karate club	34	78	4.59	0.57	-0.48
Dolphins	62	159	5.13	0.26	-0.04
Facebook	4039	88234	43.69	0.6	0.06

B. QUALITATIVE ANALYSIS

The main merit of our algorithm is that it has the capability to detect communities on any scale continuously and stably. This is in light of the tunable parameter b_{th}. Therefore, b_{th} allows users to obtain community structures of any resolution based on their actual needs. Due to different users having different needs, there is no "optimal" choice for b_{th} theoretically. To show the characteristics of community structures at multiple scales, we conduct experiments on the karate club network and dolphin network with ground truths, and the values of *EQ* and *NC*(number of communities) for different b_{th} values are listed.

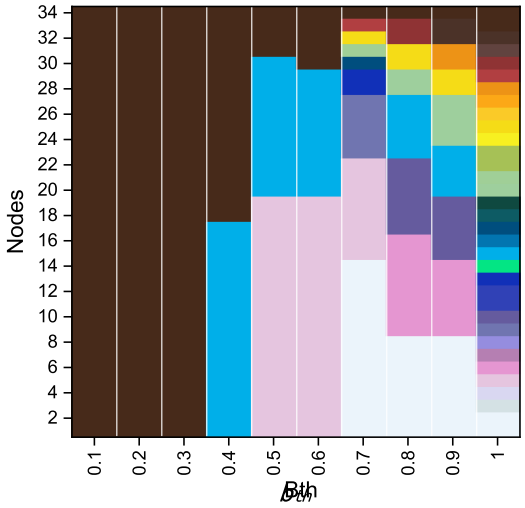


FIGURE 3. Community structures across different detection scales. We seek to reveal the changes in community structures in this synthetic graph. The belonging coefficient b_{th} varies between 0.1 and 1 with a step size of 0.1, and the communities are detected independently for each b_{th} value. Different colors in the graph represent different communities, and the area of the color block denotes the scale of each community.

1) KARATE CLUB NETWORK

The traditional karate club network is a social network consisting of friendships between 34 members from a karate club at a US university. Due to a dispute between the president and the instructor in the club, the members are split into several parts. The community structures across different detection scales are displayed in Fig. 3. The size and amount of the color blocks in Fig. 3 illustrate the scale and number of communities, respectively. It can be observed that at different b_{th} values, the community structures detected vary frequently in the community size and number. With an increase in b_{th}, there is a general trend that the number of communities increases and the size of each community decreases.

To verify the effectiveness of the algorithm and analyze the variation in the community structure, we present four typical community detection results in Fig. 4, where the values of b_{th} are 0.4, 0.51, 0.68 and 0.9. In addition, the community partition quality (*EQ*) and the community number (*NC*) of each structure are annotated. Referring to (1), the hub nodes found in the karate club are nodes 17, 4, 9 and 30, which are marked yellow in Fig. 4.

With the increase in b_{th}, the number of communities in Fig. 4(a)-(d) changes from 2, 3, 6 to 9, showing a trend of decreasing the size of each community and increasing the number of communities. When b_{th} =0.4, there are two communities, and nodes 10 and 20 overlap. When b_{th} =0.51, the left community in Fig. 4(a) splits into two subgroups: {5, 6, 7, 11, 17} and {1, 12, 2, 18, 20, 22, 4, 8, 13, 14}. When b_{th} increases to 0.68 and 0.9, subgroup {1, 12, 2, 18, 20, 22, 4, 8, 13, 14} continues to divide, while subgroup {5, 6, 7, 11, 17} remains unchanged. These results reveal that nodes {5, 6, 7, 11, 17} have higher tightness among themselves. Even though b_{th} is as high as 0.9, community {5, 6, 7,

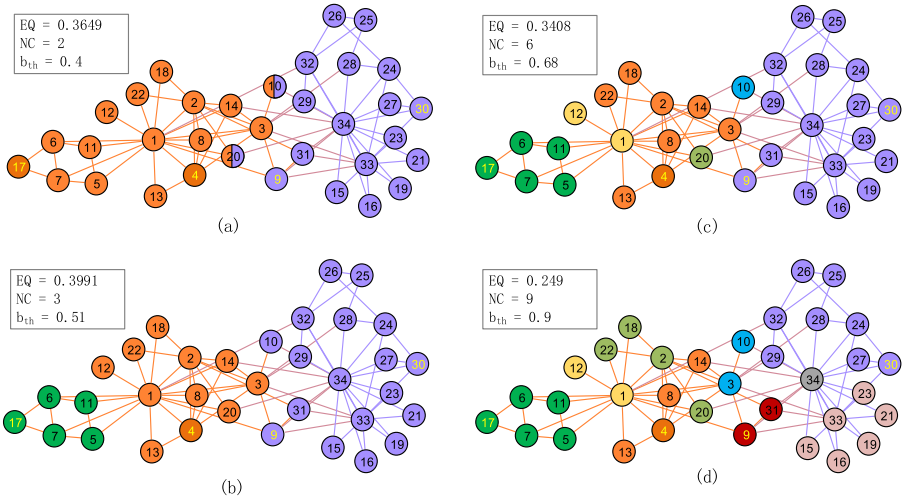


FIGURE 4. Community structures of the karate club network at multiple scales. Different colors of nodes represent different communities. With the increase in b_{th}, there is a general trend that the number of communities increases and the size of each community decreases.

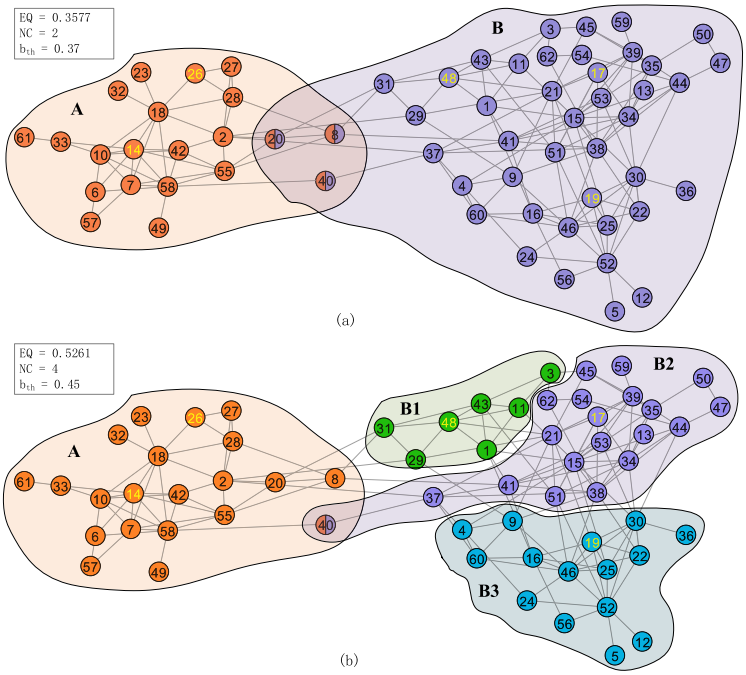


FIGURE 5. Community structures of the dolphin network for b_{th} =0.37 and b_{th} =0.45. With the increase in b_{th}, community A is unchanged, while community B splits into three subcommunities, B1, B2 and B3, indicating that the increase in b_{th} makes the communities partitioned with higher compactness.

11, 17} is still maintained. Similarly, subgroups {1,12}, {2, 18, 20, 22} and {4, 8, 13, 14} are also very tight clusters. In conclusion, when b_{th} is low, the requirement for closeness between nodes inside a community is weak. Thus, as shown in Fig. 4(a), communities with large scales and small numbers are obtained. In contrast, when b_{th} is high, only the nodes connected closely enough can form communities. As shown in Fig. 4(d), the scale of each community decreases, and the number of communities increases. Therefore, by adjusting b_{th}, communities with different levels of compactness can be screened out, and then multiscale community detection can be realized.

2) DOLPHIN NETWORK

The dolphin network is a social network based on the frequent associations among 62 dolphins living off Doubtful Sound

in New Zealand. The nodes are dolphins, and the edges represent relationships between dolphins that associate more frequently. The dolphin network has obviously modularized characteristics.

Fig. 5 illustrates two typical community structures under different detection scales, where the values of b_{th} are 0.37 and 0.45, showing the efficiency of our algorithm and the variation in the community structure. The hub nodes found by LPAMCD in the dolphin network are nodes 14, 17, 19, 26 and 48, which are marked yellow. When b_{th} increases from 0.37 to 0.45, the number of communities change from 2 to 4. This illustrates that high b_{th} values improve the resolution of community recognition. That is, the nodes inside a community have closer connections with each other, resulting in the number of communities becoming large and the scale of each community decreasing. As shown in Fig. 5, when b_{th} =0.37, two large communities, named A and B, are detected, and nodes 20, 8 and 40 overlap. This community structure is basically consistent with the traditional community detection results [1], [6]. When b_{th} =0.45, the scale of community A is unchanged, while community B splits into three subcommunities, B1, B2 and B3, indicating that an increase in b_{th} makes the communities partitioned with higher compactness.

C. QUANTITATIVE ANALYSIS

1) VARIATION IN EVALUATION PARAMETERS WITH B_{TH}

To understand the influence of the b_{th} value on community structure detection more comprehensively, we quantitatively discuss the variations in the community partition quality (EQ) and community number (NC) with b_{th} based on three real-world networks (karate club network, dolphin network and Facebook network), which further verify the effectiveness and accuracy of this algorithm.

a: KARATE CLUB NETWORK

As shown in Fig. 4(b), when b_{th} =0.51, EQ reaches the maximum value of 0.3991, meaning that from the view of the partition quality, the community structure at this time is the most significant. This proves that our algorithm can find the optimal decomposition at certain detection scales. When the value of b_{th} is 0.4, 0.68 and 0.9, as shown in Fig. 4(a), (c) and (d), EQ is 0.3649, 0.3408 and 0.249, respectively. The first two values of EQ are above 0.3, indicating that the algorithm can correctly recognize different community structures. Although in the last case, the value 0.249 is relatively low, the community structure in Fig. 4(d) still has practical significance, as the community partition target at this time is to screen out nodes with high compactness rather than find the best community partition structure. This helps the model to obtain communities with different closeness levels according to different actual needs.

We present the curves of EQ and NC with b_{th} in Fig. 6, where the range of b_{th} is [1, 0] and the step size is 0.01.

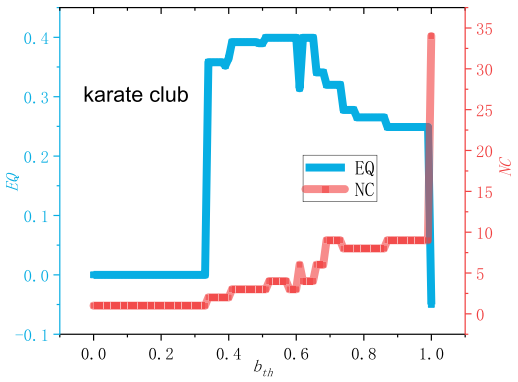


FIGURE 6. Variation in EQ and NC against b_{th} in the karate club network.

As shown in Fig. 6, when b_{th} ranges from 0 to 0.33, NC is 1 and EQ is 0, meaning that the given threshold b_{th} is relatively low at this time and the compactness between any nodes in the network can exceed it. Therefore, all nodes form one large community. With the increase in b_{th}, EQ shows an upward trend. When b_{th} ranges from 0.34 to 0.73, the corresponding values of EQ are all above 0.3 with the maximum value up to 0.3991, and the variation range of NC is 2∼9. In this interval, although the community structure changes with b_{th}, the community clustering quality is high. It shows that our algorithm can accurately identify various community structures under different detection scales. When b_{th} is greater than 0.65, EQ decreases with increasing b_{th}, indicating that the given b_{th} value is relatively high and that the compactness between some nodes cannot exceed it. The communities detected at this moment are small in size and large in number. In this way, the node sets with high closeness values are identified.

When b_{th} =1, the compactness between all pairs of nodes cannot exceed it, as the belonging coefficient of nodes must be less than 1. Therefore, each node forms a community; that is, the 34 nodes in the karate network constitute 34 communities. There is no community partition for the network, and the corresponding EQ is negative. However, the detection results at this time still have practical significance because this special case represents the lowest resolution of community detection.

In conclusion, the algorithm proposed in this paper can not only identify various community structures with different compactness at different scales but also realize full resolution detection from all nodes forming a whole community to each node forming an independent community.

b: DOLPHIN NETWORK

In the two community structures shown in Fig. 5, the values of EQ are 0.3577 and 0.5261, demonstrating that the two communities have high partition quality and that our algorithm is efficient and accurate. Fig. 7 illustrates the changes in EQ and NC with b_{th}. In the range of [0, 0.324] for b_{th}, NC is 1, and the corresponding EQ is 0 since the given b_{th}

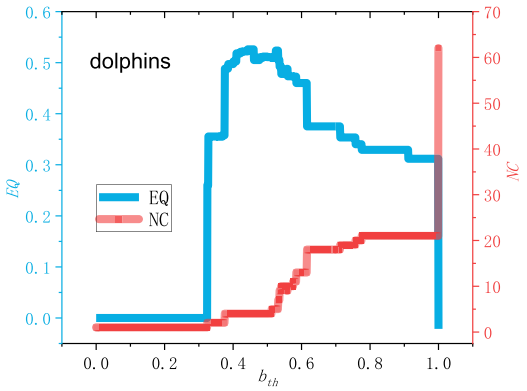


FIGURE 7. Variation in EQ and NC against b_{th} in the dolphin network.

at this time has no requirement for the compactness between nodes, leading to all nodes forming one community. As b_{th} increases, EQ shows an overall upward trend. In Fig. 7, when the b_{th} value is within the large range of [0.328, 1), the values of EQ are all above 0.3, indicating that at different scales, i.e., with different values of b_{th}, the clustering qualities of various community structures detected by LPAMCD are all relatively high. Especially when the range of b_{th} is [0.443, 0.46], EQ reaches the maximum value of 0.5261, and the community structure is shown in Fig. 5(b). As each node constitutes a community when b_{th} is 1, there are 62 communities in total, and the corresponding EQ value is negative. Again, it is verified that this algorithm can achieve full resolution (from a whole community to a single node community) community structure detection while ensuring high community partition quality.

c: FACEBOOK NETWORK

The Facebook network is a large social network whose data were collected from survey participants through the Facebook app. The nodes represent people, and edges represent interactions between people. After applying LPAMCD, we found 13 hub nodes, which are potential community centers. As the Facebook network is a large network with 4039 nodes and 88234 edges, it is unrealistic to show the node distribution of each community. Therefore, we use the community quality criterion (EQ) and number of communities (NC) to illustrate the effectiveness and efficiency of LPAMCD, as shown in Fig. 8(a).

When the value of b_{th} ranges from 0 to 0.078, the given belonging coefficient threshold is low, resulting in NC being 1 and EQ being 0. As b_{th} increases, NC increases, and the corresponding EQ presents an overall upward trend. For a large range [0.108, 1) of b_{th}, the values of EQ are all above 0.3, indicating that the community structures detected by LPAMCD are reasonable and the clustering qualities are high. The EQ value reaches a maximum of 0.7449 when b_{th} is 0.53. When b_{th} =1, as shown in Fig. 8(a), 4093 communities constructed by a single node appear, and the EQ value is negative. Fig. 8(b) shows in more detail the change

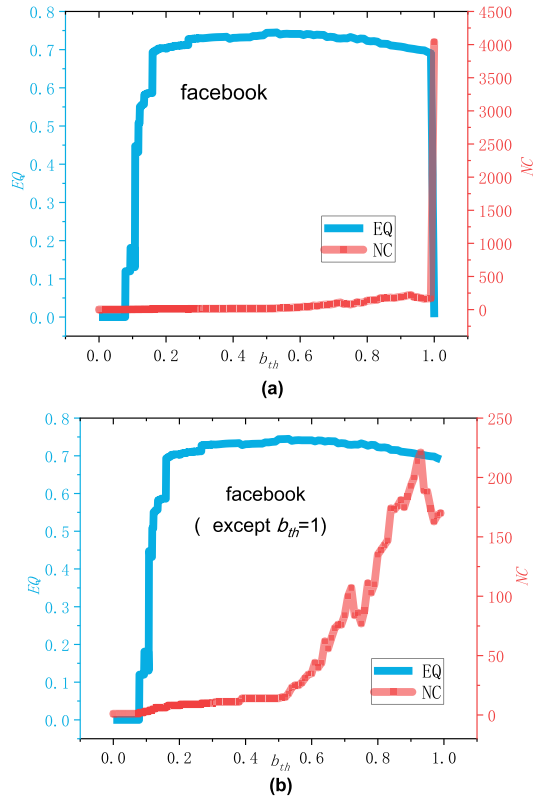


FIGURE 8. Variation in EQ and NC against b_{th} in the Facebook network.

in NC in the range [0,1) of b_{th}. The experiment on the Facebook network demonstrates that our algorithm has high performance in multiscale community detection.

From the experiments, it can be seen that b_{th} can realize full-resolution community detection of networks, and the value is determined by practical demand. However, in the case of detecting the best community partition structure, the optimal b_{th} value corresponds to the maximum EQ. Figures 6-8 show that the b_{th} values at the maximum EQ are different, which reflects the unique properties of the different networks. Therefore, the determination of the optimal b_{th} requires EQ-b_{th} analysis, and there is no uniform value. The recommended value based on the existing analysis results is b_{th} =0.4∼0.6.

2) DYNAMIC CONFRONTATION BETWEEN ADJACENT COMMUNITIES

Unexpectedly, Fig. 6 and Fig. 8(b) show that the number of communities (NC) does not strictly change linearly with the belonging coefficient threshold (b_{th}); that is, in some small areas, NC decreases slightly with the increase in b_{th}. Taking the karate club network as an example, as shown in Fig. 9, when b_{th} increases from [0.69, 0.73] to [0.74, 0.86], NC decreases from 9 to 8. It can be seen that with an increase in b_{th}, the subcommunity A-{24, 25, 26, 27, 28, 29, 30, 32, 15, 16, 19, 21, 23, 33, 34} with label A on the right of Fig. 9(a) is split into three smaller subcommunities in Fig. 9(b):

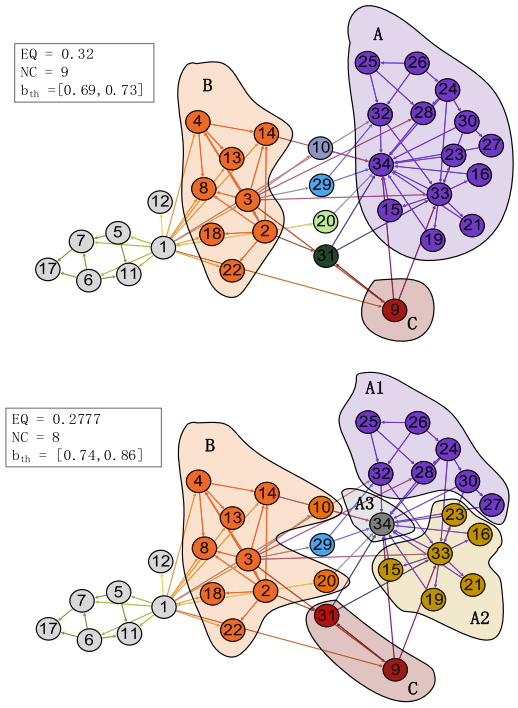


FIGURE 9. Community structures with $b_{th} \in [0.69, 0.73]$ and $b_{th} \in [0.74, 0.86]$. With the increase in b_{th}, the large community A splits into three small communities, namely, A1, A2 and A3. Therefore, the influence of community A is reduced. Then, the single nodes {10} and {20} are included in community B, and the single node {31} is included in community C. It can be seen that the total number of communities decreases from 9 to 8, although b_{th} increases. This small-scale nonlinear phenomenon is expressed by the dynamic confrontation between adjacent communities in absorbing boundary nodes, which is a novel finding in this paper.

A1-{24, 25, 26, 27, 28, 29, 30, 32}, A2-{15, 16, 19, 21, 23, 33} and A3-{34}. It is obvious that the nodes in subcommunities A1 and A2 have higher compactness, and the sizes of the communities decrease while the number increases. However, the communities {20}, {10}, {9}, and {31} constituted by single nodes in Fig. 9(a) are merged into the communities B-{3, 4, 8, 13, 14, 2, 18, 22, 20, 10} and C-{9, 31} in Fig. 9(b), resulting in the expansion of community scales and the reduction in the community number. This is very different from the traditional structure of a dendrogram [15], [21], [28], [34]. As described in Section III-B, this small-scale nonlinear phenomenon is caused by the dynamic confrontation between adjacent communities in absorbing boundary nodes, which is also a novel finding in this paper. In some networks, especially social networks, this phenomenon is consistent with real life. We will discuss it from two aspects: algorithm principle and social laws.

From the perspective of the algorithm principle, referring to (7), the value of $b(l, x)$ is determined by both the label types of neighboring nodes and the Propagation Indicator (PI) from every neighboring node to x. In the neighbors of node x, there are multiple node sets with different labels, and each node set has a certain influence on the propagation of label l to node x. Only the labels whose influence exceeds the threshold b_{th} can be propagated to node x successfully, which leads to the phenomenon of dynamic confrontation between adjacent communities with different labels. In Fig. 9(a), node 9 is a hub node with label C. When labels A, B, and C are propagated to nodes 20, 10 and 31, communities A and B produce two strong influences that are evenly matched. Therefore, nodes 20, 10, 9 and 31 form four single node communities, rather than belonging to community A or B. However, in Fig. 9(b), with the increase in b_{th}, community A is split into three smaller communities A1, A2 and A3. Obviously, the influence of labels A1, A2 and A3 on nodes 20, 10, 9 and 31 is greatly reduced, while the influence of label B on nodes 20 and 10 and that of label C on node 31 are relatively enhanced; therefore, nodes 20 and 10 are included by community B, and node 31 is included by community C. This explains that at some small scales, with the increase in the belonging coefficient threshold (b_{th}), the number of communities (NC) decreases slightly.

The phenomenon of dynamic confrontation between adjacent communities can also be interpreted from the perspective of real life. Social networks are different from biochemical networks, transport networks, etc. The connection relationship between each node of the latter is usually fixed, and the hierarchy is strict, so networks of this kind have dendrogram structures [15]. However, in social networks, the interpersonal relationship is delicate, and the decision of a person about whether to join a certain club is flexible and changes with variations in the surrounding environment. For example, if there are several communities around a person, before deciding to join one or more, the person will consider the influence from each community, including various practical factors such as the comparison and competition of two strengths, that is, which community can provide more benefits. Therefore, this paper quantitatively explains that communities in social networks do not follow the strict dendrogram structure and that the hierarchy of community members is variable.

3) ACCURACY AND STABILITY

We compare the proposed algorithm LPAMCD with SLPA and COPRA, which are found to have better performance for overlapping community detection [37], and SFLPA [1], which stably detects overlapping communities in large social networks.

For accuracy, to obtain reliable experimental results, we apply the four algorithms LPAMCD, SLPA, COPRA and SFLPA to three real-world networks: the karate club network, dolphin network and Facebook network. The quality of the community detection results is evaluated by EQ, as shown in Fig. 10. Considering that the detection results of SLPA and COPRA are unstable, we run the experiments 50 times and choose the highest value of EQ. With increasing b_{th}, the EQ curves of LPAMCD under different networks have similar features. That is, the initial value is 0, and then it gradually increases until it reaches the peak value. After this point,

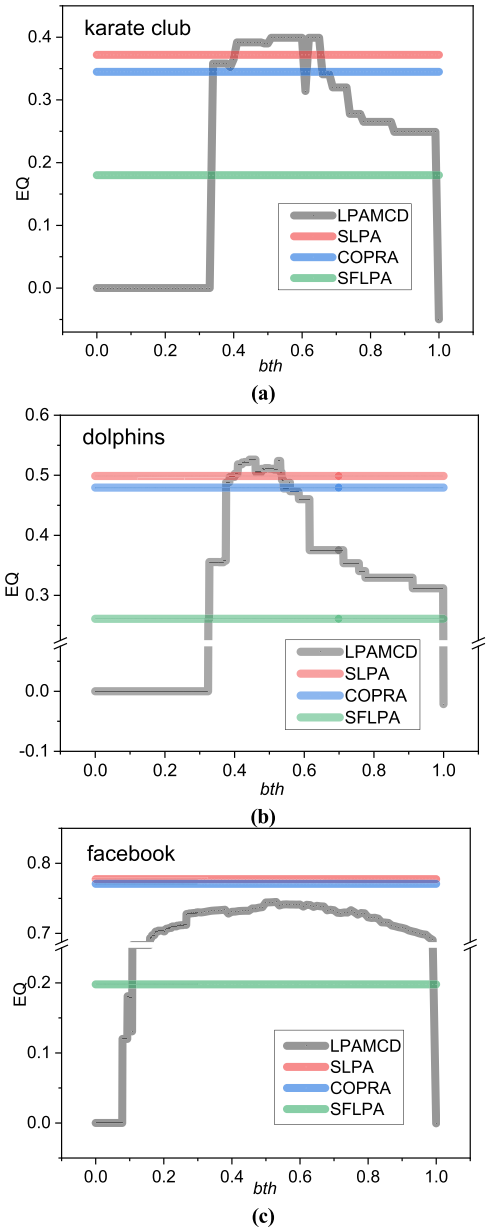


FIGURE 10. Quality of the community detection results evaluated by EQ. (a) karate club network; (b) dolphin network; (c) Facebook network.

the *EQ* value decreases to negative, as each community is composed of a single node when b_{th} is 1.

In Fig. 10, the *EQ* values of LPAMCD vary with b_{th} in a parabolic shape. Near the top of the parabola, the corresponding *EQ* values of LPAMCD are the highest compared with the other three algorithms, indicating that the community quality of LPAMCD is the best. Although the *EQ* value decreases at both ends of the parabola, the slopes on both sides differ greatly. On the left side of the parabola, as b_{th} increases, *EQ* suddenly increases at a certain value of b_{th}. On the right side, nevertheless, *EQ* decreases gently. This means that the influence of b_{th} on the community detection results is nonlinear. Within a range on both sides of the parabolic peak, the corresponding *EQ* values of LPAMCD are between those of SLPA and COPRA. Furthermore, in a large range of b_{th}, the *EQ* values of LPAMCD are larger than those of SFLPA. These results demonstrate that the community structure obtained by LPAMCD has high reliability and effectiveness, followed by the SLPA and COPRA algorithms, and the effectiveness of the SFLPA algorithms is comparatively low. When b_{th} becomes too small or too large, the *EQ* values of LPAMCD approach zero. However, this does not mean that the community structure has no practical significance. As described in Section V-C1.A, the target of community detection is to partition communities with different closeness values, such as searching for nodes with high compactness, rather than looking for the best community structure. This is exactly the innovation of this article, so the community structures within this range are in line with practical needs.

TABLE 5. Standard deviation (SD) and Range of different algorithms for four real networks.

Algorithm	Karate club		Dolphin		Facebook	
	SD	Range	SD	Range	SD	Range
SLPA	***0.144***	***0.378***	***0.048***	0.238	0.018	0.047
COPRA	0.129	0.372	0.033	***0.252***	0.018	***0.078***
LPAMCD	0	0	0	0	0	0
SFLPA	0	0	0	0	0	0

The LPAMCD overcomes the instability problem of the traditional LPA algorithm; that is, the label propagation results of each round of the latter are different. In the LPAMCD algorithm, we first select the hub nodes (community centers) and then start to synchronously propagate different labels from each hub node. During the propagation process, the quantitative relation between *b(l, x)*of each node and b_{th} is judged by referring to (5)-(7). After multiple iterations, the label propagation range is determined, and then the community structure is finally obtained. In this process, if the threshold b_{th} is fixed, the starting points and label propagation paths are definite. Therefore, the community detection result of label propagation is unique, overcoming the problem of instability.

For other algorithms, including SLPA and COPRA, instability problems exist. In Fig. 10, the *EQ* values of SLPA and COPRA are the maximum values of the 50 experiments. And the SD (standard deviation) and the range (the difference between minimum and maximum) of *EQ* in 50 experiments are shown in Table 5. These data shows that the instability of SLPA and COPAR is the strongest. Since both the SD and range of LPAMCD are 0, this algorithm overcomes the instability problem of traditional LPA, as discussed above. That is, for the same network, when b_{th} is fixed, the community structure detection result is unique. SFLPA adopts the LPA algorithm based on the idea of seed set expansion, so its detection result is also fixed. Nevertheless, combined with Fig. 10, the accuracy of this algorithm is

obviously lower than that of the other three algorithms. Therefore, the LPAMCD algorithm proposed in this paper has both high accuracy and stability.

VI. CONCLUSION

A novel multiscale community detection algorithm (LPAMCD) is proposed in this paper based on label propagation. With the variation in b_{th}, LPAMCD can detect multiscale overlapping community structures according to different compactness values between nodes at full resolution. In addition, it is found that there exists dynamic confrontation between adjacent communities, which indicates that the community structure is not a strict dendrogram, especially in social networks. By identifying hub nodes based on Node Importance as the starting points of label propagation and adopting a novel label update method based on Propagation Indicator and belonging coefficient, LPAMCD significantly increases the accuracy and stability for community detection at multiple scales.

However, LPAMCD is currently mainly applied in social networks. We will try to extend it to other kinds of networks, such as signed networks and heterogeneous networks. In addition, we currently use EQ to evaluate the quality of the community structure. Next, we will try to explore other metrics for overlapping community evaluation, such as a metric based on link prediction [36], to verify the effectiveness of our algorithm more comprehensively.

REFERENCES

[1] T. Li and P. Zhang, "Self-falsifiable hierarchical detection of overlapping communities on social networks," *New J. Phys.*, vol. 22, no. 3, Mar. 2020, Art. no. 033014.

[2] Y. Zhang, T. Lyu, and Y. Zhang, "Hierarchical community-level information diffusion modeling in social networks," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2017, pp. 753–762.

[3] A. Ashourvan, Q. K. Telesford, T. Verstynen, J. M. Vettel, and D. S. Bassett, "Multi-scale detection of hierarchical community architecture in structural and functional brain networks," *PLoS ONE*, vol. 14, no. 5, May 2019, Art. no. e0215520.

[4] W. Zhao, J. Luo, T. Fan, Y. Ren, and Y. Xia, "Analyzing and visualizing scientific research collaboration network with core node evaluation and community detection based on network embedding," *Pattern Recognit. Lett.*, vol. 144, pp. 54–60, Apr. 2021.

[5] S. Fortunato, C. T. Bergstrom, K. Boerner, J. A. Evans, D. Helbing, S. Milojevic, A. M. Petersen, F. Radicchi, R. Sinatra, and B. Uzzi, "Science of science," *Science*, vol. 359, no. 6379, p. 1007, 2018.

[6] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New J. Phys.*, vol. 11, no. 3, Mar. 2009, Art. no. 033015.

[7] K. Berahmand, A. Bouyer, and M. Vasighi, "Community detection in complex networks by detecting and expanding core nodes through extended local similarity of nodes," *IEEE Trans. Computat. Social Syst.*, vol. 5, no. 4, pp. 1021–1033, Dec. 2018.

[8] K. Berahmand, M. Mohammadi, F. Saberi-Movahed, Y. Li, and Y. Xu, "Graph regularized nonnegative matrix factorization for community detection in attributed networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 1, pp. 372–385, Jan. 2023.

[9] L. Zhou, K. Lü, and W. Liu, "An approach for community detection in social networks based on cooperative games theory," *Expert Syst.*, vol. 33, no. 2, pp. 176–188, Apr. 2016.

[10] Y. Liu, R. Gu, Z. Yang, and Y. Ji, "Hierarchical community discovery for multi-stage IP bearer network upgradation," *J. Netw. Comput. Appl.*, vol. 189, Sep. 2021, Art. no. 103151.

[11] S. Chen, Z.-Z. Wang, M.-H. Bao, L. Tang, J. Zhou, J. Xiang, J.-M. Li, and C.-H. Yi, "Adaptive multi-resolution modularity for detecting communities in networks," *Phys. A, Stat. Mech. Appl.*, vol. 491, pp. 591–603, Feb. 2018.

[12] F. Chen and K. Li, "Detecting hierarchical structure of community members in social networks," *Knowl.-Based Syst.*, vol. 87, pp. 3–15, Oct. 2015.

[13] P. Pons and M. Latapy, "Post-processing hierarchical community structures: Quality improvements and multi-scale view," *Theor. Comput. Sci.*, vol. 412, nos. 8–10, pp. 892–900, Mar. 2011.

[14] L. Zhou, K. Lü, P. Yang, L. Wang, and B. Kong, "An approach for overlapping and hierarchical community detection in social networks based on coalition formation game theory," *Expert Syst. Appl.*, vol. 42, no. 24, pp. 9634–9646, Dec. 2015.

[15] Z. Felfli, R. George, K. Shujaee, and M. Kerwat, "Community detection and unveiling of hierarchy in networks: A density-based clustering approach," *Appl. Netw. Sci.*, vol. 4, no. 1, p. 85, Dec. 2019.

[16] 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*, Dec. 2011, pp. 344–349.

[17] 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, Sep. 2007, Art. no. 036106.

[18] K. Berahmand and A. Bouyer, "LP-LPA: A link influence-based label propagation algorithm for discovering community structures in networks," *Int. J. Modern Phys. B*, vol. 32, no. 6, Mar. 2018, Art. no. 1850062.

[19] M. Lu, Z. Zhang, Z. Qu, and Y. Kang, "LPANNI: Overlapping community detection using label propagation in large-scale complex networks," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 9, pp. 1736–1749, Sep. 2019.

[20] J.-P. Attal, M. Malek, and M. Zolghadri, "Overlapping community detection using core label propagation algorithm and belonging functions," *Int. J. Speech Technol.*, vol. 51, no. 11, pp. 8067–8087, Nov. 2021.

[21] H. Shen, X. Cheng, K. Cai, and M.-B. Hu, "Detect overlapping and hierarchical community structure in networks," *Phys. A, Stat. Mech. Appl.*, vol. 388, no. 8, pp. 1706–1712, Apr. 2009.

[22] X.-Q. Cheng and H.-W. Shen, "Uncovering the community structure associated with the diffusion dynamics on networks," *J. Stat. Mech., Theory Exp.*, vol. 2010, no. 4, Apr. 2010, Art. no. P04024.

[23] E. L. Martelot and C. Hankin, "Multi-scale community detection using stability optimisation," *Int. J. Web Based Communities*, vol. 9, no. 3, pp. 323–348, 2013.

[24] M. T. Schaub, R. Lambiotte, and M. Barahona, "Encoding dynamics for multiscale community detection: Markov time sweeping for the map equation," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 86, no. 2, Aug. 2012, Art. no. 026112.

[25] P. Ronhovde and Z. Nussinov, "Local resolution-limit-free Potts model for community detection," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 81, no. 4, Apr. 2010, Art. no. 046114.

[26] V. A. Traag, P. Van Dooren, and Y. Nesterov, "Narrow scope for resolution-limit-free community detection," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 84, no. 1, Jul. 2011, Art. no. 016114.

[27] J. Huang, H. Sun, Y. Liu, Q. Song, and T. Weninger, "Towards online multiresolution community detection in large-scale networks," *PLoS ONE*, vol. 6, no. 8, Aug. 2011, Art. no. e23829.

[28] C.-C. Lin, J.-R. Kang, and J.-Y. Chen, "An integer programming approach and visual analysis for detecting hierarchical community structures in social networks," *Inf. Sci.*, vol. 299, pp. 296–311, Apr. 2015.

[29] C. Li, J. Bai, Z. Wenjun, and Y. Xihao, "Community detection using hierarchical clustering based on edge-weighted similarity in cloud environment," *Inf. Process. Manage.*, vol. 56, no. 1, pp. 91–109, Jan. 2019.

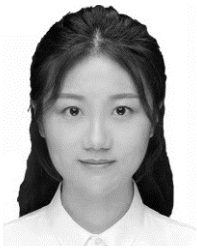
[30] S. E. Garza and S. E. Schaeffer, "Community detection with the label propagation algorithm: A survey," *Phys. A, Stat. Mech. Appl.*, vol. 534, Nov. 2019, Art. no. 122058.

[31] S. Gregory, "Finding overlapping communities in networks by label propagation," *New J. Phys.*, vol. 12, no. 10, Oct. 2010, Art. no. 103018.

[32] I. B. El Kouni, W. Karoui, and L. B. Romdhane, "Node importance based label propagation algorithm for overlapping community detection in networks," *Expert Syst. Appl.*, vol. 162, Dec. 2020, Art. no. 113020.

[33] T. Zhou, L. Lü, and Y.-C. Zhang, "Predicting missing links via local information," *Eur. Phys. J. B*, vol. 71, no. 4, pp. 623–630, Oct. 2009.

[34] C. Mu, Y. Liu, Y. Liu, J. Wu, and L. Jiao, "Two-stage algorithm using influence coefficient for detecting the hierarchical, non-overlapping and overlapping community structure," *Phys. A, Stat. Mech. Appl.*, vol. 408, pp. 47–61, Aug. 2014.
[35] M. E. J. Newman, "Modularity and community structure in networks," *Proc. Nat. Acad. Sci. USA*, vol. 103, no. 23, pp. 8577–8582, 2006.
[36] K. Shang, M. Small, Y. Wang, D. Yin, and S. Li, "A novel metric for community detection," *Europhys. Lett.*, vol. 129, no. 6, pp. 623–630, 2020.
[37] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Comput. Surv.*, vol. 45, no. 4, p. 43, 2013.



XUE ZHENG was born in Xuzhou, Jiangsu, China, in 1989. She received the B.S. and M.S. degrees in microelectronics from Xi'an Jiaotong University, Xi'an, China, in 2011 and 2013, respectively. She is currently a Lecturer with the National University of Defense Technology. Her research interests include network security and community detection of complex networks.

DONGQIU XING was born in Xi'an, Shaanxi, China, in 1977. She received the B.S. and M.S. degrees from Xi'an Shiyou University, Xi'an, in 2000 and 2007, respectively. During this period, she participated in the "863" project of China's deep-sea oil exploration instrument. She was mainly responsible for the research on the downlink technology of ground-to-underground communication. She is currently an Associate Professor with the National University of Defense Technology. Her research interest includes the application of electronic technology.

KEBIN CHEN was born in Turpan, Xinjiang, China, in 1987. He received the B.S. and M.S. degrees from Xi'an Jiaotong University, Xi'an, China, in 2011 and 2013, respectively, and the Ph.D. degree from the National University of Defense Technology, Wuhan, China, in 2023. He is currently a Lecturer with the National University of Defense Technology. His research interests include combat system-of-systems, complex networks, and systems engineering.

JING ZHAO was born in Linfen, Shanxi, China, in 1984. She received the B.S. degree from Lanzhou Jiaotong University, Lanzhou, China, in 2007. She is currently a Lecturer with the National University of Defense Technology. Her research interests include electronic technology and the application of electronic components.

YUNJUN LU was born in Kaifeng, Henan, China, in 1973. He received the B.S. degree from the Guangzhou Communication College, Guangzhou, China, in 1994, the M.S. degree from the Communication Command College, Wuhan, China, in 2000, and the Ph.D. degree from the Second Artillery Command College, Wuhan, in 2008. He is currently a Professor with the National University of Defense Technology. His research interests include operational research, systems engineering, and system simulation.