Applied Soft Computing Journal 154 (2024) 111414

Contents lists available at ScienceDirect

Applied Soft Computing

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

# Evolutionary multi-objective overlapping community detection based on fusion of internal and external connectivity and correction of node intimacy

Ronghua Shang [a,*], Sa Wang [a], Weitong Zhang [a], Jie Feng [a], Licheng Jiao [a], Rustam Stolkin [b]

[a] Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, School of Artificial Intelligence, Xidian University, Xi'an, Shanxi Province, 710071, China

[b] The Extreme Robotics Lab, University of Birmingham, UK

ARTICLE INFO

*Keywords:*
Evolutionary multi-objective optimization
Overlapping community detection
Node intimacy
Attribute information
Community fusion

ABSTRACT

In the field of community detection, node attribute information plays an important role in community division. Existing methods use topology structure and node attribute information to discover *non-overlapping* communities. However, so far, attribute information has not been fully utilized in *overlapping* community detection. To address this, we propose a new overlapping community detection method called "evolutionary multi-objective overlapping community detection based on Fusion of internal and external Connectivity and Correction of Node Intimacy" (FCCNI). Firstly, we propose a fusion strategy based on internal and external connectivity, which integrates some communities with sparse intra-connections and dense inter-connections. This automatically determines, reconfirms, and corrects the number of communities. Secondly, a function is designed to calculate the intimacy between nodes, and the node label with the highest intimacy is selected to correct the current wrong node. The correction strategy is used in two stages of initialization and multi-objective evolution to obtain a more accurate node label. Finally, a method that considers not only the connections of the community, but also the node attribute, is designed to obtain the overlapping community indirectly from the non-overlapping community. The experimental results on five real-life networks and four classical synthetic networks show that FCCNI achieves better overlapping community division, compared with six state-of-the-art comparison algorithms from the literature.

## 1. Introduction

Networks are a powerful representation of the interactions between connected entities, which emerge in a wide variety of real-world systems. Networks are used to describe complex relationships in numerous applications such as social networks [1], biological protein networks [2], science citation networks [3], etc. Entities, and the relationships between them, are respectively abstracted into nodes and edges in the network. Revealing and discovering the community structure of the network helps to understand the nature of the network more clearly [4]. The purpose of community detection is to identify node clusters with special structures that meet certain homogeneity criteria. Communities are groups of entities whose interconnections are more densely connected than the rest of the network [5]. Community detection is one of the key issues in the field of complex networks [6]. With the widespread application of information technology, networks are reflecting increasingly complex relationships. Nodes are often coupled with attributes describing their characteristics, which poses challenges for traditional methods [7]. Additionally, although the discovery of disjoint communities is the basis for a better understanding of networks, objectively speaking, there may be some overlap in the networks in practical applications. Overlapping nodes belonging to and connecting multiple communities will have a significant impact on the information flow. Therefore, the detection of overlapping communities in attributed networks remains a challenging research question.

A method of evolutionary multi-objective overlapping community detection based on fusion of internal and external connectivity and correction of node intimacy(FCCNI) is proposed in this paper. Firstly, the initial stage is supplemented by a community fusion strategy, which discovers communities with sparse intra-connections and then fuses them with another most dense connected community. It prevents the emergence of too many small communities and effectively corrects the number of communities in the initial stage. Secondly, a function to calculate the mutual intimacy of a given node is designed based on the common neighbor information of the given node and any other nodes. The label of the node with the highest intimacy is selected to correct the label of the current wrong node. The labels of nodes are corrected in the two stages of initialization and evolution

* Corresponding author.
*E-mail address:* rhshang@mail.xidian.edu.cn (R. Shang).

https://doi.org/10.1016/j.asoc.2024.111414
Received 4 October 2022; Received in revised form 8 January 2024; Accepted 11 February 2024
Available online 17 February 2024

so as to be more accurate. Finally, a strategy is designed to obtain overlapping communities indirectly from non-overlapping communities, in situations where label-based representations cannot encode overlapping communities well. The strategy not only considers the internal and external connections of the communities in which a node lies, but also takes the node's attributes into consideration to make the judgment of overlapping nodes stricter and more accurate. The nodes in overlapping communities have higher attribute homogeneity, and a relatively higher quality overlapping community can be obtained.

The remainder of this paper is structured as follows: Section 2 discusses related work. Section 3 introduces the principle of FCCNI in detail. Section 4 gives the experimental results and detailed analysis of five real-life networks and four classical synthetic networks. Section 5 provides concluding remarks.

## 2. Related work

In 2002, Girvan and Newman [2] were the first to pioneer the community detection problem based on the idea of partition, that is, the GN algorithm, which has been studied by many scholars and further proposed algorithms based on the idea of partition. Initially, these GN-derived algorithms tend to divide the network into several separate communities. The modularity concept [8] was proposed to describe the quality of the community division, and it was redefined in 2006 [9]. Community detection methods were then divided into two categories, namely optimization-based and non-optimization-based. Optimization-based algorithms such as the hybrid genetic algorithm [10], "Memetic-net" algorithm [11], and the genetic algorithm named GA-net [12], have played an important role in the field of community detection. Ma et al. proposed a fast memetic algorithm for community detection, in which the modularity was optimized and a multi-level learning strategy was used to accelerate the optimization process [13]. Zhang et al. proposed a network reduction method to reduce the size of the networks in the evolution process, which works well for community detection in large-scale complex networks [14]. Many non-optimization-based methods have also been proposed for community detection. Such as the label propagation algorithm proposed by Raghavan et al. [15], and an algorithm that found clusters based on random walks [16]. Some state-of-the-art techniques based on graph embedding strategies and deep learning are also examples of non-optimization-based methods for community detection. For example, Zhang et al. proposed a graph embedding method based on the shortest path matrix [17]. Zhu et al. proposed a structural embedding method based on non-negative matrix decomposition, which embeds the two proposed indicators of proximity and similarity into a low-dimensional vector space [18].

With the exploration of complex networks, increasing attention has been focusing on detecting overlapping communities. A variety of overlapping community detection strategies have been proposed, such as label propagation algorithm [19], link division method [20], fuzzy detection [21] and clique methods [22]. In recent years, Zhang et al. proposed a mixed representation-based algorithm for fast and effective overlapping community detection in complex networks, in which two evolving parts representing overlapping and non-overlapping nodes were used [23]. Tian et al. proposed an evolutionary multi-objective optimization-based fuzzy method, in which the community centers were optimized and the appropriate fuzzy thresholds for each node were identified to uncover diverse overlapping community structures [24]. In the algorithm proposed by El Kouni et al. the properties of the nodes were utilized to improve the label propagation process, and a new filtering method was proposed to remove unnecessary labels [25]. Ramesh et al. proposed a merged maximal clique representation method. They introduced a modified normalization factor in the definition of link strength between merged-maximal cliques, which identifies weak connections in the merged-maximal clique graph, and make the connections between merged-maximal cliques meaningful [26]. Ma et al. proposed an algorithm for overlapping community detection in large-scale complex networks, in which a local-to-global scheme that included the two stages of local community structure detection and global community structure determination was proposed [27]. In the algorithm proposed by Roy et al. the similarity between neighbor nodes based on improved random walk was calculated, and a fuzzy membership function was used to iteratively calculate the membership of all nodes to the existing community [28].

However, with the widespread application of information technology, the entity nodes are often coupled with attributes describing their characteristics. The attributed networks enrich and supplement the information of nodes by using a set of feature labels. The attribute can play a useful role in obtaining more meaningful community distribution [29]. This may help to avoid obtaining community divisions with sparse intra-connections in networks whose degree distribution is scale-free [30]. Researchers have proposed a variety of methods for processing attribute information. Some research has focused on designing new distance metrics by integrating topology and attribute, and then applying classical community detection algorithms to discover communities in these networks. For example, Zhou et al. proposed a metric that incrementally updated random walk distances when the given edge weight increased [31]. Ruan et al. proposed an approach of combining content and link information for community detection [32]. Furthermore, the community detection was transformed into a multi-objective optimization problem. For example, in Li et al. [33] and Moayedikia [34], objective functions based on the structure and attribute were optimized for community detection. Other research includes models that fuse attribute and topological structures. For example, Xu et al. proposed a Bayesian probabilistic model for attributed graphs [35]. Wang et al. proposed an innovative non-negative matrix factorization (NMF) mode, with two sets of parameters describing the community membership matrix and the community attribute matrix [36]. In addition, some algorithms based on attribute and topological representation learning mainly focused on the low-dimensional vectors of nodes and the embedding clusters. For example, Li et al. proposed a novel method for encoding inherent community structures through community structure embedding [37]. Hong et al. proposed a deep network embedding framework that was composed of personalized random walks, the enhanced matrix representation, and a deep neural network [38].

Bothorel et al. [39] believed that a good community division in the attributed network must optimize the structure and attribute characteristics simultaneously, to obtain communities with dense intra-connections and high attribute similarity. In some cases, information from attribute and structure are of two different properties that may contradict each other [36], making community detection in attributed networks difficult. The multi-objective optimization algorithm [40] shows strong superiority in dealing with the optimization problem of conflicting objectives. As a classical and mainstream algorithm for solving multi-objective optimization problems, NSGA-II [41] has played a significant role in optimization problems. Li et al. proposed a new function $S_A$ to measure the attribute similarity within the community and a new multi-individual-based mutation operator was designed under the framework of NSGA-II [33]. Sun et al. proposed a graph neural network encoding strategy in the multi-objective evolutionary algorithm (MOEA) for community detection in complex attributed networks [42]. However, these studies only focus on non-overlapping community detection in attributed networks.

In recent research about overlapping community detection of attributed networks, the algorithm proposed by Teng et al. aimed to find overlapping communities in directed and undirected networks. The decoding method considered community fitness to determine whether the given nodes are overlapping nodes [43]. In the method proposed by Reihanian et al. extended modularity and $SimAtt$ (a metric to

measure the similarity of attributes) were taken as the two optimization objectives. An extended locus-based adjacency representation is introduced in the encode and decode process [44]. Compared with the original trajectory-based adjacency representation method with only two stages, "Marking" and "Final Decoding" are added to the decoding of overlapping communities. He et al. designed a graph convolutional autoencoder to automatically fuse the information of structure and attribute [45].

In this paper, an overlapping community detection method is proposed, based on the fusion of internal and external connectivity and correction of node intimacy in the framework of evolutionary multi-objective optimization. In contrast to recent research on overlapping community detection of attributed networks, FCCNI takes comprehensive consideration of both structure and attributes when determining overlapping nodes. Additionally, the encoding and decoding method is supplemented by a fusion strategy based on internal and external connectivity. The main contributions of this paper are as follows.

(1) The number of communities is automatically initialized by using a community integration strategy based on the internal and external connections of the initial communities. This reconfirms and effectively corrects the number of communities.

(2) To make full use of the topology, the node label is corrected multiple times based on the degree of intimacy in the two stages of initialization and multi-objective evolution.

(3) In the process of indirectly obtaining and representing overlapping communities, we propose a formula that considers both the connections and attributes of nodes within the community. This formula is used to determine whether a given node is an overlapping node. Through the calculation of this formula, it is determined whether a given node is an overlapping node, so as to improve the quality of overlapping community detection.

## 3. The FCCNI algorithm

There are several problems to be solved in attributed networks. Existing methods in the literature may suffer from some of the following issues. Firstly, it is possible to obtain some unnecessary small communities due to the random selection of adjacent nodes. Secondly, the adjacent node information may not be fully utilized when determining the node label. Finally, the attribute and structure information may not be fully considered when obtaining overlapping communities from non-overlapping communities. To solve these problems and make full use of node attribute, an overlapping community detection algorithm based on fusion of internal and external connectivity, and correction of node intimacy, is proposed in this paper. The framework of FCCNI is shown in Fig. 1.

Our proposed FCCNI approach incorporates the following methods: initialization and community fusion based on internal and external connections; label modification based on node intimacy; and overlapping community division based on comprehensive consideration of structure and attribute. In Section 3.1 the main parts of FCNNI and the overall algorithm will be introduced.

### 3.1. Objective function

To obtain good overlapping community division in attributed networks, the quality of topology and the attribute similarity of nodes within the community must be optimized simultaneously. However, these different kinds of information may be conflicting. The following multi-objective problem needs to be optimized:

$$maximize\ F = (f_S(x), f_A(x)) \tag{1}$$

To measure the structure of the overlapping communities better, Shen et al. extended the classical $Q$ to $EQ$ [46], which is defined as:

$$f_S = EQ = \frac{1}{2m}\sum_{k=1}^{|C|}\sum_{v,w\in C_k}\frac{1}{O_v O_w}\left(A_{vw} - \frac{d_v d_w}{2m}\right) \tag{2}$$

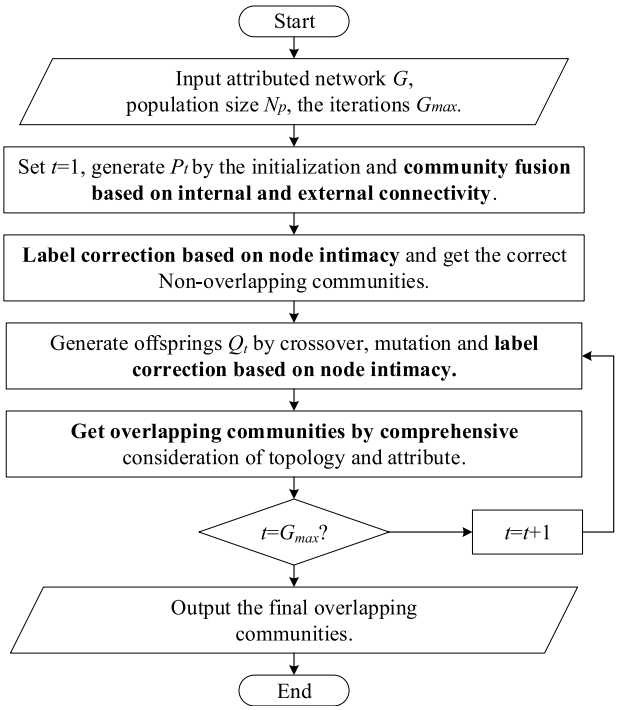


**Fig. 1.** Framework of FCCNI.

where $m$ is the number of edges, $|C|$ and $|C_k|$ represent the total number of communities and the number of nodes in the $k$th community respectively, $O_v$ and $O_w$ represent the number of communities to which node $v$ and node $w$ belong respectively. $A$ is the adjacency matrix of the network and $d$ is the node degree. As the first objective function, $EQ$ mainly evaluates the quality of the detected community based on topology. It is the proportion of edges belonging to a given cluster in real community division, minus the expected such proportion in random graphs. This has been shown to be effective in measuring the distribution of overlapping communities. Another function that measures the attribute similarity within the communities is defined in [33] as follows:

$$f_A = A_S = \sum_{k=1}^{|C|}\frac{\sum_{v,w\in C_k} s_{vw}}{|C_k|\left(|C_k|-1\right)} \tag{3}$$

where $s_{vw}$ is the judgment value of whether the attribute values of nodes $v$ and $w$ are the same. $s_{vw}$ is 1 only when the attribute values of these two nodes are equal, otherwise it is 0. Specifically, for such single-attributed networks $G=(V, E, \mathbb{A})$, $\mathbb{A}_i$ $(1 \le i \le |V|)$ is the attribute value of node $V_i$ and is one dimensional, taking discrete value. $\sum_{v,w\in C_k} s_{vw}/2$ is the real number of node pairs with the same attribute that fall in a given community. $|C_k|\left(|C_k|-1\right)/2$ is the expected fraction if nodes were associated at random. The attribute similarity $A_s$ measures the degree of homogeneity of the nodes within all the given groups by calculating the ratio of two such parts. The larger $A_s$ value means a greater attribute similarity within the community and a smaller attribute similarity between communities.

### 3.2. Initialization and community fusion based on internal and external connection judgment

In order to apply MOEA-based algorithms for community detection in complex networks, an individual encoding method must be developed. Location-based representation [47] and label-based representation [48] are commonly used in evolutionary community detection algorithms. They help to get a better initialization result based on

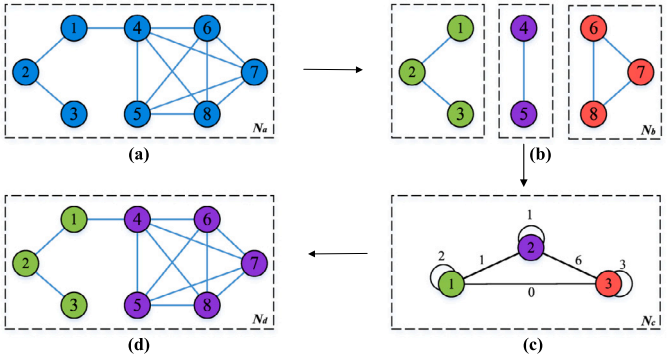


**Fig. 2.** The process of initialization and community fusion.

**Table 1**
Location-based representation of network $N_a$.

| Node | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Genotype | 2 | 3 | 2 | 5 | 4 | 8 | 6 | 7 |

topology and facilitate the development of the subsequent algorithm. However, due to the random selection of the neighbors in the representation, sometimes too many small-scale communities will be detected in the community detection. Such community divisions may ignore the macro-topological connections of the whole network, resulting in poor initial community division. Thus, based on the principle that the intra-connections in the communities are dense, and the inter-connections of the communities are sparse, the communities with dense inter-connections are fused in FCCNI. The number of communities is also reconfirmed and corrected. Specifically, for an attributed network $G$=($V$, $E$, $\mathbb{A}$), the network is encoded as $L$=($X_1$, $X_2$, …, $X_{|V|}$). The genotype $X_i$ represents the community to which node $i$ belongs, based on continuous encoding of the communities. The process of the initialization and community fusion is shown in Fig. 2. Network $N_b$ is the community division according to the location-based encoding in Table 1.

In Table 1, the genotype represents the node randomly selected from the neighbors of a given node, within the same community. Network $N_c$ is relabeled continuously according to network $N_b$ by the label-based representation. Network $N_d$ is the encoding result after fusing communities whose intra-connections are fewer than the inter-connections in network $N_c$. The selected community 2 in Fig. 2 is fused with the community of the most inter-connections.

The initialization and community fusion based on internal and external connections is summarized as Algorithm 1.

**Algorithm 1** Initialization and community fusion based on internal and external connections

**Input:** Attributed network $G = (V, E, \mathbb{A})$.
**Output:** Non-overlapping genotype $L_0$ after community fusion.

```
Get genotype L = {X_1, ..., X_|V|} by the initialization;
Get the partitioning C = {C_1, ..., C_k} according to L;
for each C_i do
    Calculate the connections between C_i and others to get S_i = {S_i1, ..., S_ik};
end for
Set the number of the merged communities m = 0;
for i = 1 → k do
    Let M = max(S_i);
    if S_ii ≠ M then
        for j = 1 → k do
            if S_ij = M and C_j ≠ ∅ then
                C_j = C_j ∪ C_i;
                C_i ← ∅, m ← m + 1;
            end if
        end for
    else
        m ← m + 1;
    end if
end for
Consecutively recode the community from 1 to m and update the node label according to this to get the updated L_0;
```

**Return** $L_0$.

### 3.3. Label correction based on node intimacy

#### 3.3.1. Node intimacy

Nodes that share a large number of neighbors indicate that there is a strong mutual relationship between them [49]. A greater number of common neighbors indicates a denser relationship between two nodes. Additionally, if there are edges between them, the mutual relationship will be higher than the case where the two nodes are not connected directly. Therefore, the strongest mutual relationship is the case where the two nodes are not only connected directly but also share a large number of common neighbors. A less strong relationship is indicated when two nodes are not directly connected, but still have a large number of common neighbors. The weakest relationship is the case where the two nodes are neither connected nor have any common neighbors. In a network $G$=($V$, $E$, $\mathbb{A}$), the importance of the mutual relationship with another node $j$ ($1 \le j \le |V|$, $j \ne i$), namely the intimacy for a given node $i$ ($1 \le i \le |V|$) is as follows($F_{ij} \ne F_{ji}$).

$$F_{ij} = \begin{cases} \frac{|N_i \cap N_j|}{|N_i|} & A_{ij} = 0 \\ \frac{|N_i \cap N_j|+1}{|N_i|} & A_{ij} = 1 \end{cases} \tag{4}$$

where $N_i$ (resp. $N_j$) represents the neighbor node set of node $i$ ($1 \le k \le |V|$) (resp. node $j$), and $|N_i|$ is the number of neighbors. $A = [A_{ij}]_{(|V|\times|V|)}$ is the adjacency matrix of the network. If there is an edge that connects node $i$ and node $j$ in the abstract complex network, $A_{ij}$ is recorded as 1. Respectively, if there are associations between two nodes in real-life, $A_{ij}$ is recorded as 1, otherwise $A_{ij}$=0. $F_{ij}$ describes the importance of the mutual relationship for the given node $i$. Here $j$ represents any other nodes rather than $i$. Accordingly, $F_{ji}$ describes the importance of mutual relationship for the given node $j$. Here $i$ represents any other nodes rather than $j$. The value of intimacy ranges from 0 to 1.

Since the number of neighbors of each node is different, the influence of neighbors on nodes will also be dispersed as the number of neighbors increases. Therefore, the interaction between each two nodes also varies with the situation of their neighbors. For the nodes with relatively few neighbors, the impact of the mutual relationship on themselves is relatively greater. In the same case, for the nodes with more neighbors, the impact may be relatively small. These situations are fully considered when designing the intimacy function.

#### 3.3.2. Node relationship sorting and label correction

The node intimacy for a given node can be obtained by the definition in formula (4). The value of the intimacy determines the importance of mutual relationship for the given node. The greater the intimacy value, the denser the relationship between nodes, and vice versa. For any given node $i$, all nodes $j$ ($1 \le j \le |V|$, $j \ne i$) can be sorted according to the degree of intimacy. For the nodes that may be divided incorrectly, the node label with the highest degree of intimacy can be

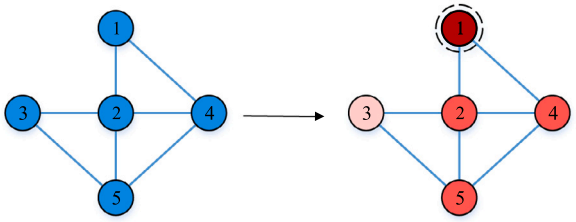


**Fig. 3.** Node relationship network.

**Table 2**
Node intimacy.

| Node | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 0.5 | 1 | 1 |
| 2 | 0.5 | 0 | 0.5 | 0.75 | 0.75 |
| 3 | 0.5 | 1 | 0 | 1 | 1 |
| 4 | 0.67 | 1 | 0.67 | 0 | 0.67 |
| 5 | 0.67 | 1 | 0.67 | 0.67 | 0 |

selected for correction in the two main stages of the algorithm. The label of the node with the highest intimacy is selected for correction, so that the topology of the graph can be more fully utilized in the selection of adjacent nodes. The correction of node labels will be also more accurate. Fig. 3 shows the node relationship network.

In the network shown in Fig. 3, the intimate relationship of the given nodes can be obtained from the initial network and their connections. The intimate degree is calculated by the formula (4) and the intimacy values for each node, with respect to neighboring nodes in the rows as shown in Table 2. Taking node 1 in the network as the focus in Fig. 3, the intensity of the color denotes the strength of the intimacy.

In Table 2, the intimacy for the given node in each row is accurately quantified by the proposed intimacy function. The relationships of the neighbors are fully considered in the intimacy, and the connections of nodes themselves are also considered. Thus the ranking of node intimacy is easy to generate.

Although the initialization process in Section 3.2 makes full use of the topology and obtains a good initialization result, there are still inaccurate divisions of multiple nodes due to the randomness of selecting neighbor nodes. At the same time, the genotypes of individuals are positive numbers that represent the community label after the initialization. Traditional crossover operators are inappropriate for our algorithm and the changes in traditional mutation operators tend to be too random. Instead, we adopt a two-way crossover [11] that selects subclass as crossing points, and a multi-individual-based mutation operator that applies heuristic information [33]. However, there is still a problem in the neighborhood correction strategy of mutation operators if the nodes to be corrected are randomly selected. Randomly selecting nodes for label correction will reduce the search and correction efficiency of label correction. Therefore, FCCNI mainly performs the label correction in two stages of the algorithm. Firstly, at the end of the initialization stage, the node label is corrected when all neighbors of the current node are not in the same community. Secondly, in the process of evolution, when performing the mutation operator based on multiple individuals, some wrongly labeled nodes may need to be corrected. In this case, the appropriate nodes are selected based on the ranking of intimacy, so that the correction is carried out in a more appropriate direction. The label correction strategy based on the ranking of node intimacy after community fusion is shown in Algorithm 2.

### 3.4. Overlapping community division based on comprehensive consideration of structure and attribute (OCCSA)

The label-based representation makes it is easier to design evolutionary operators, but cannot be used to represent overlapping nodes.

**Algorithm 2** Node modification after the community fusion

**Input:** Encoding result $L_0$ of attributed network $G$=($V$, $E$, $\mathbb{A}$), A cluster $NL = \{NL_1, ..., NL_{|V|}\}$ such that $NL_i$ is the label collection of all neighbors of node $i$.
**Output:** Corrected non-overlapping encoding result $L_1$.

**Let** $IM$ be the intimacy matrix of size $|V| \times |V|$ such that $IM(i, j)$ is the intimacy of the given node $i$ calculated by the formula (4);
$L_1 \leftarrow L_0$;
**for** $i = 1 \rightarrow |V|$ **do**
  **Let** $M = \max(IM(i, :))$;
  $S_i \leftarrow \emptyset$;
  **for** $j = 1 \rightarrow |V|$ **do**
    **if** $IM(i, j) = M$ **then**
      $S_i \leftarrow S_i \cup j$;
    **end if**
  **end for**
  **if** $|Set(NL_i)| \neq 1$ **then**
    **if** $|S_i| \neq 1$ **then**
      Randomly select a node $k$ from $S_i$ and modify $L_1$[i] as $L_1$[k];
    **else**
      modify $L_1$[i] as $L_1[S_i[0]]$;
    **end if**
  **end if**
**end for**
**Return** $L_1$.

A strategy to obtain overlapping communities based on comprehensive consideration of structure and attribute (OCCSA) is designed in this paper, so as to achieve the representation that a single node is in multiple overlapping communities. For attributed networks, the attribute may also play an important role in avoiding community division with sparse internal communication and obtaining more meaningful overlapping communities. Based on this, the process of obtaining overlapping communities reuses structure information and also uses the attribute to find overlapping nodes in FCCNI.

Firstly, by considering the internal and external connectivity of nodes in a community, a community fitness function is defined to mark the quality of a community in [50] as follows.

$$f(C_k) = \sum_{i \in C_k} \frac{d_i^{in}}{(d_i^{in} + d_i^{out})^{\alpha}} \tag{5}$$

where $d_i^{in}$ represents the degree of node $i$ inside the community $C_k$, which is also the number of connections between node $i$ and the internal nodes of the community $C_k$. Similarly, $d_i^{out}$ represents the degree of node i outside the community $C_k$.$\alpha$ is a positive real value parameter that plays a role in controlling the size of the community and it is set to 1 in this paper. However, this formula only considers the topology of the network and fails to make full use of the attribute information. This paper extends the formula and defines an evaluation function $f_{cs}$, which considers both the topology and attribute of the nodes in a single community. This functions as the criterion of judging whether the current node is an overlapping node. The definition of $f_{cs}$ is shown in formula (6).

$$f_{cs} = \lambda \sum_{i \in C_k} \frac{d_i^{in}}{d_i^{in} + d_i^{out}} + (1 - \lambda) \frac{2 \sum_{i,j \in C_k, i<j} s_{ij}}{N_{c_k}\left(N_{c_k} - 1\right)} \tag{6}$$

where $N_{c_k}$ represents the number of nodes in community $C_k$, $N_{c_k}(N_{c_k}$-1)/2 is the expected number of edges when the nodes in community $C_k$ are fully connected, and $s_{ij}$ is the judgment value of whether the attribute values of node $i$ and $j$ are the same. Only when the attribute values are equal, $s_{ij}$ is 1, otherwise it is 0. $\lambda$ is an adjustable parameter that controls the weight of community fitness and attribute similarity respectively. The value analysis of $\lambda$ will be in the next section of this

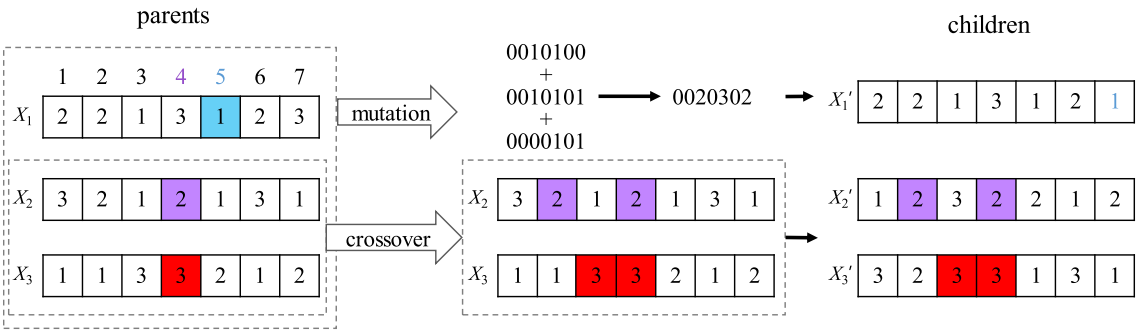


**Fig. 4.** Two-way crossover operator and multi-individual-based mutation operator.

paper. The criterion for FCCNI to judge whether the node is overlapping is: calculate and compare whether the $f_{cs}$ value of the community becomes larger after each new node joins the community. The node joins the community and becomes an overlapping node only when the $f_{cs}$ value increases.

In the remainder of this paper, the following compact formula is used to represent the process of obtaining overlapping communities.

$$C_o = OCCSA(G, L, f_{cs}) \tag{7}$$

This means that, given an attributed network $G$, the non-overlapping encoding result $L$ and the calculated $f_{cs}$ are used to judge whether the current node is an overlapping node, the overlapping division $C_o$ is obtained subsequently. $C_o$ is a set of overlapping communities of $G$,

### 3.5. The framework of FCCNI

Sections 3.2–3.4 explained: the initialization and community fusion based on the judgment of internal and external connections; the label correction based on node intimacy; and the overlapping community division, based on structure and attribute (OCCSA). This section discusses the overall algorithm. FCCNI is under the framework of NSGA-II [41], in which a fast non-dominated sort is proposed to evaluate all individuals and generate non-dominated sets of different levels.

For all individuals in the population, two statistics are required in fast non-dominated sort: (1). $n(p)$ denotes the number of solutions that are dominating the solution $p$, called domination count. (2). $S(p)$ denotes the set of solutions dominated by the solution $p$. For all solutions in the first non-dominant front, the number of solutions dominating them is 0, i.e., $n(p)$=0 for each solution $p$. Iterate over each individual in $S(p)$ at this point, subtracting its domination count by 1. If the domination count of some solutions in $S(p)$ becomes 0, then these solutions belong to the second non-dominant solution front. And so on, until all non-dominant solution fronts are determined [41].

After sorting the solutions based on the non-domination rank, the crowding distances of the solutions within each level are calculated. To calculate the crowding-distance, the population solution is sorted in ascending order by each objective function value. Then, according to the crowding-distance assignment method in [41], the distance values of the boundary solutions and the intermediate solutions corresponding to each objective function are calculated and normalized. The overall crowding distance value is the sum of the individual distance values corresponding to each objective function.

In each generation, binary tournament selection is used to generate offspring according to the rank and crowding distance of each solution. We adopt a two-way crossover operator [11] that selects subclass as crossing points, and a multi-individual-based mutation operator [33] that applies heuristic information. The two-way crossover operator and multi-individual-based mutation operator are presented in Fig. 4.

As shown in Fig. 4, three individuals are selected as parents through binary tournament selection. A new individual is generated by the multi-individual-based mutation operator. Specifically: one of the three individuals, assumed to be $X_1$, is randomly selected as the target individual. Then randomly select one gene in $X_1$, assuming gene 5, as the target gene. Mark the genes in $X_1$ with the same label as the gene 5 as "1", and mark the remaining genes as "0". The same goes for $X_2$ and $X_3$. Marker values from three individuals were added bitwise. The results of the addition serve as a criterion for updating the $X_1$ gene, refers to [33]. For example, in Fig. 4, the sum of marker results for not only one gene position is greater than or equal to "2", and the sum of marker results for no gene position is equal to "1". According to the repair criteria in [32], the label of the gene with a value equal to or greater than "2" is modified to the label of the target gene 5 in $X_1$. Therefore, the label of gene 7 in $X_1$ is modified to "1", and the other position labels remain unchanged. In the two-way crossover operator, two individuals $X_2$ and $X_3$ are chosen as parents and the gene 4 is randomly selected as the target gene. Then the genes in $X_2$ and $X_3$ in the same cluster as gene 4 are easy to determine. The two-way crossover operation is: $X_2'$ is to retain the gene label in the same cluster as gene 4, and replace the rest of the position labels with the corresponding position label in $X_3$. Similarly, for $X_3'$, retain the gene label in the same cluster as gene 4, and replace the rest of the position labels with the corresponding labels in $X_2$. Finally, two new individuals after the crossover are obtained.

In summary, three individuals are selected for crossover and mutation. In a mutation operation, one of the three individuals is selected as the target individual, and the other two individuals provide the label information. One new individual is generated after each mutation operation. In a crossover operation, one individual is selected as the target individual, and then another individual is selected to participate. Two new individuals are generated after each crossover operation.

The overall pseudo code of the algorithm is given in Algorithm 3.

It can be seen that the number of times for varying the targets depends on the number of individuals in the population $N_p$. This is because (via binary tournament selection, two-way crossover, multi-individual-based mutation operator, and label selection based on node intimacy) the same number of offspring as the number of population individuals $N_p$ should be obtained. $G_{max}$ represents the total maximum number of iterations.

## 4. Experimental results and analysis

All experiments are performed on a computer with a processor of Intel (R) i5-4590 h CPU @ 3.30 GHz, a memory of 8 GB and the operating system Windows 10. The values of evaluation metrics are recorded when running on each network. In the process of evolution, three individuals are selected and generated in one operation of crossover and mutation. The number of individuals in the population should be 3*$k$ ($k$ is a positive integer). Since the number of individuals in a population is typically set to 100 in related algorithms, the number of individuals in a population is set to 102 in FCCNI. The maximum number of evolutionary generations is set to 50. The crossover

**Algorithm 3** The framework of FCCNI

**Input:** Attributed network $G$=($V$, $E$, $\mathbb{A}$), number of individuals in a population: $N_p$, the maximum number of iterations: $G_{max}$.
**Output:** $PF_o$: The obtained overlapping community partitions in the Pareto front.

$t \leftarrow 1$;
Initialize population $P_t = \{p_1, ..., p_{Np}\}$ based on the initialization of **Algorithm 1** and then corrected by **Algorithm 2**;
**repeat**
  Offsprings $Q_t \leftarrow \emptyset$;
  **repeat**
    $[X_1, X_2, X_3] \leftarrow$ binary tournament selection ($P_t$);
    $[X_1'] \leftarrow$ multi-individual-based mutation and **further corrected by label selection based on node intimacy**;
    $[X_2', X_3'] \leftarrow$ two-way crossover on ($X_2, X_3$);
    $Q_t \leftarrow Q_t \cup [X_1', X_2', X_3']$;
  **until** $|Q_t| = N_p$
  **for** $k = 1 \rightarrow N_p$ **do**
    Calculate the $f_{cs}$ of each community in $P_{tk}$ and $Q_{tk}$;
    $P_{tko}$ = **OCCSA** ($G, P_{tk}, f_{cs}$), $Q_{tko}$ = **OCCSA** ($G, Q_{tk}, f_{cs}$);
    Calculate the two objective functions (1) and (2) based on $P_{tko}$ and $Q_{tko}$;
  **end for**
  $F \leftarrow$ fast non-dominated sort($P_t \cup Q_t$);
  $C \leftarrow$ the crowding distance of $F$;
  $P_{t+1} \leftarrow \emptyset$, $i \leftarrow 1$;
  **while** $|P_{t+1}| + |F_i| < N_p$ **do**
    $P_{t+1} \leftarrow P_{t+1} \cup F_i$;
    $i \leftarrow i + 1$;
  **end while**
  Sort $F_i$ in descending order based on $C_i$;
  $P_{t+1} \leftarrow P_{t+1} \cup F_i[1 : (N_p - |P_{t+1}|)]$, $t \leftarrow t + 1$;
**until** $t = G_{max}$
**Return** $PF_o$: The obtained overlapping community partitions in the Pareto front.

probability determines the genetic diversity of the population, and the mutation probability determines the global search ability of the population. Guided by experience of EA-based algorithms, the crossover and mutation probabilities are set to 0.9 and 0.1 respectively. In the following experimental sub-sections, the probabilities of crossover and mutation will be further analyzed.

### 4.1. Datasets

The synthetic networks used in the experiment are generated according to [51]. The generation parameters of LFR network mainly include: $N$ (number of nodes), $k$ (average node degree), $k_{max}$ (maximum node degree), $\tau_1$ (degree distribution index), $\tau_2$ (community size distribution index), $C_{min}$ (minimum community size), $C_{max}$ (maximum community size), $O_n$ (number of overlapping nodes), $O_m$ (number of communities to which overlapping nodes belong) and $\mu$ (mixed parameters). In the following experiments, four typical synthetic networks with different characteristics are designed to verify the effectiveness of FCCNI.

Parameters $N$, $\mu$, $O_n$ and $O_m$ are mostly related to overlapping communities. Thus for the sake of fairness, the above four parameters of the four classical networks are different. The others are set with the same parameters as follows: $k$=5, $k_{max}$=25, $\tau_1$=2, $\tau_2$=1, $C_{min}$=20, $C_{max}$=80. The different parameters are set as shown in Table 3.

The real-life networks used in the experiment are five networks with discrete node attributes, namely Zachary Karate Club (karate) [52], dolphins network (dolphins) [53], political books network (polbooks) [54], American college football team network (football) [2] and British political network (politics_uk) [55]. "karate" is a social network. The attribute of the nodes in the karate dataset is to which club each node (member) belongs to. "dolphins" is a network with nodes representing the wide-nosed dolphins living off Doubtful Sound. The attribute of the nodes in the dolphins dataset is which leader each node (dolphin) supports. "polbooks" is a network with nodes representing books sold by Amazon during 2004. The attribute of the nodes in polbooks dataset is the political leanings of each node (book). The attribute values can be conservative, liberal or neutral. "football" is a complex social network of American college football teams. The attribute of the nodes in football dataset is which league each node (football team) belongs to. "politics_uk" is a dataset collected from Twitter in 2012. The attribute of the nodes in politics_uk dataset is the political leanings of each node (user). The attribute values can be conservative, labor, libdem, snp and other. The basic information of all networks are given in Table 4, in which $\bar{d}$ represents the average degree and $N_A$ is the number of attribute values.

**Table 3**
Parameter settings of LFR.

| Network | $N$ | μ | $O_n$ | $O_m$ |
|---|---|---|---|---|
| LFR0 | 1000 | 0.1 | 300 | 2 |
| LFR1 | 1000 | 0.2 | 300 | 2 |
| LFR2 | 1000 | 0.1 | 300 | 3 |
| LFR3 | 5000 | 0.1 | 1500 | 2 |

**Table 4**
The basic information of all networks of the experiments.

| Network | Nodes | Edges | $\bar{d}$ | $N_A$ |
|---|---|---|---|---|
| karate | 34 | 78 | 4.6 | 2 |
| dolphins | 62 | 159 | 5.1 | 2 |
| polbooks | 105 | 441 | 8.4 | 3 |
| football | 115 | 615 | 10.7 | 12 |
| politics_uk | 419 | 27 340 | 130.5 | 5 |
| LFR0 | 1000 | 4586 | 9.2 | 29 |
| LFR1 | 1000 | 4832 | 9.7 | 33 |
| LFR2 | 1000 | 4510 | 9.6 | 41 |
| LFR3 | 5000 | 23 178 | 9.3 | 147 |

### 4.2. Comparison algorithms

Six overlapping community detection algorithms are compared to verify the effectiveness of FCCNI. They are divided into two categories, non-EA-based and EA-based algorithms. Comprising the non-EA-based algorithms, SLPA is based on dynamic label propagation [56], OCDDP is based on peak density [57], SPLIT is under the ego-splitting framework [58], and NI-LPA is the node importance-based label algorithm. The two multi-objective evolutionary algorithms are MOEA-SAov [43] and CEMOV [59]. The parameter $r$ of SLPA changes from 0.1 to 0.45 in a step of 0.05, and the training resolution of SPLIT changes from 0.5 to 10. The other parameters are set according to the suggestions of the papers on which they are based. For a fair comparison, the number of individuals in a population is set to 102 and the maximum number of evolutionary generations is set to 50 in the EA-based algorithms. In the following, "-" means that the algorithm failed return the results of the network in an acceptable time.

### 4.3. Evaluation metrics

For the real networks, the community structure is expected to be obvious and nodes in the communities are expected to be as dense and homogeneous as possible. So two evaluation indicators, namely density $D$ and entropy $E$ [43], are used to jointly evaluate the performance. $D$ is defined as follows.

$$D = \sum_{k=1}^{|C|} \frac{e_k}{e} \tag{8}$$

where $|C|$ is the number of communities, $e_k$ and $e$ respectively represent the number of edges in community $C_k$ and the network. $D$ measures the ratio of edges within the communities to the edges in the entire network. If the intra-connections of communities are denser and the connections between communities are sparser, the $D$ value is greater. $E$ is defined as follows.

$$E = \sum_{k=1}^{|C|} \frac{n_k}{n} \cdot entropy(C_k) \tag{9}$$

$$entropy(C_k) = -\sum_{a \in C_k} p_a \cdot log(p_a) \tag{10}$$

where $n_k$ and $n$ respectively represent the number of nodes in community $C_k$ and in the network. $p_a$ is the percentage of nodes with attribute value $a$ in community $C_k$. Smaller entropy values indicate stronger homogeneity of the nodes in the communities. On the contrary, larger entropy indicates weaker homogeneity of the nodes in the communities. Thus, larger density $D$ and smaller entropy $E$ correspond to the division of the nodes into the communities with denser intra-connections and higher homogeneity. That is, theoretically, it is more desirable to obtain community division results with larger density $D$ and smaller entropy $E$.

In addition, to further verify the performance of FCCNI, generalized normalized mutual information (*gNMI*) [50] is used to evaluate the consistency quality between the real overlapping communities and the detected ones on networks with known distributions. For real partition $A$ and detected partition $B$, *gNMI* is defined as follows.

$$gNMI(A,B) = \frac{-2\sum_{i=1}^{N_A}\sum_{j=1}^{N_B} C_{ij}\log(C_{ij}N/C_{i\cdot}C_{\cdot j})}{\sum_{i=1}^{N_A} C_{i\cdot}\log(C_{i\cdot}/N) + \sum_{j=1}^{N_B} C_{\cdot j}\log(C_{\cdot j}/N)} \tag{11}$$

$N_A$ and $N_B$ respectively represent the number of communities in real partition $A$ and in detected partition $B$ by the algorithm, and $N$ is the number of nodes in the network. $C_{ij}$ is the number of nodes both in community $i$ of $A$ and community $j$ of $B$. $C_{i\cdot}(C_{\cdot j})$ is the sum of elements in row $i$ (column $j$) in $C$. *gNMI* $\in$ [0, 1]. Generally speaking, the larger the *gNMI* value, the more accurate the community detection of the algorithm. Only when the community divisions of the algorithm are the same as the real distributions *gNMI*=1.

For the non-dominated front, the Inverse Generational Distance (*IGD*) [60] is used for evaluation. *IGD* mainly evaluates the convergence and distribution performance of the algorithm by measuring the minimum distance between the individuals obtained by the algorithm and the individuals on the real non-dominated front. *IGD* is defined as follows.

$$IGD(T,S) = \frac{\sum_{v \in T} d(v,S)}{|T|} \tag{12}$$

where $T$ is the set of nodes evenly distributed on the real non-dominated front, and the number of corresponding nodes is given by $|T|$. In practical applications, the real non-dominated front is also the non-dominated front generated by the non-dominated-sort on several non-dominated fronts. $S$ is the obtained optimal non-dominated solution set by the algorithm. $d(v, S)$ is the minimum Euclidean distance from individual $v$ in $T$ to population $S$. The convergence performance is good when $d(v, S)$ is relatively small.

On the contrary, when the distribution performance is very poor, most individuals in the population are concentrated in a narrow area and the $d(v, S)$ of multiple individuals will be large. Thus the smaller *IGD* means the better comprehensive performance of the algorithm, including both convergence and distribution.

For synthetic networks with real community structure, $f_1{-}score$ is used to evaluate the ability of the algorithm to detect overlapping nodes. The metrics comprehensively weigh the harmonic value of *recall* and *precision*[61]. The definitions are as follows.

$$recall = \frac{|O_A \cap O_T|}{O_T} \tag{13}$$

$$precision = \frac{|O_A \cap O_T|}{|O_A|} \tag{14}$$

$$f_1{-}score = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{15}$$

where $O_A$ and $O_T$ respectively indicate the detected overlapping node set and the real overlapping node set, and $|\cdot|$ represents the corresponding number. It can be seen that *recall* represents the ratio of the correct overlapping nodes detected by the algorithms to the real overlapping nodes. *precision* reveals the accuracy of detecting overlapping nodes. $f_1{-}score$ combines the two indicators of *recall* and *precision*, and assigns equal importance to these two indicators. It represents the overall performance of the algorithm. Its value is 1 only when the detected overlapping nodes are completely correct.

### 4.4. Discussion of the parameter

In FCCNI, an adjustable parameter is introduced to control the proportion of community fitness and attribute similarity. It plays an important role in judging whether each node joins another community as an overlapping node. To explore the difference of community detection results with the variation of $\lambda$, different experiments were done in all real-life datasets. For all the networks, $\lambda$ ranges from 0 to 1.0 with the step of 0.1. The final results are shown in Fig. 5.

In Fig. 5, The *gNMI* is calculated by (11) and the maximum *gNMI* is reported as the *gNMI_max* over 20 runs. The average value of 20 times is reported as *gNMI_avg*. (a) and (b) respectively draw the *gNMI_max* and *gNMI_avg* of the community division on real-life networks with different $\lambda$ values. As can be seen from the two figures, on the two datasets of karate and dolphins, the *gNMI_max* stays unchanged with the parameter $\lambda$ changing from 0 to 0.8. When the parameter $\lambda$ changes from 0.1 to 0.5, the *gNMI_avg* value changes slightly. However, when $\lambda$ is 0.5, the network of dolphins achieves the best result of *gNMI_avg*. It can be seen that when $\lambda$ is greater than 0.5, *gNMI_max* and *gNMI_avg* are gradually decreasing on two networks of polbooks and football, which means the difference between the detected communities and the real one is larger and larger. When $\lambda$ is 0.5, *gNMI_max* achieves the best on football. When $\lambda$ is 0.0, both *gNMI_max* and *gNMI_avg* performance the best on polbooks. The *gNMI_max* does not change much on politics_uk, however, the *gNMI_avg* achieves the best when $\lambda$ is 0.5.

In order to explore the changes of $D$ and $E$ values corresponding to the community divisions of the real-life networks. The values are presented when the maximum *gNMI* value is obtained with different $\lambda$. The results are shown in Table 5.

As can be seen from Table 5, on the network of dolphins, the changes of $\lambda$ have no effect on the $D$ and $E$ values under the maximum *gNMI* except for $\lambda$ is 1.0. For the two networks of polbooks and football, when $\lambda$ is 0.5, the $E$ value is the smallest. However, the $D$ value is the largest when $\lambda$ is 0.2 on the polbooks, and when $\lambda$ is 1.0, the $D$ value is the largest in football. While on the network of politics_uk, the $D$ achieves best when $\lambda$ is 0.3 and the $E$ achieves best when $\lambda$ is 0.9. In this experiment, a phenomenon can be found that the maximum *gNMI* and the maximum $D$ of the same network are sometimes not in a one-to-one correspondence. In addition, the optimal $D$ and the optimal $E$ are not in a one-to-one correspondence. To some extent, it shows that there are still some limitations to evaluate algorithms only by density $D$ or entropy $E$. It is precisely because of this that the subsequent experiments are conducted based on the maximum *gNMI_avg*. Since the *gNMI* measures the similarity between the detected communities and the real clusters. Therefore, the adjustable parameter $\lambda$ is set to 0.5 on the four datasets of karate, dolphins, football, and politics_uk. $\lambda$ is set to 0.0 for polbooks. For fair comparison, $\lambda$ is set to 0.5 on the four synthetic networks since most of the networks perform the best with $\lambda$ = 0.5.

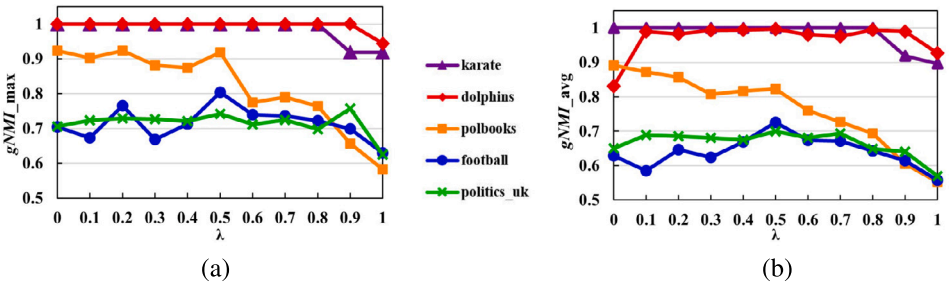


**Fig. 5.** *gNMI_max* and *gNMI_avg* of FCCNI with different $\lambda$ in real-life networks. (a) *gNMI_max*. (b) *gNMI_avg*.

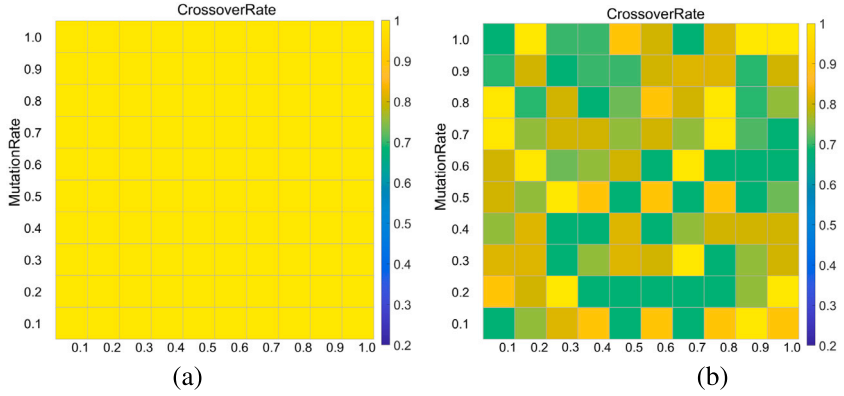


**Fig. 6.** *gNMI* values of FCCNI with different probabilities of crossover and mutation operators. (a) Karate network. (b) Dolphin network.

**Table 5**
*D* and *E* values under the *gNMI_max* obtained by the FCCNI algorithm.

| Metrics | $\lambda$ | karate | dolphins | polbooks | football | politics_uk |
|---|---|---|---|---|---|---|
| | 0.0 | 0.8718 | **0.9623** | 0.0159 | 0.7366 | 0.0066 |
| | 0.1 | 0.8718 | **0.9623** | 0.0068 | 0.6943 | 0.0056 |
| | 0.2 | 0.8718 | **0.9623** | **2.1610** | 0.4943 | 0.0165 |
| | 0.3 | 0.8718 | **0.9623** | 1.5147 | 0.7236 | **0.8605** |
| | 0.4 | 0.8718 | **0.9623** | 1.1927 | 0.7325 | 0.0037 |
| *D* | 0.5 | 0.8718 | **0.9623** | 1.7939 | 0.6639 | 0.0152 |
| | 0.6 | 0.8718 | **0.9623** | 1.9093 | 0.4323 | 0.0059 |
| | 0.7 | 0.8718 | **0.9623** | 1.4853 | 0.3840 | 0.0069 |
| | 0.8 | 0.8718 | **0.9623** | 1.4558 | 0.5498 | 0.0347 |
| | 0.9 | **0.8846** | **0.9623** | 1.0771 | 0.5400 | 0.0214 |
| | 1.0 | 0.8205 | **0.9623** | 0.9569 | **0.8000** | 0.3719 |
| | 0.0 | **0** | **0** | 0.5174 | 0.7887 | 1.2062 |
| | 0.1 | **0** | **0** | 0.6453 | 0.9613 | 0.8601 |
| | 0.2 | **0** | **0** | 0.4090 | 0.8029 | 0.7492 |
| | 0.3 | **0** | **0** | 0.1536 | 1.0504 | 0.4262 |
| | 0.4 | **0** | **0** | 0.2466 | 1.1204 | 0.8142 |
| *E* | 0.5 | **0** | **0** | **0.1308** | **0.7305** | 0.7532 |
| | 0.6 | **0** | **0** | 0.4240 | 1.0591 | 0.8866 |
| | 0.7 | **0** | **0** | 0.1982 | 0.8954 | 0.7167 |
| | 0.8 | **0** | **0** | 0.4699 | 0.9357 | 0.3941 |
| | 0.9 | 0.1614 | **0** | 0.3602 | 1.2649 | **0.3628** |
| | 1.0 | 0.1614 | 0.0936 | 0.6318 | 1.2185 | 0.9677 |

In addition, for the probabilities of crossover and mutation operators, a grid search of *gNMI* value is performed on the two datasets of karate and dolphin as in Fig. 6. The change in the color of the square indicates the change of the *gNMI* value.

As shown in Fig. 6, due to the small scale and simple structure of the karate dataset, the probabilities of crossover and mutation operators have no obvious impact on the detection results of its community structure. For the dolphin dataset, the effect of the probabilities of crossover and mutation operators on the results is relatively random. It can also be seen that the crossover and mutation operators in the proposed algorithm are targeting an almost random diversification.

### 4.5. Experimental results and analysis on all networks

Following the above analysis, the experiments are conducted in five real-life networks and four classical synthetic networks. The results are as follows.

#### 4.5.1. Experimental results of the non-dominated front

The non-dominated fronts obtained on all the networks by FCCNI and the other two evolutionary algorithms, called MOEA-SAov and CEMOV, are shown in Fig. 7.

Since CEMOV cannot return the results on LFR3 in an acceptable time, the result of CEMOV on LFR3 is not presented. The relationship between $EQ$ and $A_s$ in the obtained non-dominated solutions can be seen in Fig. 7. In the nine networks, $A_s$ shows a downward trend with the increase of $EQ$, demonstrating that the two measurement indicators restrict each other. The $EQ$ aims to gather nodes that are dense in topology, while the $A_s$ gathers nodes with the same attribute values. When the modularity is between 0.3 and 0.7, it indicates a good community division of the method. On the three networks of karate, dolphins and polbooks, FCCNI can obtain non-dominated solutions with modularity much larger than 0.3.

The extended modularity of the non-dominated solutions obtained by MOEA-SAov and CEMOV on the three networks of karate, dolphins and polbooks is much less than FCCNI. It can be seen that FCCNI can detect communities with higher quality on these three networks. On the politics_uk network with complex connections, MOEA-SAov and CEMOV can hardly obtain a non-dominated solution with an $EQ$ value larger than 0.3, but FCCNI can still effectively detect the communities. At the same time, compared with MOEA-SAov and CEMOV, FCCNI can obtain non-dominated solutions with higher attribute similarity under the same modularity on the real networks of dolphins, karate, polbooks, football and all the synthetic networks.

The convergence performance and distribution performance of FCCNI is further evaluated by inverse generational distance evaluation

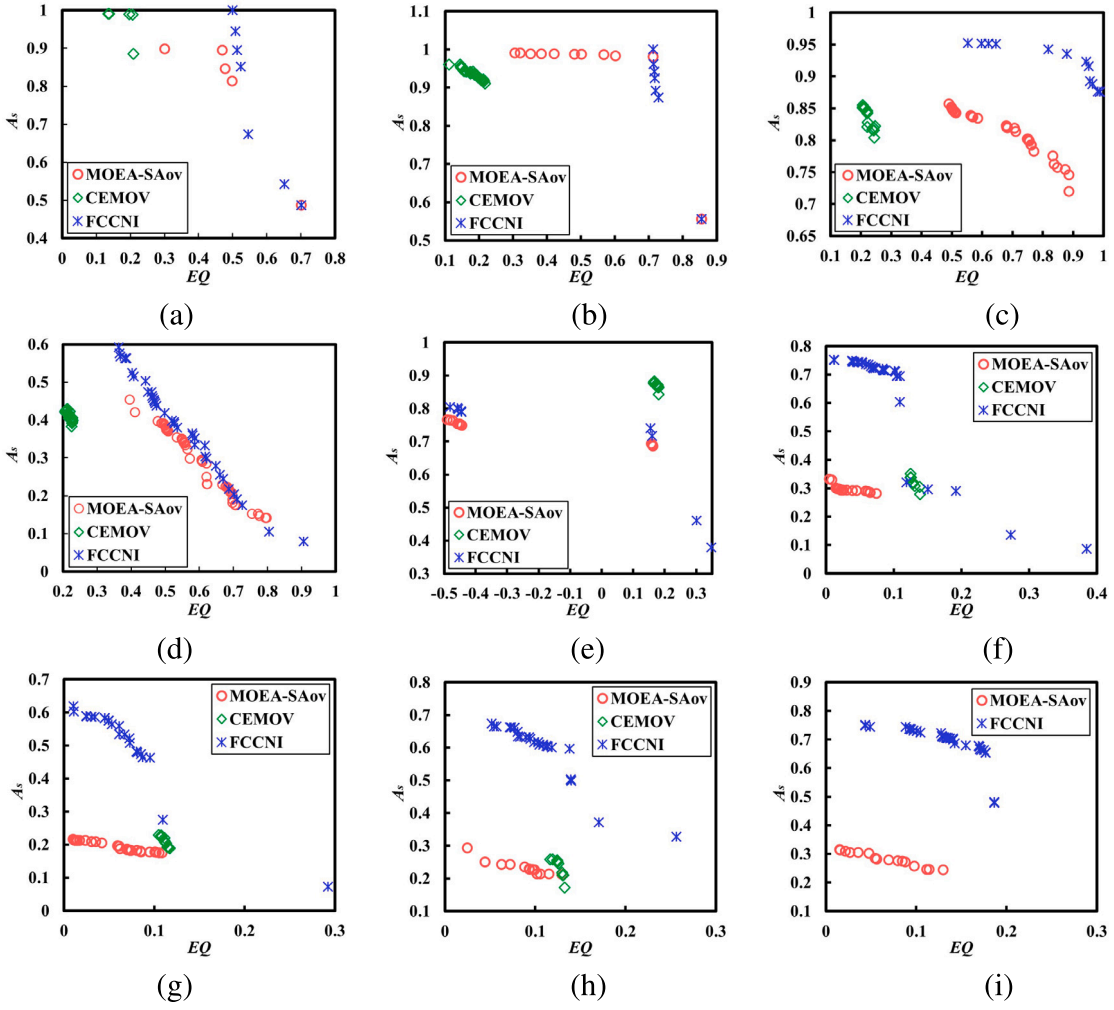


**Fig. 7.** Non-dominated fronts obtained by MOEA-SAov, CEMOV and FCCNI on nine networks. (a) karate. (b) dolphins. (c) polbooks. (d) football. (e) politics uk. (f) LFR1. (g) LFR2. (h) LFR3. (i) LFR4.

**Table 6**
*IGD* values of MOEA-SAov, CEMOV and FCCNI on nine networks.

| Algorithms | Metric | karate | dolphins | polbooks | football | politics_uk | LFR0 | LFR1 | LFR2 | LFR3 |
|---|---|---|---|---|---|---|---|---|---|---|
| MOEA-SAov | *IGD_min* | 0.0604 | 0.0415 | 0.1517 | 0.0343 | 0.2091 | 0.3163 | 0.2190 | 0.3191 | 0.3690 |
| | *IGD_avg* | 0.0856 | 0.0684 | 0.1765 | 0.0568 | 0.2168 | 0.3368 | 0.2367 | 0.3394 | 0.3874 |
| | std | 0.0563 | 0.0495 | 0.0368 | 0.0723 | 0.0358 | 0.0413 | 0.0297 | 0.0320 | 0.0438 |
| CEMOV | *IGD_min* | 0.4305 | 0.5602 | 0.5889 | 0.4001 | **0.0992** | 0.2823 | 0.2004 | 0.3406 | – |
| | *IGD_avg* | 0.4687 | 0.5602 | 0.6037 | 0.4133 | 0.1025 | 0.3068 | 0.2367 | 0.3368 | – |
| | std | 0.0468 | 0 | 0.0368 | 0.0279 | 0.0269 | 0.0369 | 0.0569 | 0.0316 | – |
| FCCNI | *IGD_min* | **0.0070** | **0** | **0** | **0.0065** | 0.1040 | **0.0032** | **0.0247** | **0** | **0** |
| | *IGD_avg* | 0.008 | **0** | **0** | 0.0069 | 0.1096 | 0.0076 | 0.0316 | **0** | **0** |
| | std | 0.0137 | **0** | **0** | 0.0213 | 0.0185 | 0.0134 | 0.0132 | **0** | **0** |

index (*IGD*). The *IGD* values of the three EA-based algorithms MOEA-SAov, CEMOV and FCCNI are shown in Table 6. Each algorithm was applied to each dataset 10 times, and the maximum value of the 10 independent runs was recorded, and also, the average value and the standard deviation of the 10 independent runs were considered as *IGD_avg* and std.

It can be seen from Table 6 that the *IGD* of FCCNI on the four real-life networks are smaller than MOEA-SAov. The non-dominated solutions of FCCNI perform better in terms of convergence and distribution on the four networks of karate, dolphins, polbooks and football. Although CEMOV performs best on the network of politics_uk, there is not much difference between the *IGD* of FCCNI and it. Specially, on the four networks of dolphins, polbooks, LFR2 and LFR3, the *IGD* value of FCCNI is 0, which indicates that the non-dominated level of all individuals in the non-dominated front obtained by FCCNI is in the first level in the comparison with MOEA-SAov and CEMOV. It also means that the non-dominated solutions of FCCNI completely dominate the non-dominated solutions of MOEA-SAov and CEMOV in the four networks. Moreover, the standard deviation results of FCCNI are also better than those of the other two algorithms.

#### 4.5.2. Experimental results of D and E values

To explore the change of the number of overlapping communities obtained by FCCNI and the relationship between density *D* and entropy *E* in each network. FCCNI does not need to set the number of communities in advance, and can automatically determine the number of communities within a certain range and then get the number reconfirmed or recorrected, which is beneficial for decision makers to

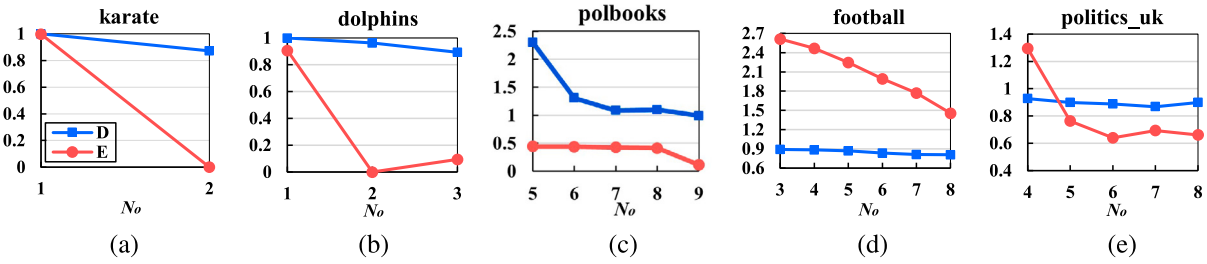


**Fig. 8.** $D$ and $E$ values of FCCNI with different $N_o$ on real-life network.

**Table 7**
$D$ and $E$ values of the algorithms on real-life network.

| Networks | Metrics | SLPA | OCDDP | SPLIT | NI-LPA | MOEA-SAov | CEMOV | FCCNI |
|---|---|---|---|---|---|---|---|---|
| karate | $D$ | 0.8718 | 0.7436 | **0.9615** | 0.9021 | 0.8590 | 0.8846 | 0.8718 |
| | $E$ | 0.3333 | 0.4725 | 0.9133 | 0.4216 | 0.4327 | 0.3367 | **0.0000** |
| dolphins | $D$ | 0.9120 | 0.7987 | **0.9937** | 0.9686 | 0.9623 | 0.9623 | 0.9623 |
| | $E$ | 0.0668 | 0.3454 | 1.2227 | 0.2563 | 0.0936 | 0.3266 | **0.0000** |
| polbooks | $D$ | 0.9560 | 0.9297 | 0.9524 | 0.9501 | 0.9560 | 0.9547 | **0.9932** |
| | $E$ | 0.6921 | 0.3409 | 0.8031 | 0.6710 | 0.6314 | 0.3001 | **0.1151** |
| football | $D$ | 0.7912 | 0.7341 | **0.9674** | 0.7520 | 0.7945 | 0.7928 | 0.7945 |
| | $E$ | 1.4818 | 2.3605 | 9.6385 | 2.4856 | 1.7608 | 2.3449 | **1.3438** |
| politics_uk | $D$ | 0.8708 | 0.8305 | 0.8324 | 0.8548 | 0.8770 | 0.8706 | **0.8883** |
| | $E$ | 0.8081 | **0.1713** | 0.2058 | 0.4125 | 0.6434 | 0.2191 | 0.6401 |

choose. The relationships between $D$ and $E$ under different numbers of overlapping community distributions by FCCNI in the real-life network are shown in Fig. 8, where $N_o$ is the number of overlapping community.

It can be observed that $D$ and $E$ values have similar trend. When the $D$ value is larger, the corresponding $E$ value is also larger. It means the intra-connections of the community is relatively dense, while the degree of homogeneity within the community is relatively low, and vice versa. This also explains that there are contradictions between two different kinds of information from attribute and structure in these datasets to a certain extent. Good results cannot be achieved at the same time. And with the increase of the number of communities, $D$ and $E$ almost show a trend of decreasing. The community structure is less and less obvious at this time. However, the overall attribute homogeneity of the community is increasing, which also enables decision-makers to flexibly choose their own solutions according to actual needs.

To further study the relationship between density $D$ and entropy $E$ in the communities obtained by FCCNI, it is compared with other algorithms. According to the analysis of Fig. 8, the change range of entropy $E$ is relatively larger than that of density $D$. So the solution of $D$ value densest to the other algorithms is selected from the divisions for comparison and analysis. The detailed $D$ and $E$ values of the algorithms on real-life network are shown in Table 7.

For each network in Table 7, the optimal values of $D$ and $E$ are shown in bold. The non-EA-based algorithms tend to detect fixed communities. It can be seen that SPLIT gets communities with higher $D$. Since the EA-based algorithm gets a series of non-dominant solutions after each run. In this series of solutions, the solution corresponding to the largest D value may not be the solution corresponding to the minimum E value. Therefore, by drawing on the practice in [43], the solutions with the closest D or E values compared with other EA-based algorithms should be selected for comparison to ensure fairness. In our experiments, the solutions with the closest D values with other EA-based algorithms are selected for comparison. On the five real-life networks, although the $D$ values are generally similar, the difference between the corresponding $E$ values is relatively obvious. FCCNI achieves best in $E$ on four networks of karate, dolphins, polbooks and football. It indicates that when all the algorithms detect communities with high denseness, the degree of attribute homogeneity in the communities detected by FCCNI is much higher than that of the other algorithms. Especially when the detected communities of FCCNI have the same $D$ value as MOEA-SAov or SLPA on the three real-life networks of karate, dolphins and football, it achieves a better $E$ value. This further shows that FCCNI can detect communities with higher attribute similarity.

#### 4.5.3. Comparison of all algorithms in terms of maximum EQ and $A_s$

In order to evaluate the effectiveness of FCCNI, the maximum $EQ$ and $A_s$ achieved by the seven algorithms are compared. The maximum $EQ$ values of all the algorithms are shown in Fig. 9.

It can be seen from Fig. 9 that on the real-life network, FCCNI shows advantages to a certain extent, which means the divided community structure of FCCNI is more obvious and the intra-connections within the communities are denser. Especially on the network of polbooks, FCCNI obtains the maximum $EQ$ and performs best among the five real-life networks. Although FCCNI does not perform the best in LFR2 and LFR3, it still performs better than most of the algorithms. FCCNI is verified to be effective in obtaining communities with dense intra-connections in most of the networks. Meanwhile, the maximum $A_s$ is further explored and plotted in Fig. 10.

As shown in Fig. 10, FCCNI achieves the best $A_s$ in three real-life networks and all the synthetic networks, which means the homogeneity of attribute in the detected communities is relatively high. Although FCCNI does not perform the best on two networks of LFR2 and LFR3 in terms of $EQ$ in Fig. 9, it can be seen that FCCNI performs much better in $A_s$. FCCNI has advantages in detecting communities with higher degree of homogeneity.

#### 4.5.4. Comparison of all algorithms in terms of gNMI

Although the aforementioned indicators can reflect the denseness of the communities and the homogeneity of attribute to a certain extent, the similarity between the detected communities and the real communities of the network cannot be reflected. To explore the community divisions with real divisions, the *gNMI* value on nine datasets is analyzed in this paper, and meanwhile compared with the other six algorithms. Each algorithm was applied to each dataset 10 times, and the maximum value of the 10 independent runs was recorded, and also, the average value and the standard deviation of the 10 independent runs were considered as *gNMI_avg* and *std*. The *gNMI* values of all the algorithms on nine networks are shown in Table 8, in which "-" means that the algorithm cannot return the results of the network in an acceptable time.

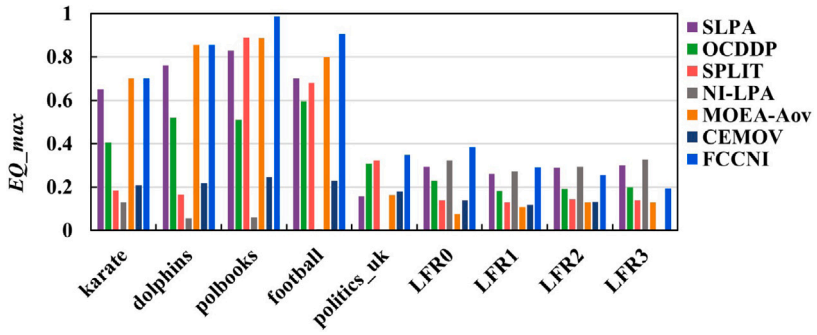


**Fig. 9.** The maximum $EQ$ of seven algorithms on nine networks.

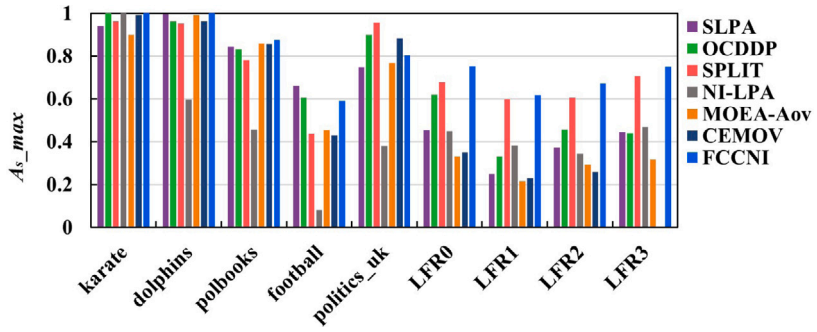


**Fig. 10.** The maximum $A_s$ of seven algorithms on nine networks.

**Table 8**
$gNMI$ values of the algorithms on nine networks.

| Networks | Metrics | SLPA | OCDDP | SPLIT | NI-LPA | MOEA-SAov | CEMOV | FCCNI |
|---|---|---|---|---|---|---|---|---|
| | *gNMI_max* | 0.9183 | 0.4715 | 0.3485 | 0.7091 | 0.9186 | 0.8368 | **1.0000** |
| karate | *gNMI_avg* | 0.9053 | 0.4715 | 0.3485 | 0.7091 | 0.9082 | 0.5786 | **0.9673** |
| | std | 0.1905 | **0** | **0** | **0** | 0.0196 | 0.1215 | 0.0603 |
| | *gNMI_max* | **1.0000** | 0.4675 | 0.1977 | 0.4445 | 0.9445 | 0.4232 | **1.0000** |
| dolphins | *gNMI_avg* | 0.8471 | 0.4675 | 0.1977 | 0.4445 | 0.9445 | 0.3545 | **0.9963** |
| | std | 0.2095 | **0** | **0** | **0** | **0** | 0.0423 | 0.0143 |
| | *gNMI_max* | 0.5057 | 0.3460 | 0.2437 | 0.4778 | 0.4713 | 0.5000 | **0.9234** |
| polbooks | *gNMI_avg* | 0.4585 | 0.3460 | 0.2437 | 0.4778 | 0.3934 | 0.4402 | **0.8906** |
| | std | 0.0608 | **0** | **0** | **0** | 0.0672 | 0.0803 | 0.0266 |
| | *gNMI_max* | 0.7846 | 0.7144 | 0.3743 | 0.5000 | 0.7473 | 0.5000 | **0.8041** |
| football | *gNMI_avg* | 0.6732 | 0.7144 | 0.3743 | 0.5000 | 0.6901 | 0.4689 | **0.7255** |
| | std | 0.0813 | **0** | **0** | **0** | 0.0702 | 0.0646 | 0.0469 |
| | *gNMI_max* | 0.7051 | 0.6974 | 0.4395 | 0.5000 | 0.7020 | 0.5980 | **0.7120** |
| politics_uk | *gNMI_avg* | 0.5623 | 0.6974 | 0.4395 | 0.5000 | 0.6187 | 0.5121 | **0.6988** |
| | std | 0.0888 | **0** | **0** | **0** | 0.0590 | 0.0481 | 0.0361 |
| | *gNMI_max* | 0.2649 | 0.1796 | 0.1157 | 0.3604 | 0.2630 | 0.3165 | **0.5520** |
| LFR0 | *gNMI_avg* | 0.2349 | 0.1796 | 0.1157 | 0.3604 | 0.2480 | 0.3076 | **0.5286** |
| | std | 0.0213 | **0** | **0** | **0** | 0.0144 | 0.0212 | 0.0147 |
| | *gNMI_max* | 0.1899 | 0.0231 | 0.0717 | 0.3116 | 0.2220 | 0.3133 | **0.5546** |
| LFR1 | *gNMI_avg* | 0.1424 | 0.0231 | 0.0717 | 0.3116 | 0.2120 | 0.3063 | **0.5256** |
| | std | 0.0340 | **0** | **0** | **0** | 0.0134 | 0.0153 | 0.0214 |
| | *gNMI_max* | 0.2414 | 0.0524 | 0.0891 | 0.3266 | 0.1701 | 0.3004 | **0.4280** |
| LFR2 | *gNMI_avg* | 0.2349 | 0.0524 | 0.0891 | 0.3266 | 0.1478 | 0.2983 | **0.4035** |
| | std | 0.0306 | **0** | **0** | **0** | 0.0159 | 0.0193 | 0.0155 |
| | *gNMI_max* | 0.2311 | 0.0276 | 0.0781 | **0.4162** | 0.1525 | – | 0.2231 |
| LFR3 | *gNMI_avg* | 0.2085 | 0.0276 | 0.0781 | **0.4162** | 0.1346 | – | 0.2054 |
| | std | 0.0116 | **0** | **0** | **0** | 0.0184 | – | 0.0129 |

*gNMI_max* and *gNMI_avg* are respectively shown in Table 8, with the optimal value on each dataset marked in bold. It can be seen that the *gNMI_max* obtained by FCCNI is significantly higher than that of the other six comparison algorithms on five real-life networks and three synthetic networks. It shows strong superiority in terms of *gNMI* value in these networks. Although SLPA achieves the same maximum value of *gNMI* as FCCNI on the network of dolphins, FCCNI achieves a better average *gNMI* value. Especially on the network of polbooks, the results of FCCNI are the most significantly improved over the comparison algorithms. In addition, the maximum *gNMI* of FCCNI on five real-life networks is larger than 0.7, which shows that the FCCNI obtains community divisions more similar to the real community. In general, FCCNI has better performance and can obtain communities that are more similar to the real communities. In the four synthetic networks, although FCCNI does not achieve the best in LFR3, it performs much better in the other three synthetic networks. Besides, although FCCNI does not have much advantage over standard deviation since some algorithms tend to find certain communities, FCCNI performs better

**Table 9**
Results of Wilcoxon Signed Rank Test on *gNMI* of all the algorithms.

| *gNMI* | FCCNI/SLPA | | FCCNI/OCDDP | | FCCNI/SPLIT | | FCCNI/NI-LPA | | FCCNI/MOEA-SAov | | FCCNI/CEMOV | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *p* | *h* | *p* | *h* | *p* | *h* | *p* | *h* | *p* | *h* | *p* | *h* |
| karate | 0.000062 | 1 | 0.000016 | 1 | 0.000016 | 1 | 0.000016 | 1 | 0.000033 | 1 | 0.000064 | 1 |
| dolphins | 0.000377 | 1 | 0.000024 | 1 | 0.000024 | 1 | 0.000024 | 1 | 0.000097 | 1 | 0.000087 | 1 |
| polbooks | 0.000183 | 1 | 0.000064 | 1 | 0.000064 | 1 | 0.000064 | 1 | 0.000168 | 1 | 0.000183 | 1 |
| football | 0.064022 | 0 | 0.115258 | 0 | 0.000064 | 1 | 0.000064 | 1 | 0.002202 | 1 | 0.000183 | 1 |
| politics_uk | 0.005777 | 1 | 0.115258 | 0 | 0.000022 | 1 | 0.000064 | 1 | 0.02094 | 1 | 0.01133 | 1 |
| LFR0 | 0.000183 | 1 | 0.000666 | 1 | 0.000064 | 1 | 0.000064 | 1 | 0.000121 | 1 | 0.000178 | 1 |
| LFR1 | 0.000183 | 1 | 0.000022 | 1 | 0.000064 | 1 | 0.000064 | 1 | 0.000183 | 1 | 0.000183 | 1 |
| LFR2 | 0.000183 | 1 | 0.000046 | 1 | 0.000064 | 1 | 0.000064 | 1 | 0.000183 | 1 | 0.000183 | 1 |
| LFR3 | 0.73373 | 0 | 0.000022 | 1 | 0.000064 | 1 | 0.000064 | 1 | 0.000183 | 1 | - | - |

than the other algorithms such as SLPA, MOEA-SAov and CEMOV on most of the networks.

The Wilcoxon Signed Rank Test is then used to statistically analyze the results obtained by FCCNI and the other six compared algorithms. The results shown in Table 9 are obtained by running all the algorithms 10 times, in which "-" means that the algorithm cannot return the results of the network in an acceptable time. In Table 9, $p$ represents the probability that the medians of the two samples are equal. $h$ shows the difference between the medians of the samples. When $h = 0$, it means that the difference is not so significant. On the contrary, $h = 1$ means that the difference is significant. In the results of *gNMI* shown in Table 9, FCNNI can obtain the highest values on most networks, and the results of Wilcoxon Signed Rank Test show that most of the differences between FCNNI and the compared algorithms are obvious.

### 4.5.5. Comparison of all algorithms in terms of $f_1-score$

To further verify the performance of FCCNI, the indicator $f_1-score$ is mainly analyzed to measure the detection of overlapping nodes. The specific results of $f_1-score$ are shown in Table 10. Each algorithm was applied to each dataset 10 times, and the maximum value of the 10 independent runs was recorded. Additionally, the average value and the standard deviation of the 10 independent runs were considered as $f_1-score\_avg$ and std.

From the results shown in Table 10, it can be seen that FCCNI improves $f_1-score$ on four classical networks, which means that our algorithm is more effective in finding real overlapping communities. Among them, the *recall* value of FCCNI on these four synthetic networks can reach more than 0.7 with the best improvement effect. It also means that the correct overlapping nodes detected by FCCNI account for a higher proportion of the real overlapping nodes. Although SLPA achieves higher *precision* due to the fewer detected overlapping nodes, the overlapping nodes correctly detected by FCCNI are more, thus its overall performance $f_1-score$ is improved by better balancing the two one-sided indicators. In addition, compared with LFR0, the fuzziness of community structure increases in LFR1. The overall ability to find overlapping nodes is slightly reduced. Compared with LFR0, the number of communities to which overlapping nodes belong increases in LFR2, and the effect of finding real overlapping communities is also slightly reduced. Compared with LFR0, LFR3 has the same proportion of overlapping nodes, but the total number of community nodes increases. It can be seen that the values of *recall*, *precision* and $f_1-score$ obtained by FCCNI are reduced. However, FCCNI is still more accurate than the comparison algorithms in these cases. In addition, it can still be seen that although FCCNI does not have much advantage over standard deviation since some algorithms tend to find certain communities, FCCNI performs better than some other algorithms.

**Table 10**
Experimental results of the algorithms on $f_1-score$.

| Algorithms | Metrics | LFR0 | LFR1 | LFR2 | LFR3 |
|---|---|---|---|---|---|
| SLPA | *recall* | 0.4361 | 0.4920 | 0.5642 | 0.4685 |
| | *precision* | 0.3427 | 0.3368 | 0.3661 | 0.3824 |
| | $f_1-score\_max$ | 0.3837 | 0.3999 | 0.4441 | 0.4211 |
| | $f_1-score\_avg$ | 0.3765 | 0.3854 | 0.4365 | 0.4037 |
| | std | 0.0278 | 0.0375 | 0.0247 | 0.0347 |
| OCDDP | *recall* | 0.4333 | 0.1300 | 0.1500 | 0.1240 |
| | *precision* | 0.2462 | 0.2143 | 0.3147 | 0.2394 |
| | $f_1-score\_max$ | 0.4606 | 0.1618 | 0.2032 | 0.1634 |
| | $f_1-score\_avg$ | 0.4606 | 0.1618 | 0.2032 | 0.1634 |
| | std | **0** | **0** | **0** | **0** |
| SPLIT | *recall* | 0.4233 | 0.4467 | 0.4267 | 0.4493 |
| | *precision* | 0.1406 | 0.1427 | 0.1488 | 0.1495 |
| | $f_1-score\_max$ | 0.2111 | 0.2163 | 0.2163 | 0.2243 |
| | $f_1-score\_avg$ | 0.2111 | 0.2163 | 0.2163 | 0.2243 |
| | std | **0** | **0** | **0** | **0** |
| NI-LPA | *recall* | 0.8156 | 0.8378 | 0.7968 | 0.7685 |
| | *precision* | 0.2937 | 0.3107 | 0.3178 | 0.2867 |
| | $f_1-score\_max$ | 0.4318 | 0.4532 | 0.4543 | 0.4176 |
| | $f_1-score\_avg$ | 0.4318 | 0.4532 | 0.4543 | 0.4176 |
| | std | **0** | **0** | **0** | **0** |
| MOEA-SAov | *recall* | 0.7600 | 0.6967 | 0.7200 | 0.7276 |
| | *precision* | 0.3304 | 0.3120 | 0.3224 | 0.3180 |
| | $f_1-score\_max$ | 0.4606 | 0.4309 | 0.4454 | 0.4423 |
| | $f_1-score\_avg$ | 0.4467 | 0.4247 | 0.4361 | 0.4268 |
| | std | 0.0357 | 0.0467 | 0.0286 | 0.0394 |
| CEMOV | *recall* | 0.8300 | 0.8233 | 0.8033 | – |
| | *precision* | 0.3136 | 0.3001 | 0.3118 | – |
| | $f_1-score\_max$ | 0.4552 | 0.4399 | 0.4492 | – |
| | $f_1-score\_avg$ | 0.4437 | 0.4276 | 0.4267 | – |
| | std | 0.0216 | 0.0198 | 0.0276 | – |
| FCCNI | *recall* | 0.8833 | 0.8000 | 0.8667 | 0.7861 |
| | *precision* | 0.3232 | 0.3236 | 0.3175 | 0.3186 |
| | $f_1-score\_max$ | **0.4773** | **0.4607** | **0.4647** | **0.4534** |
| | $f_1-score\_avg$ | **0.4687** | **0.4538** | **0.4498** | **0.4461** |
| | std | 0.0134 | 0.0286 | 0.0367 | 0.0291 |

**Table 11**
The comparison of all algorithms on running time.

| Networks | SLPA | OCDDP | SPLIT | NI-LPA | MOEA-SAov | CEMOV | FCCNI |
|---|---|---|---|---|---|---|---|
| karate | <1s | <1 s | <1 s | <1 s | 1 s | 152 s | 1 s |
| dolphins | <1 s | <1 s | <1 s | <1 s | 4 s | 413 s | 4 s |
| polbooks | <1 s | <1 s | <1 s | <1 s | 22 s | 326 s | 28 s |
| football | <1 s | <1 s | <1 s | <1 s | 16 s | 355 s | 19 s |
| politics_uk | 4 s | <1 s | <1 s | <1 s | 6369 s | 4830 s | 7838 s |
| LFR0 | 4 s | 3 s | 7 s | 2 s | 2342 s | 33274 s | 2840 s |
| LFR1 | 4 s | 3 s | 8 s | 2 s | 2574 s | 34625 s | 3213 s |
| LFR2 | 4 s | 3 s | 7 s | 2 s | 2365 s | 35416 s | 3871 s |
| LFR3 | 24 s | 26 s | 295 s | 19 s | 21513 s | – | 25820 s |

### 4.5.6. Running time

In this section, FCCNI is compared with the four non-EA-based algorithms and the two EA-based algorithms in terms of running time. The running times of these seven methods on different datasets are shown in Table 11.

As can be seen from Table 11, the non-EA-based algorithms tend to detect communities in a few seconds. The label-propagation-based algorithms have low complexity. For EA-based algorithms, the process of community detection takes some time. The proposed FCCNI needs more time than MOEA-SAov in all networks. However, FCCNI needs less time than CEMOV. As the number of nodes increases, the advantages

**Table 12**
The *gNMI* values of CNI and FCCNI on nine datasets.

| *gNMI* | CNI | | FCCNI | |
|---|---|---|---|---|
| | *gNMI_max* | *gNMI_avg(std)* | *gNMI_max* | *gNMI_avg(std)* |
| karate | **1.0000** | **0.9673(0.0603)** | **1.0000** | **0.9673(0.0603)** |
| dolphins | **1.0000** | 0.9681(0.0799) | **1.0000** | **0.9963(0.0143)** |
| polbooks | 0.9222 | 0.8769(0.0416) | **0.9234** | **0.8906(0.0266)** |
| football | 0.7660 | 0.6863(0.0570) | **0.8041** | **0.7255(0.0469)** |
| politics_uk | 0.7055 | 0.6530(0.0741) | **0.7120** | **0.6988(0.0361)** |
| LFR0 | 0.5505 | 0.5206(0.0149) | **0.5520** | **0.5286(0.0147)** |
| LFR1 | 0.5466 | 0.5233(**0.0204**) | **0.5546** | **0.5256**(0.0214) |
| LFR2 | 0.4205 | 0.4010(**0.0147**) | **0.4280** | **0.4035**(0.0155) |
| LFR3 | 0.2192 | 0.2046(0.0142) | **0.2231** | **0.2054(0.0129)** |

of the running time of FCCNI compared with CEMOV become more apparent. Although FCCNI sacrifices limited running time, it achieves a significant improvement in community detection.

#### 4.5.7. More experiments

In MOEAs for community detection, the number of communities usually can be automatically determined during the search process [62]. The number of communities can also be optimized after the initialization phase of the algorithm. This section explores how the community fusion strategy, that reconfirms and corrects the number of communities, affects the performance of FCCNI. The algorithm without the community fusion strategy is named as CNI. The $gNMI$ values of CNI and FCCNI on the nine datasets are shown in Table 12. The best values are shown in bold in Table 12. It can be seen that although on the karate dataset, the results obtained by FCCNI and CNI are the same, the values of $gNMI\_max$ and $gNMI\_avg$ obtained by FCCNI are better than CNI on the other eight datasets. Furthermore, FCCNI achieves the best results in terms of standard deviation on most datasets. In general, the results indicate that the community fusion strategy is indeed beneficial for overlapping community detection in FCCNI.

## 5. Conclusions

An overlapping community detection algorithm, based on fusion of internal and external connectivity and correction of node intimacy (FCCNI), has been proposed in this paper. Node attribute and topological structure are simultaneously utilized to obtain community divisions with high quality. Firstly, some communities with sparse intra-connections but dense inter-connections are integrated in FCCNI. This automatically determines the number of communities, and also reconfirms and effectively corrects the number of communities. Secondly, the intimacy function between a given node and any other node is designed according to their common neighbors and intra connections. The labels of the nodes are corrected in two stages of initialization and an evolution process based on node intimacy. A more accurate non-overlapping community division can be obtained in the initialization stage. Subsequently, the efficiency of correction is improved by exploiting the node intimacy in the evolution process. In the process of obtaining overlapping communities from non-overlapping communities, a formula that considers both the topology and node attribute is designed to judge whether the node belongs to multiple communities. This enables the attribute similarity of each overlapping community, and the overall quality of community division, to be greatly improved.

Five real-life networks and four classical synthetic networks have been used to test the performance of FCCNI. The experimental results demonstrate that FCCNI has significant advantages in discovering overlapping communities and finding valid overlapping nodes. Furthermore, the obtained communities have denser intra-connections and higher homogeneity compared with other methods. However, FCCNI only focuses on single attributed networks, in which all nodes have only one attribute. Future research will consider community detection in continuous attributed networks and multi-attributed networks.

## CRediT authorship contribution statement

**Ronghua Shang:** Writing – review & editing, Conceptualization. **Sa Wang:** Data curation, Methodology, Writing – original draft. **Weitong Zhang:** Software, Writing – review & editing. **Jie Feng:** Conceptualization. **Licheng Jiao:** Conceptualization, Supervision. **Rustam Stolkin:** Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grants Nos. 62176200 and 62271374, the Natural Science Basic Research Program of Shaanxi under Grant Nos. 2022JC-45 and 2022JQ-616, the Open Research Projects of Zhejiang Lab under Grant 2021KG0AB03, the Guangdong Basic and Applied Basic Research Foundation under Grant No. 2021A1515110686, the Postdoctoral Research Project Funding of Shaanxi under Grant 2023BSHTBZZ30, and the Research Project of SongShan Laboratory under Grant YYJC052022004.

## References

[1] S. Wasserman, K. Faust, Social network analysis methods and applications, Contemp. Sociol. 91 (435) (1994).
[2] M. Girvan, M.E. Newman, Community structure in social and biological networks, Proc. Natl Acad. Sci. 99 (12) (2002) 7821–7826.
[3] M.E. Newman, The structure of scientific collaboration networks, Proc. Natl. Acad. Sci. 98 (2) (2001) 404–409.
[4] R. Shang, H. Liu, L. Jiao, A.M.G. Esfahani, Community mining using three closely joint techniques based on community mutual membership and refinement strategy, Appl. Soft Comput. 61 (2017) 1060–1073.
[5] S. Fortunato, Community detection in graphs, Phys. Rep. 486 (3–5) (2010) 75–174.
[6] A. Lancichinetti, S. Fortunato, Community detection algorithms: a comparative analysis, Phys. Rev. E 80 (5) (2009) 056117.
[7] X. Wang, J. Li, L. Yang, H. Mi, Unsupervised learning for community detection in attributed networks based on graph convolutional network, Neurocomputing 456 (2021) 147–155.
[8] M.E. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.
[9] M.E. Newman, Modularity and community structure in networks, Proc. Natl. Acad. Sci. 103 (23) (2006) 8577–8582.
[10] K.I. Lee, H.S. Oh, S.H. Jung, Y.S. Chung, Moving least square-based hybrid genetic algorithm for optimal design of $W$ -band dual-reflector antenna, IEEE Trans. Magn. 55 (6) (2019) 1–4.
[11] M. Gong, B. Fu, L. Jiao, H. Du, Memetic algorithm for community detection in networks, Phys. Rev. E 84 (5) (2011) 056101.

[12] C. Pizzuti, GA-Net: A genetic algorithm for community detection in social networks, in: International Conference on Parallel Problem Solving from Nature, Springer, 2008, pp. 1081–1090.
[13] L. Ma, M. Gong, J. Liu, Q. Cai, L. Jiao, Multi-level learning based memetic algorithm for community detection, Appl. Soft Comput. 19 (2014) 121–133.
[14] X. Zhang, K. Zhou, H. Pan, L. Zhang, X. Zeng, Y. Jin, A network reduction-based multiobjective evolutionary algorithm for community detection in large-scale complex networks, IEEE Trans. Cybern. 50 (2) (2020) 703–716.
[15] U.N. Raghavan, R. Albert, S. Kumara, Near linear time algorithm to detect community structures in large-scale networks, Phys. Rev. E 76 (3) (2007) 036106.
[16] M. Rosvall, C.T. Bergstrom, Maps of random walks on complex networks reveal community structure, Proc. Natl. Acad. Sci. 105 (4) (2008) 1118–1123.
[17] W. Zhang, R. Shang, L. Jiao, Complex network graph embedding method based on shortest path and MOEA/D for community detection, Appl. Soft Comput. 97 (2020) 106764.
[18] J. Zhu, C. Wang, C. Gao, F. Zhang, Z. Wang, X. Li, Community detection in graph: An embedding method, IEEE Trans. Netw. Sci. Eng. 9 (2) (2021) 689–702.
[19] S. Gregory, Finding overlapping communities in networks by label propagation, New J. Phys. 12 (10) (2010) 103018.
[20] L. Huang, G. Wang, Y. Wang, E. Blanzieri, C. Su, Link clustering with extended link similarity and EQ evaluation division, PLoS One 8 (6) (2013) e66005.
[21] K. Nath, S. Roy, S. Nandi, InOvIn: A fuzzy-rough approach for detecting overlapping communities with intrinsic structures in evolving networks, Appl. Soft Comput. 89 (2020) 106096.
[22] X. Wen, W.-N. Chen, Y. Lin, T. Gu, H. Zhang, Y. Li, Y. Yin, J. Zhang, A maximal clique based multiobjective evolutionary algorithm for overlapping community detection, IEEE Trans. Evol. Comput. 21 (3) (2016) 363–377.
[23] L. Zhang, H. Pan, Y. Su, X. Zhang, Y. Niu, A mixed representation-based multiobjective evolutionary algorithm for overlapping community detection, IEEE Trans. Cybern. 47 (9) (2017) 2703–2716.
[24] Y. Tian, S. Yang, X. Zhang, An evolutionary multiobjective optimization based fuzzy method for overlapping community detection, IEEE Trans. Fuzzy Syst. 28 (11) (2019) 2841–2855.
[25] I.B. El Kouni, W. Karoui, L.B. Romdhane, Node importance based label propagation algorithm for overlapping community detection in networks, Expert Syst. Appl. 162 (2020) 113020.
[26] A. Ramesh, G. Srivatsun, Evolutionary algorithm for overlapping community detection using a merged maximal cliques representation scheme, Appl. Soft Comput. 112 (2021) 107746.
[27] H. Ma, H. Yang, K. Zhou, L. Zhang, X. Zhang, A local-to-global scheme-based multi-objective evolutionary algorithm for overlapping community detection on large-scale complex networks, Neural Comput. Appl. 33 (2021) 5135–5149.
[28] U.K. Roy, P.K. Muhuri, S.K. Biswas, NeSiFC: neighbors' similarity-based fuzzy community detection using modified local random walk, IEEE Trans. Cybern. 52 (10) (2021) 10014–10026.
[29] C. Liu, J. Liu, Z. Jiang, A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks, IEEE Trans. Cybern. 44 (12) (2014) 2274–2287.
[30] S. Kelley, M. Goldberg, M. Magdon-Ismail, K. Mertsalov, A. Wallace, Defining and discovering communities in social networks, in: Handbook of Optimization in Complex Networks, Springer, 2012, pp. 139–168.
[31] Y. Zhou, H. Cheng, J.X. Yu, Clustering large attributed graphs: An efficient incremental approach, in: 2010 IEEE International Conference on Data Mining, IEEE, 2010, pp. 689–698.
[32] Y. Ruan, D. Fuhry, S. Parthasarathy, Efficient community detection in large networks using content and links, in: Proceedings of the 22nd International Conference on World Wide Web, 2013, pp. 1089–1098.
[33] Z. Li, J. Liu, K. Wu, A multiobjective evolutionary algorithm based on structural and attribute similarities for community detection in attributed networks, IEEE Trans. Tybern. 48 (7) (2017) 1963–1976.
[34] A. Moayedikia, Multi-objective community detection algorithm with node importance analysis in attributed networks, Appl. Soft Comput. 67 (2018) 434–451.
[35] Z. Xu, Y. Ke, Y. Wang, H. Cheng, J. Cheng, A model-based approach to attributed graph clustering, in: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, 2012, pp. 505–516.
[36] X. Wang, D. Jin, X. Cao, L. Yang, W. Zhang, Semantic community identification in large attribute networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 30,No. 1, 2016.
[37] Y. Li, C. Sha, X. Huang, Y. Zhang, Community detection in attributed graphs: An embedding approach, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, No. 1, 2018.
[38] R. Hong, Y. He, L. Wu, Y. Ge, X. Wu, Deep attributed network embedding by preserving structure and attribute information, IEEE Trans. Syst. Man Cybern. Syst. 51 (3) (2021) 1434–1445.
[39] C. Bothorel, J.D. Cruz, M. Magnani, B. Micenkova, Clustering attributed graphs: models, measures and methods, Netw. Sci. 3 (3) (2015) 408–444.
[40] C.A.C. Coello, G.B. Lamont, D.A. Van Veldhuizen, et al., Evolutionary algorithms for solving multi-objective problems, vol. 5, Springer, 2007.
[41] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[42] J. Sun, W. Zheng, Q. Zhang, Z. Xu, Graph neural network encoding for community detection in attribute networks, IEEE Trans. Cybern. 52 (8) (2021) 7791–7804.
[43] X. Teng, J. Liu, M. Li, Overlapping community detection in directed and undirected attributed networks using a multiobjective evolutionary algorithm, IEEE Trans. Cybern. 51 (2021) 138–150.
[44] A. Reihanian, M.-R. Feizi-Derakhshi, H.S. Aghdasi, An enhanced multi-objective biogeography-based optimization for overlapping community detection in social networks with node attributes, Inform. Sci. 622 (2023) 903–929.
[45] C. He, Y. Zheng, J. Cheng, Y. Tang, G. Chen, H. Liu, Semi-supervised overlapping community detection in attributed graph with graph convolutional autoencoder, Inform. Sci. 608 (2022) 1464–1479.
[46] H. Shen, X. Cheng, K. Cai, M.-B. Hu, Detect overlapping and hierarchical community structure in networks, Physica A 388 (8) (2009) 1706–1712.
[47] M. Tasgin, A. Herdagdelen, H. Bingol, Community detection in complex networks using genetic algorithms CoRR 2005, (3120), 2007.
[48] Y. Park, M. Song, A genetic algorithm for clustering problems, 1989, pp. 568–575.
[49] B. Yan, S. Gregory, Finding missing edges and communities in incomplete networks, J. Phys. A 44 (49) (2011) 495102.
[50] A. Lancichinetti, S. Fortunato, J. Kertész, Detecting the overlapping and hierarchical community structure in complex networks, New J. Phys. 11 (3) (2009) 033015.
[51] A. Lancichinetti, S. Fortunato, F. Radicchi, Benchmark graphs for testing community detection algorithms, Phys. Rev. E 78 (4) (2008) 046110.
[52] W.W. Zachary, An information flow model for conflict and fission in small groups, J. Anthropol. Res. 33 (4) (1977) 452–473.
[53] D. Lusseau, K. Schneider, O.J. Boisseau, P. Haase, E. Slooten, S.M. Dawson, The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations, Behav. Ecol. Sociobiol. 54 (4) (2003) 396–405.
[54] V. Krebs, Books About U.S. Politics, 2004.
[55] D. Greene, P. Cunningham, Producing a unified graph representation from multiple social network views, in: Proceedings of the 5th Annual ACM Web Science Conference, 2013, pp. 118–121.
[56] J. Xie, B.K. Szymanski, X. Liu, SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process, in: 2011 IEEE 11th International Conference on Data Mining Workshops, IEEE, 2011, pp. 344–349.
[57] X. Bai, P. Yang, X. Shi, An overlapping community detection algorithm based on density peaks, Neurocomputing 226 (2017) 7–15.
[58] A. Epasto, S. Lattanzi, R. Paes Leme, Ego-splitting framework: From non-overlapping to overlapping clusters, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 145–154.
[59] W. Zheng, J. Sun, Q. Zhang, Z. Xu, Continuous encoding for overlapping community detection in attributed network, IEEE Trans. Cybern. (2022) http://dx.doi.org/10.1109/TCYB.2022.3155646.
[60] A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, E. Tsang, Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion, in: 2006 IEEE International Conference on Evolutionary Computation, IEEE, 2006, pp. 892–899.
[61] Q. Chen, T.-T. Wu, A method for local community detection by finding maximal-degree nodes, in: 2010 International Conference on Machine Learning and Cybernetics, Vol. 1, IEEE, 2010, pp. 8–13.
[62] C. Pizzuti, Evolutionary computation for community detection in networks: A review, IEEE Trans. Evol. Comput. 22 (3) (2018) 464–483.