A Mixed Representation-Based Multiobjective Evolutionary Algorithm for Overlapping Community Detection

Lei Zhang, Hebin Pan, Yansen Su, Xingyi Zhang, *Member, IEEE*, and Yunyun Niu

***Abstract*—Designing multiobjective evolutionary algorithms (MOEAs) for community detection in complex networks has attracted much attention of researchers recently. However, most of the existing methods focus on addressing the task of nonoverlapping community detection, where each node must belong to one and only one community. In fact, communities are often overlapped with each other in many real-world networks, thus it is necessary to design overlapping community detection algorithms. To this end, this paper proposes a mixed representation-based MOEA (MR-MOEA) for overlapping community detection. In MR-MOEA, a mixed individual representation scheme is proposed to fast encode and decode the overlapping divisions of complex networks. Specifically, this mixed representation consists of two parts: one represents all potential overlapping nodes and the other delegates all nonoverlapping nodes. These two parts evolve together to detect the overlapping communities of networks based on different updating strategies suggested in MR-MOEA. We verify the effectiveness of the proposed algorithm MR-MOEA on ten real-world complex networks and the experimental results demonstrate that MR-MOEA is superior over six representative algorithms for overlapping community detection.**

***Index Terms*—Complex network, individual representation scheme, multiobjective evolutionary algorithm (MOEA), overlapping community detection.**

I. Introduction

IN REAL-WORLD, networks are deployed in many areas to represent various kinds of complex systems, such as social networks [1], biological networks [2], and the Internet networks [3]. By analyzing different networks, researchers have found their potential properties such as the small-world property [4] and the scale-free property [5], which help us to significantly improve the understanding of the complex world. In recent years, another outstanding property of networks has become a hot research topic, that is the community structure [6]–[8]. Community structure refers to that the nodes in the same community have dense connections and have sparse connections between different communities [9]. Thus, detecting community structure in complex networks has attracted much attention of researchers, which is one of the theoretical foundations of social science, network science, biological science, physical science, and so on [10].

Since Girvan and Newman [9] first proposed the problem of community detection in 2002, a large number of scholars have been devoted to designing a variety of methods for nonoverlapping community detection (i.e., each node must belong to one and only one community) [11]–[14]. However, the communities in many real-world networks often overlap to some extent. That is to say, some nodes in the networks may belong to multiple communities since they may have diverse roles in the network. For example, in social networks, one might be a member of basketball club and football club simultaneously. Thus, Palla *et al.* [15] extended the problem of nonoverlapping community detection to find overlapping communities. After that, researchers have proposed a variety of strategies and methods for overlapping community detection in complex networks, such as community detection algorithms based on clique percolation theory [16], dynamic label propagation [17], link community detection [18], and local connectedness [19].

Among the exiting methods for community structure detection, one category is based on the evolutionary algorithms (EAs) since the community detection is usually an NP-hard optimization problem [20]. EAs are a class of artificial intelligent optimization meta-heuristics inspired by biology evolution, which can often perform well in solving complex problems because they ideally do not make any assumption for the corresponding problem. The work in [21] considered the community detection as a single-objective optimization problem. However, one single objective cannot describe the problem well, thus a number of multiobjective optimization algorithms [multiobjective evolutionary algorithms (MOEAs)] were proposed to find communities [22]–[29]. The existing MOEA-based community detection algorithms have mainly demonstrated two advantages in comparison with single-objective EAs as reported in [23] and [26]. On the one hand, the multiple conflicting objectives can alleviate potential disadvantages of single-objective optimization in community

Manuscript received December 31, 2016; revised April 13, 2017; accepted May 29, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61272152, Grant 61672033, Grant 61502001, Grant 61502004, and Grant 61502012, in part by the Beijing Natural Science Foundation under Grant 4164096, in part by the Fundamental Research Funds for the Central Universities under Grant 2652015340, and in part by the Academic and Technology Leader Imported Project of Anhui University under Grant J01006057. This paper was recommended by Associate Editor J. Liu. *(Corresponding author: Xingyi Zhang.)*

L. Zhang, H. Pan, Y. Su, and X. Zhang are with the Institute of Bio-Inspired Intelligence and Mining Knowledge, School of Computer Science and Technology, Anhui University, Hefei 230601, China (e-mail: zl@ahu.edu.cn; bimk_phb@163.com; suyansen1985@163.com; xyzhanghust@gmail.com).

Y. Niu is with the School of Information Engineering, China University of Geosciences, Beijing 100083, China (e-mail: niuyunyun1003@163.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2017.2711038

detection, e.g., the resolution limit in modularity Q [30]. On the other hand, MOEA-based community detection algorithms can easily obtain a set of network divisions at different hierarchical levels, since MOEAs find a set of Pareto optimal solutions instead of a single solution in one run, where one solution corresponds to a network division. However, most of the exiting works focus on developing MOEAs to detect nonoverlapping communities and there are few MOEAs to detect overlapping communities due to the challenging of individual representation for encoding and decoding overlapping communities [28], [29], [31], [32].

To this end, in this paper, we propose a mixed representation-based MOEA (MR-MOEA) for overlapping community detection. In the proposed algorithm, we develop a mixed individual representation scheme to fast encode and decode the overlapping division of networks. This mixed representation consists of candidate overlapping nodes-based representation and nonoverlapping nodes-based representation, where different individual updating strategies are proposed for evolution. The motivation behind this new representation scheme is due to the fact that overlapping community detection is a very challenging task in complex networks, where the key difficulty lies in the identification of overlapping nodes between communities. Based on the above reason, we divide nodes into candidate overlapping nodes and nonoverlapping nodes with the aim to identify overlapping nodes from the candidate overlapping nodes by MOEAs, and the nonoverlapping nodes are used for detecting communities associated with the overlapping nodes. Specifically, the contributions of this paper can be summarized as follows.

1) We propose a mixed representation scheme to fast encode and decode the overlapping division of networks. Specifically, an effective algorithm is first suggested to find candidate overlapping nodes from the given network, and the nodes in the network are divided into two categories: 1) candidate overlapping nodes and 2) candidate nonoverlapping nodes. For candidate overlapping nodes, two statuses (i.e., activated and depressed) are used for encoding. For nonoverlapping nodes, the vector-based representation [27] is directly adopted for encoding.
2) Based on this mixed representation, we propose a MOEA named MR-MOEA for detecting overlapping communities. In MR-MOEA, different updating strategies are suggested for candidate overlapping nodes and nonoverlapping nodes, respectively, to perform effective search. We also adopt the framework of multi-objective discrete particle swarm optimization (MODPSO) [27] in MR-MOEA to guarantee a good tradeoff between the convergence and diversity of populations during evolution.
3) We evaluate the effectiveness of proposed algorithm MR-MOEA on ten real-world complex networks compared with six representative baseline methods. The experimental results show the superior performance of our method over the competing ones in detecting overlapping communities, which indicates that the proposed MR-MOEA is competitive and promising.

The remaining of this paper is organized as follows. We first give some preliminaries about overlapping community detection in EAs and related work in Section II. Then in Section III, we present the proposed MOEA, in which the mixed representation scheme is described in detail. Next, we report the empirical results to show the effectiveness of the proposed method in Section IV and finally conclude this paper in Section V.

II. Preliminaries and Related Work

In this section, we first give some preliminaries about community detection problems, and then present the related work about EAs for the problem of community detection.

A. Community Detection Problem

In this paper, only undirected and unweighted simple networks are considered. Formally, a network can be represented as a graph denoted by $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_n\}$ is the set of nodes and $E = \{(i, j)|v_i \in V, v_j \in V \text{ and } i \neq j\}$ is the set of edges. Given a network G, the task of community detection is to divide the nodes of the whole network G into small groups which are also called communities. Let $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ be the set of all communities in G, where C_i satisfies the following conditions:

$$C_i \subset V \text{ and } C_i \neq \emptyset, i = 1, 2, \ldots, k \tag{1}$$

$$C_i \neq C_j, \ \forall i \neq j \text{ and } i, j \in \{1, 2, \ldots, k\} \tag{2}$$

$$\bigcup_{i=1}^{k} C_i = V. \tag{3}$$

Note that each community is a proper subset of V and also not null, which means it is meaningless to find a community with all nodes or a community without any node. The joint set of all communities is equal to V. If the set of communities satisfies

$$C_i \cap C_j = \emptyset, \ \forall i \neq j \text{ and } i, j \in \{1, 2, \ldots, k\} \tag{4}$$

then $\mathcal{C}$ is a set of nonoverlapping communities. If the set of communities satisfies

$$C_i \cap C_j \neq \emptyset, \exists i \neq j \text{ and } i, j \in \{1, 2, \ldots, k\} \tag{5}$$

then $\mathcal{C}$ is a set of overlapping communities. In this paper, we focus on the problem of overlapping community detection.

B. Community Detection Problem in EAs

The community detection problem in EAs can be modeled as a multiobjective optimization problem with two objectives [26], [27], [29], [31], [32]. The first objective is to maximize the intralink density, i.e., link density between nodes in the same community. The other one is to minimize the interlink density, i.e., link density between nodes in different communities. There are several criteria proposed for measuring intralink and interlink densities. In this paper, we adopt the kernel k-means (KKM) [33] for measuring intralink density, while the ratio cut (RC) [27] is adopted for measuring interlink density. Given a network $G = (V, E)$ (suppose

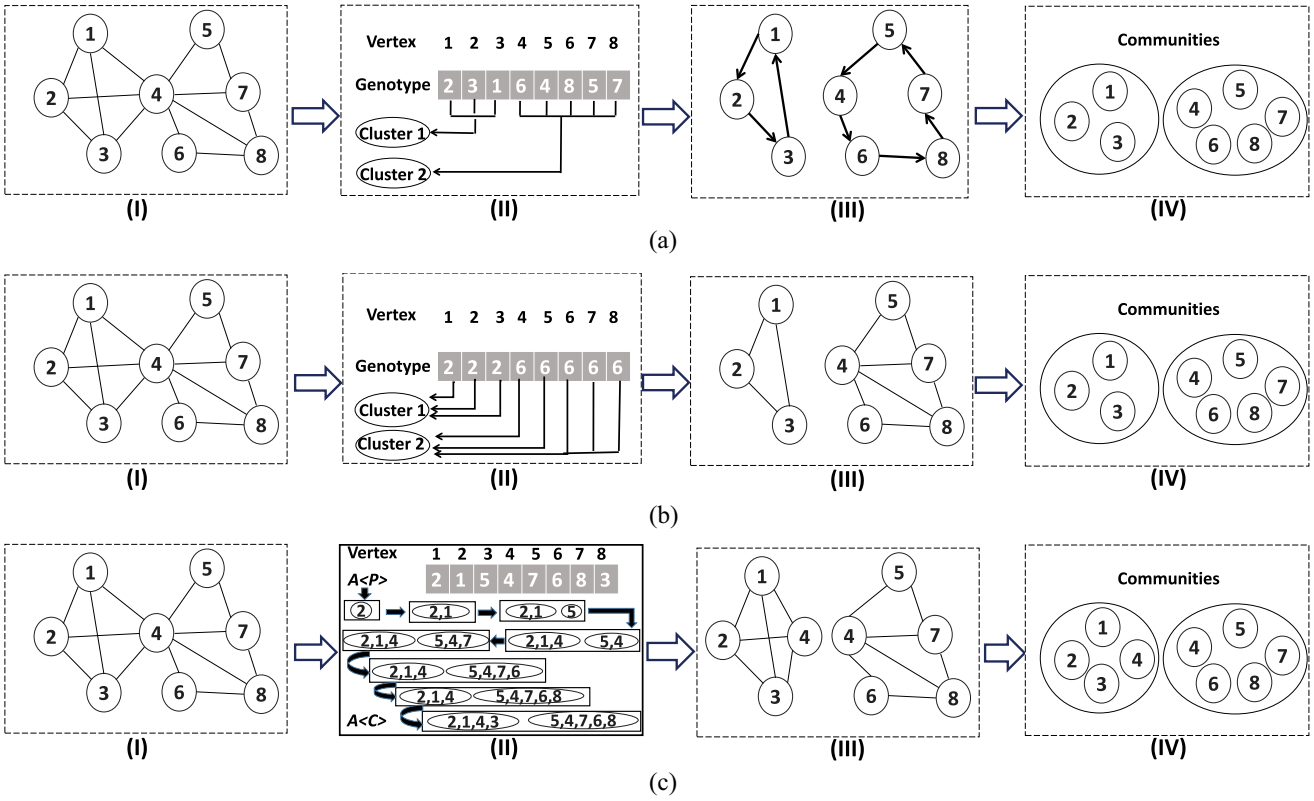


Fig. 1. Illustrative example of three existing individual representation schemes. (I) Initial network. (II) One possible genotype. (III) Transforming the genotype into a division of the network (note that the directed arrow are used only to help the understanding). (IV) Obtained communities (every connected component is considered as a community). (a) Locus-based individual representation scheme. (b) Vector-based individual representation scheme. (c) Indirect encoding method based on permutation.

$|V| = n$ and $|E| = m$) and a division $\mathcal{C}$ with k communities ($\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$). Let A be the adjacent matrix. Given one community C_1 and $\overline{C_1} = \mathcal{C} - C_1$, we define $L(C_1, C_1) = \sum_{i \in C_1, j \in C_1} A_{ij}$ and $L(C_1, \overline{C_1}) = \sum_{i \in C_1, j \in \overline{C_1}} A_{ij}$. Formally, these two measures are defined as

$$\min \begin{cases} \text{KKM} = 2(n-k) - \sum_{i=1}^{k} \frac{L(C_i, C_i)}{|C_i|} \\ \text{RC} \quad = \sum_{i=1}^{k} \frac{L(C_i, \overline{C_i})}{|C_i|}. \end{cases} \tag{6}$$

Based on the above definitions of KKM and RC, we can observe that the right operand of KKM can be considered as the sum of the link density of intracommunities, while RC can be considered as the sum of the link density of intercommunities. Minimizing both KKM and RC can guarantee that the links within a community are dense while the links between communities are spare. Thus, the community detection problem can be formulated as a two-objective optimization problem by minimizing both objectives KKM and RC.

C. Related Work

In recent years, many EAs have been proposed to detect communities in complex networks, especially for nonoverlapping community detection [10].

The community detection was first investigated as a single objective optimization problem in EAs [25], [34]. For example, Shi *et al.* [25] proposed a single objective genetic algorithm for detecting communities in complex networks. The genetic algorithm developed in [25] can only be applied to detect nonoverlapping communities in the complex networks, since it adopted the locus-based representation. As shown in Fig. 1(a), in this representation, each individual was represented as an integer array and the ith component of this array denotes one neighbor of node i.

With the development of MOEAs, more researchers are devoted to developing competitive MOEAs for community detection [26]–[29], [31], [32]. Pizzuti [26] proposed a multiobjective genetic algorithm named MOGA-Net to find communities in complex networks, where two objectives community score and community fitness were suggested to use. Empirical results indicated that the MOGA-Net can achieve promising performance in community detection, but this algorithm can only be used for nonoverlapping community detection due to the fact that it also adopted the locus-based representation.

Gong *et al.* [27] proposed a multiobjective discrete particle swarm optimization algorithm, named MODPSO, to solve the problem of community detection. In MODPSO, the framework of decomposition was introduced to guarantee a good tradeoff between the convergence and diversity of populations. The MODPSO was developed for nonoverlapping community detection and the empirical results showed that it could considerably improve the performance of community detection because the vector-based representation scheme was adopted

for individual encoding and decoding. Different from the locus-based representation scheme, in MODPSO, each individual was represented as an integer array and the *i*th component of this array denotes the community label, which can be seen from Fig. 1(b).

To address the problem of overlapping community detection, Liu *et al.* [28] proposed a MOEA named MEA_CDPs. In MEA_CDPs, an indirect encoding method based on permutation was proposed for individual representation, which was demonstrated to be suited for both nonoverlapping and overlapping community detection in complex networks. In the indirect encoding method, each gene of an individual is a random integer within the number of nodes in the network and thus a decoder is needed to transform them to the corresponding communities. In the decoding process, each node could be assigned to multiple communities according to its fitness evaluation, thus the scheme of individual representation can be adopted for overlapping community detection.

In [29], the indirect encoding method based on permutation was also considered and a new algorithm named MEAs_SCN was proposed for signed networks. It is worth noting that although the indirect encoding method based on permutation has been shown to be effective for overlapping community detection, it still suffers from high time-consuming for individual decoding, which can be seen from Fig. 1(c).

Li *et al.* [32] proposed an improved multiobjective quantum-behaved particle swarm optimization (named IMOQPSO) based on spectral-clustering to handle the problem of overlapping community detection. In this algorithm, a spectral clustering-based method was proposed for representing individuals. Each individual in this scheme represents the information of one community, the coordinates of the community center. Although the spectral clustering-based method is suitable for overlapping community detection, it still has some limitations, e.g., the number of communities needs to be set in advance and the evolutionary operators are difficult to be designed.

Recently, Wen *et al.* [31] proposed a maximal clique-based MOEA (MCMOEA) for overlapping community detection. In MCMOEA, a new representation scheme was suggested based on the maximal clique. Since any two maximal cliques may share the same nodes of the network, the overlap is an intrinsic property of the maximal clique. Thus, MCMOEA can handle the problem of overlapping community detection. However, based on maximal clique-based representation, some overlapping nodes could be missed if the maximal cliques cannot be found in the corresponding communities.

Different from the above works, this paper proposes an MR-MOEA for overlapping community detection. In MR-MOEA, we propose a mixed representation scheme consisting of candidate overlapping node-based representation and nonoverlapping node-based representation for fast encoding and decoding the overlapping divisions. As will be seen in Section IV, the performance of MR-MOEA is competitive compared to the state-of-the-art algorithms for overlapping community detection.

III. Proposed Algorithm MR-MOEA

In this section, we will present the proposed MR-MOEA method for overlapping community detection, including the proposed mixed representation scheme, evolutionary operators, the overall procedure and complexity analysis.

A. Mixed Representation Scheme

1) Main Idea: The mixed representation scheme in MR-MOEA is developed to address the problem of overlapping community detection, which consists of candidate overlapping node-based representation and nonoverlapping node-based representation. The main idea of mixed representation is to divide the nodes of a network into two groups based on the probability of each node being an overlapping node, and evolve them using different strategies to identify the overlapping nodes and communities of nonoverlapping nodes in the network. Specifically, we first find all candidate overlapping nodes from network G and then the nodes in G can be split into two groups: 1) the candidate overlapping node set (denoted as O) and 2) nonoverlapping node set (denoted as S). For candidate overlapping node set O, two possible statuses, i.e., activated or depressed, are assigned to each node in O. The node with status activated means this candidate overlapping node is an overlapping node, while the node with status depressed indicates that it is a nonoverlapping node. For nonoverlapping node set S, each node in S is assigned to a community label.

Given a network $G(V, E)$, let n be the number of all nodes in V, O, and S are the sets of candidate overlapping nodes and nonoverlapping nodes in V ($O \cap S = \emptyset$ and $O \cup S = V$), respectively. Formally, each individual can be defined as follows.

Definition 1 (Mixed Individual Representation): An individual I can be split into two components. The first component is the "status vector of candidate overlapping nodes," which is denoted as

$$I\langle O\rangle = \left\langle s_{v'_1}, s_{v'_2}, \ldots, s_{v'_m}\right\rangle \tag{7}$$

where $v'_i \in O$ and $1 \le i \le m$. The status for candidate overlapping node can be represented by "0" or "-1," where 0 means activated and -1 means depressed. The second component is "label vector of nonoverlapping nodes," which is denoted as

$$I\langle S\rangle = \left\langle l_{v'_{m+1}}, l_{v'_{m+2}}, \ldots, l_{v'_n}\right\rangle \tag{8}$$

where $l_{v'_k}$ is the label of node v'_k, $v'_k \in S$ and $m+1 \le k \le n$. The nodes in S with the same label are considered belonging to the same community.

Fig. 2 gives an illustrative example for the mixed representation, where the complex network contains eight nodes and 13 edges and an individual is denoted as $I = \langle 3, 3, 3, 0, 5, 5, 5, 5\rangle$ for $O = \{n_4\}$, $S = \{n_1, n_2, n_3, n_5, n_6, n_7, n_8\}$, $I\langle O\rangle = \langle 0\rangle$, and $I\langle S\rangle = \langle 3, 3, 3, 5, 5, 5, 5\rangle$.

For this mixed representation scheme, it is easy and fast to decode the individual. To be specific, for the nonoverlapping nodes, all nodes with the same label belong to a community. As for each candidate overlapping node, if it is identified as

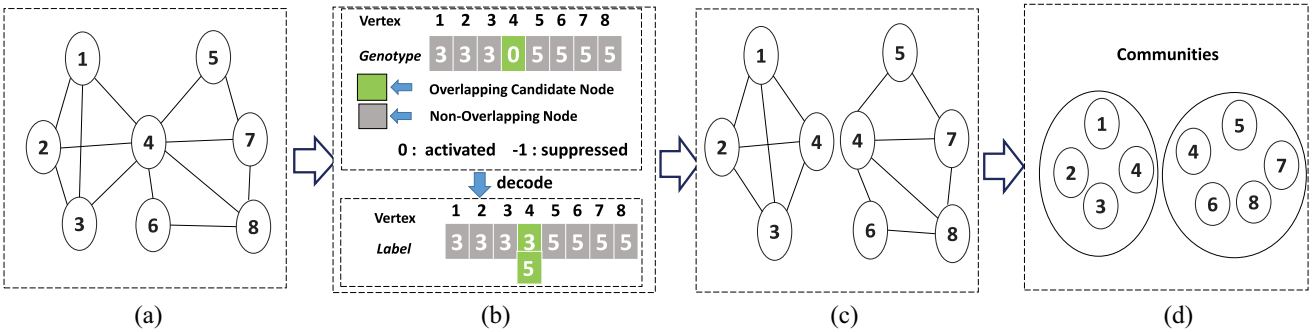


Fig. 2. Illustrative example of the mixed representation scheme. (a) Initial network. (b) One possible genotype and its corresponding decoding procedure. (c) Transforming the genotype into a division of the network. (d) Obtained communities.

an overlapping node, then this node is considered as an overlapping node of all communities which the node connects to; if the candidate overlapping node is identified as a nonoverlapping node, then the label of this node is determined by the maximum number of neighbors in a community. Let NTagSet be the label set of neighbors of a candidate overlapping node n_i. If n_i is activated, then all labels in NTagSet are used as the labels of n_i. If n_i is depressed, then we use the label in NTagSet with the maximum appearance (denoted as *NTagbest*) as the label of n_i. Formally

$$\text{gene } (i) = \begin{cases} \text{NTagSet} & \text{if node } n_i \text{ is activated} \\ \text{NTagbest} & \text{if node } n_i \text{ is depressed.} \end{cases} \tag{9}$$

As shown in Fig. 2(b), the mixed individual representation vector is [3, 3, 3, 0, 5, 5, 5, 5]. The candidate overlapping node n_4 is identified as an overlapping node and the neighbors of n_4 is $\{n_1, n_2, n_3, n_5, n_6, n_7, n_8\}$, thus NTagSet of n_4 is $\{3, 5\}$ since only two labels 3 and 5 appear in its neighbors. Because node n_4 is identified as an overlapping node, the labels of n_4 are 3 and 5. According to the fact that nodes with the same label belong to the same community, we can quickly get the overlapping communities. As shown in Fig. 2(c) and (d), the network can be divided into two communities: 1) $\{n_1, n_2, n_3, n_4\}$ and 2) $\{n_4, n_5, n_6, n_7, n_8\}$ according to the decoded genotype shown in Fig. 2(b). If we modify the mixed individual representation vector in Fig. 2 as $[3, 3, 3, -1, 5, 5, 5, 5]$, then the node n_4 is depressed. In other words, this node is identified as a nonoverlapping node, thus the label 5 is used as the label of this node, since in NTagSet the number of appearances of label 3 is 3 while that of label 5 is 4 (e.g., $NTagbest = 5$). This means that node n_4 belongs to only one community $\{n_4, n_5, n_6, n_7, n_8\}$. In the following, we will suggest an algorithm to find candidate overlapping nodes.

2) Candidate Overlapping Nodes Finding Algorithm: Though analyzing the overlapping communities in real complex networks, the following two observations can be found.

1) *Observation 1:* For each overlapping node of several communities, there usually exists one neighboring node in each community which is densely connected to the overlapping node in the community.
2) *Observation 2:* The links between communities that have at least an overlapping node are spare enough to make these communities unable to form one community.

The above two observations are directly obtained from the definition of community structure "community structure refers to that the nodes in the same community have dense connections and have sparse connections between different communities [9]." To be specific, for observation 1, it can easily be found that an overlapping community cannot be formed if the links between the overlapping node and the community is much sparer than those between the overlapping node and the other communities that share the overlapping node according to the definition of overlapping community structure. The dense links between the overlapping node and the community usually lead to the fact that the links between the overlapping node and one of its neighbors in the community are also dense in complex networks. This phenomenon is very common in real-world networks, such as the scientific collaboration network, where there exist many researchers with broad research interests who belong to different communities simultaneously. For one such researcher, in each of his communities, we can find that he/she often co-authors much more articles with a very small number of researchers in the community. As for observation 2, it can also be easily obtained from the definition of community structure since the links between communities should be sparer than those in each community. Otherwise, the communities will form a larger community.

Based on the two observations, we give the following definitions. Let n_i be a node in G.

Definition 2 (Key Neighboring Node): The key neighboring node of n_i, denoted as n_i^{KN}, is the node in the neighborhood of n_i which has the largest number of common neighboring nodes with n_i.

Definition 3 (Key Neighboring Subgraph): The key neighboring subgraph of n_i, denoted as G_i^{KN}, is the subgraph consisting of key neighboring node of n_i (i.e., n_i^{KN}) and common neighboring nodes of n_i and n_i^{KN}.

For example, as shown in Fig. 2(a), the neighboring nodes of n_1 are $\{n_2, n_3, n_4\}$. The common neighboring nodes of n_1 and n_2, n_1 and n_3, and n_1 and n_4 are $\{n_3, n_4\}$, $\{n_2, n_4\}$, and $\{n_2, n_3\}$, respectively. Since the number of common neighboring nodes of n_1 and n_2, n_3, and n_4 are all equal to 2, the key neighboring node of n_1 can be one of n_2, n_3, and n_4. Suppose the key neighboring node of n_1 is n_2, then the common neighboring nodes of n_1 and n_2 is $\{n_3, n_4\}$ and the key neighboring subgraph of n_1 is the subgraph of G consisting of three nodes $\{n_2, n_3, n_4\}$.

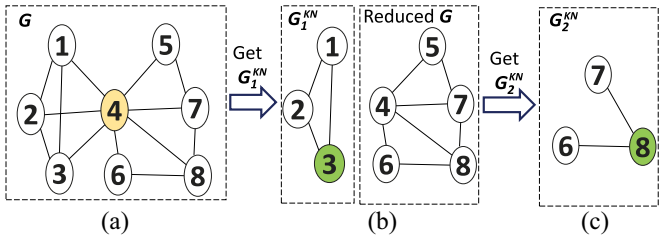


Fig. 3. Procedure to determine whether node n_4 is an overlapping candidate node. (a) Initial graph topology G. (b) n_3 is one of key neighboring nodes of n_4 and G_1^k containing nodes $\{n_1, n_2, n_3\}$ is the key neighboring subgraph of n_4. After removing G_1^{KN} from G, the reduced G is obtained. (c) n_8 is another key neighboring node of n_4 and G_2^{KN} containing nodes $\{n_6, n_7, n_8\}$ is another key neighboring subgraph of n_4.

Based on the above definitions and observations, we can find that a node n_i would have a high probability of being an overlapping node if the following two conditions are both satisfied.

1) *Condition 1:* There are at least two different key neighboring subgraphs of n_i in the complex network.
2) *Condition 2:* The links between any two key neighboring subgraphs of n_i should be spare.

The above two conditions are suggested to characterize the two observations, respectively. Due to the fact that we do not know the exact communities which an overlapping node locates, we consider each key neighboring subgraph as a community in this paper. Although the above two conditions may not find all overlapping nodes, experimental results presented in Section IV will demonstrate their effectiveness in finding candidate overlapping nodes.

Let A be the adjacent matrix of network G. Given two key neighboring subgraphs G_1^{KN} and G_2^{KN}, we define $L(G_1^{\mathrm{KN}}, G_2^{\mathrm{KN}}) = \sum_{i \in G_1^{\mathrm{KN}}, j \in G_2^{\mathrm{KN}}} A_{ij}$ as the number of links between G_1^{KN} and G_2^{KN}. Then, the link closeness (denoted as LC) is defined to measure the LC between two neighboring subgraphs. Formally

$$\mathrm{LC}\left(G_1^{\mathrm{KN}}, G_2^{\mathrm{KN}}\right) = \max\left\{\frac{L\left(G_1^{\mathrm{KN}}, G_2^{\mathrm{KN}}\right)}{L\left(G_1^{\mathrm{KN}}, G_1^{\mathrm{KN}}\right)}, \frac{L\left(G_1^{\mathrm{KN}}, G_2^{\mathrm{KN}}\right)}{L\left(G_2^{\mathrm{KN}}, G_2^{\mathrm{KN}}\right)}\right\}. \tag{10}$$

The links of key neighboring subgraphs G_1^{KN} and G_2^{KN} are spare if the LC is smaller than a given threshold μ. The parameter μ can directly control the number of candidate overlapping nodes. A larger value of μ will lead to more candidate overlapping nodes and vice versa. In the proposed MR-MOEA, the threshold of μ does not need to set too small, since the candidate overlapping nodes should contain all true overlapping nodes. In Section IV-C, we will give the sensitivity analysis of parameter μ.

Take the network shown in Fig. 2(a) as an example, for node n_1, there is only one key neighboring subgraph, which consists of $\{n_2, n_3, n_4\}$. The condition 1 is not satisfied, hence n_1 is not considered as a candidate overlapping node. For node n_4, the key neighboring subgraph of n_4 containing nodes $\{n_1, n_2, n_3\}$ (i.e., G_1^{KN}) is first found. After removing the nodes n_1, n_2, n_3 and their related links from G, we can get another key neighboring subgraph of n_4 (i.e., G_2^{KN}) containing nodes $\{n_6, n_7, n_8\}$. Fig. 3 gives the main procedure of determining whether n_4 is a candidate overlapping node or not. As can be seen, there is no link between G_1^{KN} and G_2^{KN}, thus $\mathrm{LC}(G_1^{\mathrm{KN}}, G_2^{\mathrm{KN}}) = \max\{(0/3), (0/2)\} = 0$. Suppose $\mu = 0.1$, the links between G_1^{KN} and G_2^{KN} are spare, thus n_4 is a candidate overlapping node. Algorithm 1 summarizes the details of finding candidate overlapping nodes in complex networks.

Algorithm 1: *Find_OverlappingCandidates*(G)

Input: A: the adjacent matrix of G; *Num*: the number of nodes in network G;
Output: $\boldsymbol{O}$: Candidate Overlapping Nodes;

```
O ← ∅;
for i = 1 to Num do
    NB_i ← get neighbors of node n_i;
    G_{i.1}^{KN}, G_{i.2}^{KN} ← ∅; //two key neighboring subgraphs
    for j = 1 to 2 do
        n_i^{KN} ← get one key neighboring node of n_i from NB_i;
        CN_i ← get common neighboring nodes between n_i and n_i^{KN};
        G_{i.j}^{KN} ← n_i^{KN} ∪ CN_i;
        NB_i = NB_i − G_{i.j}^{KN};
    end for
    if LC(G_{i.1}^{KN}, G_{i.2}^{KN}) ≤ μ then
        O ← n_i ∪ O;
    end if
end for
```

B. Evolutionary Operators

With the proposed mixed representation, we adopt particles used in particle swarm optimization (PSO) algorithm [35] as individuals to create good solutions by learning their own historical best solution and the global historical best solution. This is because that crossover and mutation operators are prone to create undesirable solutions for the nonoverlapping part in the mixed representation using labels of nodes [27]. In PSO, each particle has a "position" and a "velocity," where position vector is used to represent a solution and velocity vector is utilized to update the solution.

1) Particle Position and Velocity Vectors: The position of a particle in PSO is defined as the mixed representation $X_i = \langle x_1, x_2, \ldots, x_n \rangle$, where x_i $(1 \le i \le n)$ is a random integer from -1 to n and n is the total number of nodes in the network. In the mixed representation, there are two kinds of nodes, i.e., candidate overlapping nodes and nonoverlapping nodes. If n_i is a candidate nonoverlapping node, then $x_i \in \{1, 2, \ldots, n\}$. If n_i is a candidate overlapping node, then $x_i \in \{-1, 0\}$.

The velocity vector of particle i is defined as $V_i = \langle v_1, v_2, \ldots, v_n \rangle$, $v_i \in \{0, 1\}$, $1 \le i \le n$. If $v_i = 1$, the corresponding element x_i in the position vector will be changed; otherwise, x_i keeps unchanged.

2) Particle Velocity Updating: The velocity of each particle will be updated by learning from its personal best position, i.e., Pbest_i, and the global best position found by the population, Gbest. Here, we adopt the approach suggested in [27]

for updating the velocity. Specifically

$$V_i = \text{sig}(\omega V_i + c_1 r_1(\text{Pbest}_i \oplus X_i) + c_2 r_2(\text{Gbest} \oplus X_i)) \quad (11)$$

where ω is the inertia weight, c_1 is the cognitive component, c_2 is the social component, and r_1 and r_2 are two numbers randomly generated in the range [0, 1].

In (11), the operator $\oplus$ is defined as a XOR operator. Suppose the function $Y = \text{sig}(X)$, where $Y = \langle y_1, y_2, \ldots, y_n \rangle$ and $X = \langle x_1, x_2, \ldots, x_n \rangle$. The value of y_i is defined as follows:

$$\begin{cases} y_i = 0 & \text{if rand } (0, 1) \geq \text{sigmoid}(x_i) \\ y_i = 1 & \text{if rand } (0, 1) < \text{sigmoid}(x_i) \end{cases} \quad (12)$$

where the sigmoid function is defined as $\text{sigmoid}(x) = (1/1 + e^{-x})$.

Just as the work in [27] suggested, the inertia weight ω is randomly generated in the range of [0, 1], the cognitive component c_1 and the social component c_2 are both set to the value of 1.494 in MR-MOEA for all experiments conducted in this paper.

3) Particle Position Updating: Based on the above velocity updating rule and the proposed mixed representation, we can define the position updating rule as the following form:

$$X_{ij}^{t+1} = X_{ij}^t \otimes V_{ij}^t \quad (13)$$

where $\otimes$ is an operator for updating particles defined as

$$X_{ij}^{t+1} = \begin{cases} X_{ij}^t & \text{if} \quad V_{ij} = 0 \\ \sim X_{ij}^t & \text{if} \quad V_{ij} = 1 \,\&\, \& \, X_{ij}^t \in \{-1, 0\} \\ \text{NBest}_{ij} & \text{if} \quad V_{ij} = 1 \,\&\, \& \, X_{ij}^t \in \{1, 2, \ldots, n\} \end{cases} \quad (14)$$

where $X_{ij}^t \in \{-1, 0\}$ (or $\in \{1, 2, \ldots, n\}$) means that the node n_j in individual X_i^t is a candidate overlapping node (or a nonoverlapping node).

From (14), it can be found that we here adopt a different updating strategy for the candidate overlapping nodes and nonoverlapping nodes. Specifically, if $V_{ij} = 0$, then X_{ij}^{t+1} will be assigned with the value of X_{ij}^t no matter whether the associated node n_j is considered as an overlapping one or not in particle X_i^t. If $V_{ij} = 1$ and n_j is a candidate overlapping node, then X_{ij}^{t+1} will be assigned with the opposite value of X_{ij}^t (i.e., $\sim X_{ij}^t$). For example, if $X_{ij}^t = 0$, then $X_{ij}^{t+1} = -1$, and if $X_{ij}^t = -1$, then $X_{ij}^{t+1} = 0$. If $V_{ij} = 1$ and n_j is a nonoverlapping node, then X_{ij}^{t+1} will be assigned with the label possessed by the majority of the neighbors of n_j (i.e., $NBest_{ij}$).

Fig. 4 gives a schematic example of the detailed operations about particle status updating rules. In this figure, X_i is the current position, Pbest_i is the personal best position of particle i and Gbest is the global best solution of the population. $V1$, $V2$, and $V3$ are three intermediate variables. $V4$ and X_i^{t+1} are figured out by (11) and (14), respectively.

C. Overall Procedure of MR-MOEA

MR-MOEA is implemented using the proposed mixed representation scheme, the improved particle updating rules as evolutionary operators described above, and the framework of MODPSO [27] to guarantee a good tradeoff between the convergence and diversity of populations during evolution. We adopt Tchebycheff approach [29] instead of the weighted sum approach in MR-MOEA for decomposition, since the shape of Pareto front is unknown and the weighted sum approach is more suitable for concave Pareto front. Let pop be the number of subproblems (i.e., the size of population) and $\{\lambda_1, \lambda_2, \ldots, \lambda_{\text{pop}}\}$ be a set of evenly spread weight vectors, where $\lambda_i = \langle \lambda_i^1, \lambda_i^2 \rangle$ satisfying $\lambda_i^1 + \lambda_i^2 = 1$ ($\lambda_i^1, \lambda_i^2 \in [0, 1]$).

Algorithm 2: General Framework of MR-MOEA

Input: A: the adjacent matrix of a network; *gene*: the number of generations; *pop*: the size of population; c_1, c_2: the learning factors; ω: the inertia weight; $\{\lambda_1, \lambda_2, \ldots, \lambda_{pop}\}$: the set of weight vectors; *NS*: the size of neighbours;
Output: Optimal solutions

Step1: the candidate overlapping nodes finding
$O \leftarrow$ candidate overlapping nodes are obtained by Algorithm 1;
Step2: initialization
Individual initialization: if a node in individual X_m belongs to O, then the label of the node is -1 or 0. Otherwise, the label is the order of the node. Calculate the objective function values of X_m;
Position initialization:$P = \{X_1, X_2, \ldots, X_{pop}\}$;
Velocity initialization:$V = \{V_1, V_2, \ldots, V_{pop}\}$;
Initialize reference point z^*;
Pbest initialization:$Pbest = \{X_1, X_2, \ldots, X_{pop}\}$;
Step3: population evolution
$N = \{N_1, N_2, \ldots, N_{pop}\} \leftarrow$ obtain the neighbors of each individual by computing Euclidean distance based on the set of weight vectors;
for $t = 1$ to *gene* **do**
  for $i = 1$ to *pop* **do**
    $Gbest_i \leftarrow$ Randomly select an individual from N_i;
    $V_i \leftarrow$ Compute velocity according to Equation (11);
    $X_i^{t+1} \leftarrow X_i^t \otimes V_i$ (refer to Equation (14));
    Compute objective function, if X_i^{t+1} is better than any individual I in N_i, then replace I with X_i^{t+1} and update reference point z^*;
    if X_i^{t+1} dominates $Pbest_i$, then replace $Pbest_i$;
  end for
end for

Algorithm 2 shows the general framework of MR-MOEA, which consists of three steps: the candidate overlapping nodes finding, initialization, and population evolution. In the first step, the candidate overlapping nodes are obtained by using Algorithm 1. In the second step, the population with pop individuals is initialized based on the mixed representation scheme proposed in Section III-A. Specifically, for pop individuals position initialization (i.e., $P = \{X_1, X_2, \ldots, X_{\text{pop}}\}$), if a node is a candidate overlapping node identified by Algorithm 1, then the label of node is -1 (depressed) or 0 (activated) randomly chosen. Otherwise, the label of node is the order of the node. For pop individuals velocity initialization (i.e., $V = \{V_1, V_2, \ldots, V_{\text{pop}}\}$), each V_i is assigned with the vector $\mathbf{0}$. The reference point z^* is initialized by the best values of KKM and RC found in the initial population. The initial Pbest_i associated with X_i is assigned with X_i^1, $i \in \{1, 2, \ldots, n\}$. In the third step, the population evolves according to the evolutionary operators described in Section III-B. A new individual X_i^{t+1} is generated by performing particle velocity updating and position updating rules on each current individual X_i^t. After X_i^{t+1} is generated, its objective function values are calculated. If X_i^{t+1}

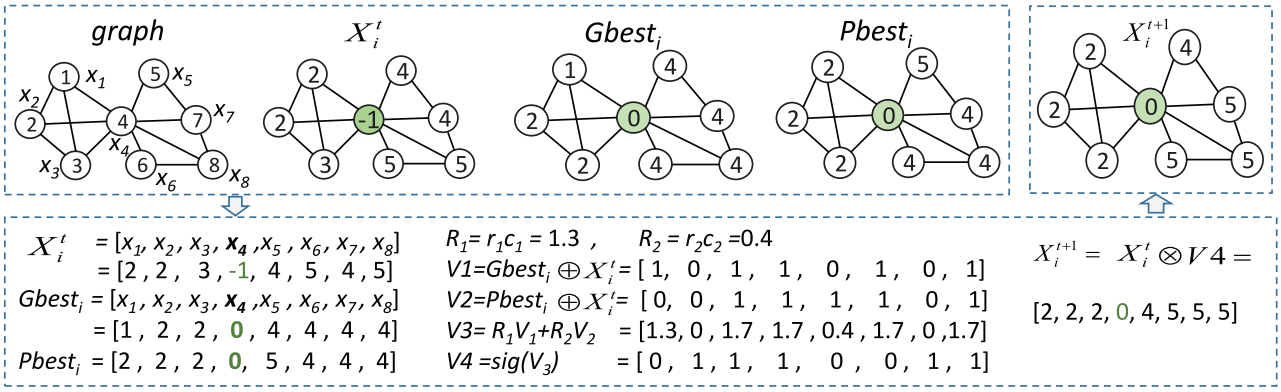


Fig. 4. Example that show how a particle updates its status, where X_i is the current position, Pbest_i is the personal best position of particle i, and Gbest is the global best solution of the swarm. $V1$, $V2$, and $V3$ are three intermediate variables. $V4$ and X_i^{t+1} are figured out by (11) and (14), respectively.

is better than any individual I in N_i (i.e., neighbors of individual X_i^t by calculating Euclidean distance based on the weight vector λ), then replace I with X_i^{t+1} in N_i and update reference point z^* based on the updated population. If X_i^{t+1} dominates Pbest_i, then replace the Pbest_i with X_i^{t+1}. The algorithm halts until the maximum number of generations reaches.

D. Complexity Analysis

Let n be the number of nodes in the network G to be considered, m be the number of edges in G, gene be the maximum number of generations, pop be the size of population. The computational complexity of MR-MOEA can be obtained by calculating the complexity of the above three steps. In the first step, each node should be checked whether it is a candidate overlapping node or not, thus the computational complexity of the first step is $O(n \times D \times D)$, where D is the largest node degree. In the second step, the computational complexity of initialization is $O(\text{pop} \times n)$. In the third step, the major computational cost is taken for the particle updating and the calculation of objective function values of individuals, thus the computational complexity of the third step is $O((n+m) \times \text{pop} \times \text{gene})$. Because we often have more edges than nodes in a network, the complexity of third step is $O(m \times \text{pop} \times \text{gene})$. Hence, the computational complexity of MR-MOEA is $O(m \times \text{pop} \times \text{gene})$, which is lower than most existing MOEAs for overlapping community detection, such as MEAs_SCN whose complexity is $O(n^2 \times \text{pop} \times \text{gene})$ as reported in [29].

IV. Experimental Results

In this section, we will conduct a series of experiments based on ten real-world networks with different characteristics to validate the performance of the proposed algorithm MR-MOEA by comparing it with six representative baseline algorithms. In the following, we first present the experimental setting (including comparison algorithms, real-world data sets and evaluation metrics), then compare the experimental results with baseline algorithms, finally discuss the sensitivity of parameter in the proposed MR-MOEA.

TABLE I
Real-World Networks With Different Characteristics

Real Networks	Nodes	Edges	Ave. Degree	Real Clusters
karate [38]	34	78	4.59	2
dolphin [39]	62	159	5.13	2
football [40]	115	613	10.66	12
polbook [40]	105	441	8.4	3
Yeast-D2 [41]	1443	6993	9.69	162
Y2H [42]	1966	2705	2.75	203
jazz [43]	198	2742	27.70	Unknown
netscience [27]	1589	2742	3.45	Unknown
PPI [44]	2456	6265	5.26	Unknown
blogs [45]	3984	6803	3.41	Unknown

A. Experimental Setting

1) Comparison Algorithms: In this paper, six representative algorithms are chosen to compare with the proposed MR-MOEA. Specifically, MR-MOEA is compared with three representative algorithms which are not based on MOEAs, i.e., Zhang and Wu's [36] algorithm, local maximum degree nodes based algorithm [19], and nonnegative matrix factorization approach (NMF) [37]. In addition, we also compared MR-MOEA with three representative MOEAs for overlapping community detection, i.e., IMOQPSO [32], MEAs_SCN [29] and MCMOEA [31]. For each baseline algorithm, we use the code provided by the authors and adopt the parameters suggested in their papers. It is worth noting that MEAs_SCN can also be used for unsigned networks despite that it was suggested mainly for addressing signed networks, as reported in [29].

For a fair comparison, in the four MOEAs the population size PS is all set to 100 and the maximum number of generations gene is set to 100. The threshold μ for controlling the number of candidate overlapping nodes in MR-MOEA is set to 0.1. The experimental results for all algorithms are obtained by averaging over 20 independent runs. All the experiments are carried out on computers with AMD A6-3620 2.20-GHz CPU, 4-GB RAM and Windows 7 operating system.

2) Real-World Networks: We adopt ten popular real-world networks with different characteristics to evaluate the performance of proposed algorithm. These networks are Zachary's [38] karate club, dolphin social network [39],

TABLE II
COMPARISON RESULTS OF Q_{ov} ON THE TEN REAL-WORLD NETWORKS, WHERE SYMBOLS $+$, $-$, AND $\approx$ INDICATE THAT THE PERFORMANCE IS SIGNIFICANTLY BETTER, SIGNIFICANTLY WORSE AND STATISTICALLY SIMILAR TO THAT OF MR-MOEA, RESPECTIVELY

Network	Metric	MR-MOEA	IMOQPSO	MEAs_SCN	MCMOEA	Zhang	LMD	NMF
karate	Q_{ov}_max	**0.229**	0.213	0.204	0.210	0.216	0.216	0.205
	Q_{ov}_avg	**0.223**	0.208^{-}	0.180^{-}	0.208^{-}	0.212^{-}	0.204^{-}	0.205^{-}
	Std	0.007	0.004	0.022	0.002	0.005	0.024	0
dolphin	Q_{ov}_max	**0.271**	0.264	0.221	0.206	0.261	0.261	0.200
	Q_{ov}_avg	**0.264**	0.258^{-}	0.201^{-}	0.198^{-}	0.251^{-}	0.194^{-}	0.200^{-}
	Std	0.011	0.008	0.017	0.049	0.089	0.102	0
football	Q_{ov}_max	**0.306**	0.243	0.226	0.279	0.282	0.284	0.303
	Q_{ov}_avg	**0.303**	0.235^{-}	0.207^{-}	0.274^{-}	0.271^{-}	0.246^{-}	$\mathbf{0.303}^{\approx}$
	Std	0.005	0.014	0.020	0.087	0.011	0.090	0
polbook	Q_{ov}_max	**0.267**	0.244	0.246	0.228	0.237	0.263	0.259
	Q_{ov}_avg	**0.265**	0.241^{-}	0.216^{-}	0.222^{-}	0.225^{-}	0.241^{-}	0.259^{-}
	Std	0.005	0.004	0.042	0.011	0.012	0.071	0
Yeast-D2	Q_{ov}_max	**0.410**	0.302	0.336	0.257	0.391	0.346	0.309
	Q_{ov}_avg	**0.405**	0.254^{-}	0.334^{-}	0.253^{-}	0.389^{-}	0.275^{-}	0.309^{-}
	Std	0.004	0.039	0.002	0.003	0.007	0.063	0
Y2H	Q_{ov}_max	**0.299**	0.281	0.071	0.228	0.244	0.281	0.228
	Q_{ov}_avg	**0.286**	0.281^{-}	0.052^{-}	0.227^{-}	0.243^{-}	0.281^{-}	0.228^{-}
	Std	0.008	0	0.030	0.001	0.002	0	0
jazz	Q_{ov}_max	**0.223**	0.156	0.140	0.156	0.148	0.146	0.155
	Q_{ov}_avg	**0.221**	0.087^{-}	0.136^{-}	0.144^{-}	0.139^{-}	0.143^{-}	0.155^{-}
	Std	0.002	0.051	0.006	0.024	0.010	0.007	0
netscience	Q_{ov}_max	**0.460**	0.362	0.421	0.449	0.397	0.396	0.413
	Q_{ov}_avg	**0.456**	0.359^{-}	0.401^{-}	0.447^{-}	0.393^{-}	0.395^{-}	0.411^{-}
	Std	0.001	0.003	0.009	0.003	0.002	0.001	0.006
PPI	Q_{ov}_max	**0.313**	0.256	0.215	0.199	0.283	0.283	0.237
	Q_{ov}_avg	**0.311**	0.251^{-}	0.207^{-}	0.196^{-}	0.278^{-}	0.278^{-}	0.230^{-}
	Std	0.006	0.005	0.007	0.004	0.004	0.004	0.005
blogs	Q_{ov}_max	**0.394**	0.356	0.245	0.328	0.351	0.352	0.342
	Q_{ov}_avg	**0.389**	0.353^{-}	0.243^{-}	0.327^{-}	0.344^{-}	0.345^{-}	0.338^{-}
	Std	0.011	0.002	0.003	0.001	0.007	0.012	0.003
$+/-/\approx$			0/10/0	0/10/0	0/10/0	0/10/0	0/10/0	0/9/1

American college football [40], books about U.S. politics [40], Yeast PPI dataset Yeast-D2 [41], Yeast PPI dataset Y2H [42], scientist collaboration network [27], jazz musicians network [43], Yeast PPI dataset [44], and blogs network [45]. The characteristics of these networks are given in Table I. Note that karate, dolphin, football, polbooks, Yeast-D2, Y2H are networks with ground truth community structure and the true community structure of the remaining four networks is still unknown.

3) Evaluation Metrics: In this paper, we use two popular metrics to evaluate the effectiveness of the proposed method.

The first metric is the extended modularity (Q_{ov}) [46], which is often used when the true community structure is not known. For most real networks, their ground truth community structures are unknown. Formally, the Q_{ov} is defined as

$$Q_{ov} = \frac{1}{2m}\sum_{i=1}^{l}\sum_{v\in C_i, w\in C_i}\frac{1}{O_v O_w}\left[A_{vw} - \frac{k_v k_w}{2m}\right] \tag{15}$$

where A is the adjacent matrix, C_i denotes a community ($1 \leq i \leq l$ and l is the number of communities), m is the number of edges, O_v is the number of communities that node v belongs to, and k_v is the degree of node v. The larger the value of Q_{ov}, the better the result of overlapping community structure detected.

The other one is the generalized normalized mutual information (gNMI) [47], which can only be used when the true community structure is already known. Thus, gNMI is used to estimate the similarity between the true community result and the detected one. Formally, the gNMI is defined as

$$\text{gNMI}(A, B) = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} C_{ij}\log\left(C_{ij}N/C_{i.}C_{.j}\right)}{\sum_{i=1}^{C_A} C_{i.}\log\left(C_{i.}/N\right) + \sum_{j=1}^{C_B} C_{.j}\log\left(C_{.j}/N\right)} \tag{16}$$

where C_A (C_B) is the number of communities in division A (B), C is the confusion matrix whose element C_{ij} is the number of nodes shared by community i in division A and by community j in division B, $C_{i.}$ ($C_{.j}$) is the sum of elements of C in row i (column j), and N is the number of nodes in the network. A larger value of gNMI indicates the community structure detected by an algorithm is more similar to the ground truth.

In summary, gNMI can only be used on networks with ground truth community structure, while Q_{ov} can be used both on networks with ground truth and those without ground truth. Hence, in our experiments, Q_{ov} is used for measuring ten real networks, and gNMI is only used for the six real networks with true community structure.

B. Comparison Results Between MR-MOEA and Baselines

In the following, we first give the comparison results in terms of Q_{ov} on ten real datasets, and then present the comparison results in terms of gNMI on the six real datasets with ground truth community structure. Finally, we give further discussion on the communities detected by the proposed MR-MOEA. Note that, for the MOEA-based algorithms, the solution with the best value of Q_{ov} and gNMI

TABLE III
gNMI VALUES OF THE SEVEN ALGORITHMS ON THE SIX REAL-WORLD NETWORKS WITH GROUND TRUTH, WHERE SYMBOLS $+$, $-$, AND $\approx$ INDICATE THAT THE PERFORMANCE IS SIGNIFICANTLY BETTER, SIGNIFICANTLY WORSE AND STATISTICALLY SIMILAR TO THAT OF MR-MOEA, RESPECTIVELY

Network	Metric	MR-MOEA	IMOQPSO	MEAs_SCN	MCMOEA	Zhang	LMD	NMF
karate	$gNMI\_max$	**1**	0.708	0.383	0.918	0.513	0.513	0.837
	$gNMI\_avg$	**1**	0.698^{-}	0.375^{-}	0.890^{-}	0.496^{-}	0.447^{-}	0.837^{-}
	Std	0	0.024	0.042	0.069	0.052	0.104	0
dolphin	$gNMI\_max$	**1**	1	0.421	0.473	0.293	0.611	0.907
	$gNMI\_avg$	**1**	0.756^{-}	0.412^{-}	0.342^{-}	0.277^{-}	0.456^{-}	0.907^{-}
	Std	0	0.475	0.017	0.161	0.089	0.132	0
football	$gNMI\_max$	0.803	0.809	**0.927**	0.712	0.761	0.783	0.793
	$gNMI\_avg$	**0.803**	$0.798^{\approx}$	0.788^{+}	0.696^{-}	0.757^{-}	0.762^{-}	$0.793^{\approx}$
	Std	0	0.015	0.320	0.037	0.007	0.028	0
polbook	$gNMI\_max$	0.149	0.432	**0.482**	0.104	0.137	0.137	0.388
	$gNMI\_avg$	0.139	0.389^{+}	$\mathbf{0.416}^{+}$	0.098^{-}	0.093^{-}	0.118^{-}	0.388^{+}
	Std	0.014	0.032	0.062	0.008	0.059	0.017	0
Yeast-D2	$gNMI\_max$	**0.263**	0.205	0.252	0.257	0.184	0.205	0.228
	$gNMI\_avg$	**0.258**	0.201^{-}	0.245^{-}	0.252^{-}	0.182^{-}	0.201^{-}	0.228^{-}
	Std	0.005	0.004	0.007	0.007	0.001	0.004	0
Y2H	$gNMI\_max$	0.118	0.018	**0.121**	0.025	0.007	0.018	0.026
	$gNMI\_avg$	**0.115**	0.018^{-}	0.091^{-}	0.023^{-}	0.007^{-}	0.018^{-}	0.026^{-}
	Std	0.002	0.001	0.053	0.002	0	0	0
$+/-/\approx$			1/4/1	2/4/0	0/6/0	0/6/0	0/6/0	1/4/1

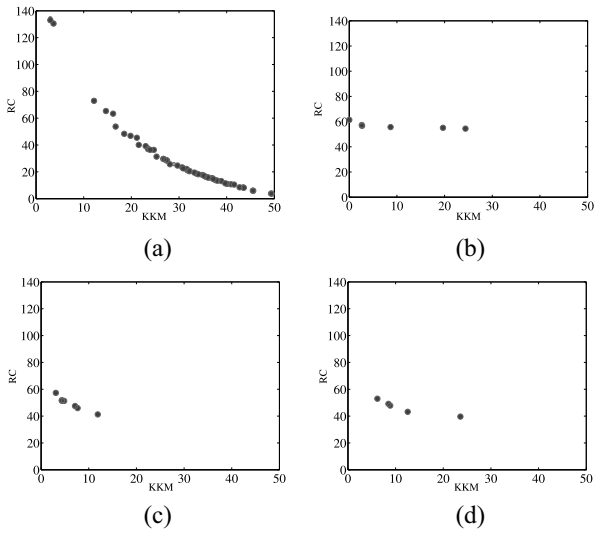


Fig. 5. Nondominated solutions obtained by four MOEA algorithms in objective space on network *karate*. (a) Our algorithm MR-MOEA. (b) IMOQPSO. (c) MEAs_SCN. (d) MCMOEA.

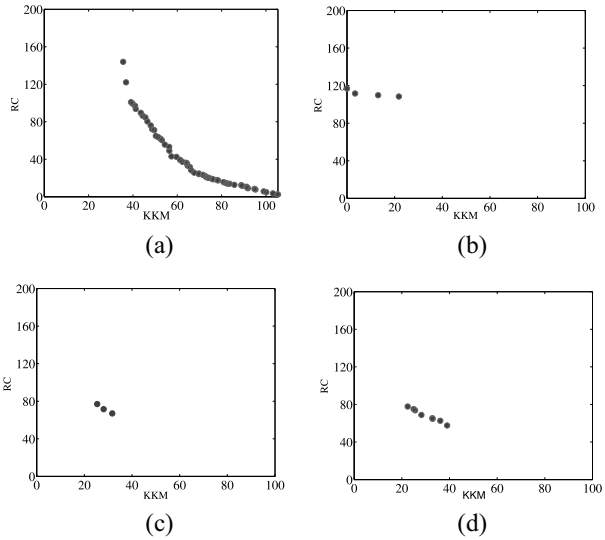


Fig. 6. Nondominated solutions obtained by four MOEA algorithms in objective space on network *dolphin*. (a) Our algorithm MR-MOEA. (b) IMOQPSO. (c) MEAs_SCN. (d) MCMOEA.

in the obtained nondominated solution set is used for comparison since this way has been widely adopted in existing MOEA-based community detection algorithm for comparing the performance [29], [31], [32].

1) Experimental Results in Terms of Q_{ov}: Table II shows the Q_{ov} values of the proposed algorithm MR-MOEA and the other six community detection algorithms on the ten real-world networks with different characteristics. We adopt the Wilcoxon rank sum test at a significance level of 0.05 to evaluate the statistical difference of the performance of the algorithms, where the symbols "+," "−," and "≈" indicate that the result is significantly better, significantly worse and statistically similar to that obtained by MR-MOEA, respectively. As can be seen, the proposed algorithm is not only better than non-MOEA-based algorithms but also better than MOEA-based algorithms in terms of Q_{ov}. MR-MOEA obtains the best Q_{ov} values on all ten real-world networks. It can also be found that most of the MOEA-based overlapping community detection algorithms achieve a better Q_{ov} value than non-MOEAs on the ten real-world networks, which shows that EAs are a promising framework for handling the problem of overlapping community detection.

Compared with the three MOEA-based algorithms under comparison, the proposed MR-MOEA achieves a better performance on the ten real-world networks in terms of Q_{ov}. The superior performance of MR-MOEA is attributed to the mixed representation scheme suggested in MR-MOEA. The IMOQPSO adopts the edge-based encoding, leading to the fact that its search space is much larger than node-based encoding used in MR-MOEA since a network often has

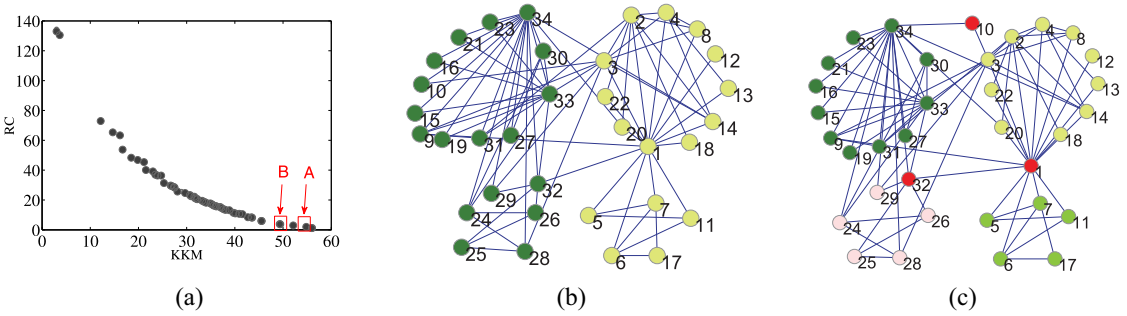


Fig. 7. Community structures obtained by MR-MOEA on network *karate*. Nodes with the same color belong to the same community and the red ones are overlapping nodes between different communities. (a) Nondominated solutions in final population. (b) Community structure of the solution A in (a). (c) Community structure of the solution B in (a).

more edges than nodes. The MEAs_SCN adopts the indirect encoding and MCMOEA uses maximal clique-based individual scheme, whose length of encoding a given network is not larger than the mixed representation. But, many duplicate solutions will occur in the population at each generation in these two encoding schemes, which deteriorates the performance of MOEAs when the population size is fixed. To verify the above analysis, we plot the final nondominated solutions obtained by the four MOEAs in objective space for networks karate and dolphin, which are shown in Figs. 5 and 6, respectively. As can be seen, the proposed MR-MOEA obtains much more nondominated solutions than the three compared MOEAs at the final generation on both networks, which addresses for the better performance of MR-MOEA.

2) Experimental Results in Terms of gNMI: In order to further show the performance of the proposed algorithm, we also adopt gNMI as the performance metric. However, gNMI can only be used in the datasets with the ground truth community structure. In our experiments, there are six datasets with real community structure. Table III shows the gNMI values of MR-MOEA and the other six community detection algorithms on the six networks, karate, dolphin, football, polbook, and Yeast-D2. As shown in this table, we can observe that MR-MOEA can obtain greatly better results than baseline algorithms on karate, dolphin, Yeast-D2, and Y2H in terms of both maximal gNMI (*NMI_max*) and average gNMI (*NMI_avg*). On network football, the maximal gNMI of MR-MOEA from 20 runs is 0.803, which is smaller than that of MEAs_SCN (i.e., 0.927). However, MR-MOEA can get the best average gNMI value on football. The similar results can also be found for network Y2H. On the network polbook, the performance of MR-MOEA is worse than IMOQPSO, MEAs_SCN and NMF in terms of gNMI. By comparing Tables II and III, it can be found that the superiority of proposed MR-MOEA seems to decrease when the performance metric gNMI is used instead of Q_{ov}. This fact may show that the objective functions adopted by the proposed MR-MOEA are closely related to the Q_{ov}, since gNMI is not always consistent with Q_{ov} as reported in [31], [45], and [48].

Based on the empirical results shown in Tables II and III, we can conclude that the proposed MR-MOEA algorithm holds a competitive performance in terms of both performance metrics Q_{ov} and gNMI on real networks.

3) Further Discussion on the Communities Detected by the Proposed MR-MOEA: It is worth mentioning that the proposed MR-MOEA algorithm is based on a MOEA framework, and each nondominated solution in the final generation corresponds to a division of the given network. In other words, MR-MOEA can provide different choices of divisions in one run for decision makers. To illustrate this advantage, we present the final nondominated solutions and the divisions associated with some solutions of our algorithm on two networks, karate and dolphin, which are shown in Figs. 7 and 8, respectively. Fig. 7(a) shows all nondominated solutions found by MR-MOEA in one run for karate network, and the divisions associated with two nondominated solutions A and B in Fig. 7(a) are presented in Fig. 7(b) and (c), respectively. As can be seen in Fig. 7(b), for solution A, the network is divided into two communities without any overlapping node. For solution B, the network is divided into two communities with three overlapping nodes (nodes 1, 10, and 32). Similar results can also be found in Fig. 8 for dolphin network. Therefore, we can conclude that the proposed MR-MOEA can be used to detect the hierarchical community structure for a given network, which provides a variety of divisions under different scales for a given network.

C. Sensitivity Analysis of Parameter μ in MR-MOEA

As stated in Section III-A, there is one parameter μ for controlling the number of candidate overlapping nodes at the beginning of applying the proposed MR-MOEA. In the following, we investigate the influence of parameter μ on the performance of the proposed MR-MOEA.

Table IV presents the number of candidate overlapping nodes on ten real networks under different μ values increasing from 0 to 0.4 at the interval of 0.1. As can be seen, the number of candidate overlapping nodes is closely related to the value of μ. A larger value of μ will lead to have more candidate overlapping nodes. Take the network football as an example, the number of candidate overlapping nodes is 0 for $\mu = 0$, and increases to 32 when $\mu = 0.4$. Fig. 9 presents the Q_{ov} value of MR-MOEA on ten real networks under different values of μ increasing from 0 to 0.4 at the interval of 0.1. As

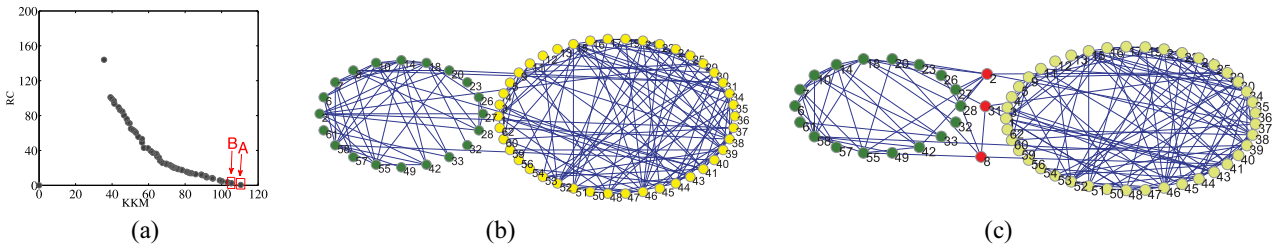


Fig. 8. Community structures obtained by MR-MOEA on network *dolphin*. Nodes with the same color are from the same community and the red ones are overlapping nodes between different communities. (a) Nondominated solutions in final population. (b) Community structure of the solution A in (a). (c) Community structure of the solution B in (a).

TABLE IV
NUMBER OF CANDIDATE OVERLAPPING NODES OBTAINED BY ALGORITHM 1 UNDER DIFFERENT μ AND THE AVERAGE NUMBER OF OVERLAPPING NODES ON NONDOMINATED SOLUTIONS OBTAINED BY ALGORITHM 2 WITH $\mu = 0.1$ ON TEN REAL NETWORKS

Network	Number of nodes in network G	Number of candidate overlapping nodes on different μ					Average number of overlapping nodes on obtained non-dominated solutions with $\mu = 0.1$
		0	0.1	0.2	0.3	0.4	
karate	34	2	3	3	3	3	1.48
dolphin	62	8	14	15	16	16	5.17
football	115	0	19	26	30	32	9.76
polbook	105	0	11	15	18	19	6.74
Yeast-D2	1443	24	131	174	209	226	67.7
Y2H	1966	497	501	502	502	502	178.3
jazz	198	1	41	65	79	87	20.78
netscience	1589	25	86	99	103	103	41.39
PPI	2445	412	526	538	566	570	195.43
blogs	3982	421	505	518	539	547	154.65

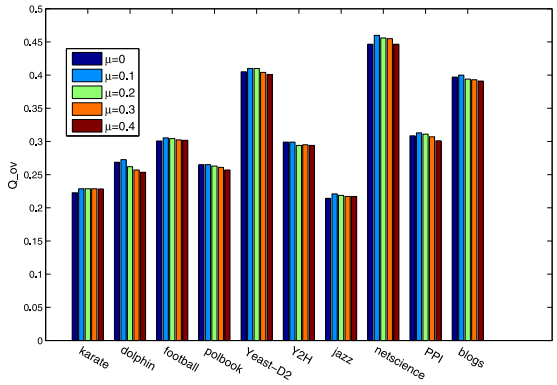


Fig. 9. Q_{ov} values of MR-MOEA under different parameter values of μ on ten real networks.

can be observed, the performance of MR-MOEA is not very sensitive to the value of μ, despite that there is a little influence. On the ten real networks, the Q_{ov} value of MR-MOEA rises from $\mu = 0$, reaches the top when $\mu = 0.1$, and then drops as the value of μ increases. This observation suggests that $\mu = 0.1$ is a generally good setting for real networks with different characteristics. Hence, the μ is always set to 0.1 in our experiments.

Table IV also shows the average number of overlapping nodes for all nondominated solutions obtained by the proposed MR-MOEA on ten real networks when $\mu = 0.1$. From the table, it can be seen that the number of final overlapping nodes obtained by MR-MOEA is much smaller than that of candidate overlapping nodes detected at the beginning of MR-MOEA, which demonstrates that the MR-MOEA can identify the true overlapping nodes from the candidate ones during the evolution, thereby confirming the effectiveness of the evolutionary operators in MR-MOEA.

From the above empirical results, we can conclude that the proposed MR-MOEA is a competitive and promising method for overlapping community detection.

V. CONCLUSION

In this paper, we have proposed an MR-MOEA for detecting overlapping communities. In the proposed algorithm, a mixed individual representation has been developed to fast encode and decode for overlapping communities. This mixed representation consists of candidate overlapping node-based representation and nonoverlapping node-based representation, and different individual updating strategies have been proposed for overlapping nodes and nonoverlapping nodes. Experimental results on ten real-world networks indicate that the proposed MR-MOEA is superior over 6 representative algorithms, including three non-MOEA-based algorithms and three MOEA-based algorithms for overlapping community detection.

The work in this paper has shown the effectiveness of the mixed individual representation suggested in the proposed MR-MOEA. In the future, we would like to further explore this encoding scheme for overlapping community detection in complex networks. It is also interesting to consider this encoding scheme in designing promising MOEAs to solve other kinds of complex networks, such as signed networks [29] and dynamic networks [49]. In addition, many promising MOEAs recently developed in evolutionary computation community also deserve to be explored in designing MOEA-based community detection algorithms for complex networks, such

as large-scale many-objective evolutionary algorithm [50], knee point-driven evolutionary algorithm [51], non-dominated sorting genetic Algorithm-III [52], inverse modeling multi-objective evolutionary algorithm [53], and adaptive cross-generation differential evolution operators for multi-objective optimization algorithm [54].

REFERENCES

[1] S. Wasserman and K. Faust, *Social Network Analysis Methods and Applications* (Structural Analysis in the Social Sciences), vol. 24. New York, NY, USA: Cambridge Univ. Press, 2015, pp. 219–220.

[2] C. Pizzuti and S. E. Rombo, "Algorithms and tools for protein-protein interaction networks clustering, with a special focus on population-based stochastic methods," *Bioinformatics*, vol. 30, no. 10, pp. 1343–1352, 2014.

[3] R. Pastor-Satorras and A. Vespignani, *Evolution and Structure of the Internet: A Statistical Physics Approach*. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[4] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[5] R. Albert, H. Jeong, and A.-L. Barabási, "The Internets achilles heel: Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, 2000.

[6] A. D. King, N. Przulj, and I. Jurisica, "Protein complex prediction via cost-based clustering," *Bioinformatics*, vol. 20, no. 17, pp. 3013–3020, 2004.

[7] P. D. Meo, E. Ferrara, G. Fiumara, and A. Provetti, "Mixing local and global information for community detection in large networks," *J. Comput. Syst. Sci.*, vol. 80, no. 1, pp. 72–87, 2014.

[8] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, no. 7307, pp. 761–764, 2010.

[9] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Nat. Acad. Sci. USA*, vol. 99, no. 12, pp. 7821–7826, 2002.

[10] Q. Cai, L. Ma, M. Gong, and D. Tian, "A survey on network community detection based on evolutionary computation," *Int. J. Bio Inspired Comput.*, vol. 8, no. 2, pp. 84–98, 2016.

[11] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proc. Nat. Acad. Sci. USA*, vol. 105, no. 4, pp. 1118–1123, 2008.

[12] P. Pons and M. Latapy, "Computing communities in large networks using random walks," in *Proc. 20th Int. Symp. Comput. Inf. Sci.*, 2005, pp. 284–293.

[13] B. Karrer and M. E. J. Newman, "Stochastic blockmodels and community structure in networks," *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, vol. 83, no. 1, pp. 211–222, 2011.

[14] J. Reichardt and S. Bornholdt, "Statistical mechanics of community detection," *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, vol. 74, no. 1, p. 016110, 2006.

[15] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

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

[17] W. Ren, G. Yan, X. Liao, and L. Xiao, "Simple probabilistic algorithm for detecting community structure," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 79, no. 3, 2009, Art. no. 036111.

[18] D. Jin, B. Gabrys, and J. Dang, "Combined node and link partitions method for finding overlapping communities in complex networks," *Sci. Rep.*, vol. 5, Feb. 2015, Art. no. 8600.

[19] Q. Chen, T. Wu, and M. Fang, "Detecting local community structures in complex networks based on local degree central nodes," *Phys. A Stat. Mech. Appl.*, vol. 392, no. 3, pp. 529–537, 2013.

[20] U. Brandes *et al.*, "On modularity—NP-completeness and beyond," 2006.

[21] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E*, vol. 69, no. 6, 2004, Art. no. 066133.

[22] M. Gong, L. Ma, Q. Zhang, and L. Jiao, "Community detection in networks by using multiobjective evolutionary algorithm with decomposition," *Phys. A Stat. Mech. Appl.*, vol. 391, no. 15, pp. 4050–4060, 2012.

[23] C. Shi, Z. Yan, Y. Cai, and B. Wu, "Multi-objective community detection in complex networks," *Appl. Soft Comput.*, vol. 12, no. 2, pp. 850–859, 2012.

[24] C. Pizzuti, "GA-Net: A genetic algorithm for community detection in social networks," in *Proc. Int. Conf. Parallel Problem Solving Nat.*, Dortmund, Germany, 2008, pp. 1081–1090.

[25] C. Shi, Z. Yan, Y. Wang, Y. Cai, and B. Wu, "A genetic algorithm for detecting communities in large-scale complex networks," *Adv. Complex Syst.*, vol. 13, no. 1, pp. 3–17, 2010.

[26] C. Pizzuti, "A multiobjective genetic algorithm to find communities in complex networks," *IEEE Trans. Evol. Comput.*, vol. 16, no. 3, pp. 418–430, Jun. 2012.

[27] M. Gong, Q. Cai, X. Chen, and L. Ma, "Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 82–97, Feb. 2014.

[28] J. Liu, W. Zhong, H. A. Abbass, and D. G. Green, "Separated and overlapping community detection in complex networks using multiobjective evolutionary algorithms," in *Proc. Congr. Evol. Comput.*, 2010, pp. 1–7.

[29] C. Liu, L. Jing, and Z. Jiang, "A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2274–2287, Dec. 2014.

[30] S. Fortunato and M. Barthelemy, "Resolution limit in community detection," *Proc. Nat. Acad. Sci. USA*, vol. 104, no. 1, pp. 36–41, 2007.

[31] X. Wen *et al.*, "A maximal clique based multiobjective evolutionary algorithm for overlapping community detection," *IEEE Trans. Evol. Comput.*, vol. 21, no. 3, pp. 363–377, Jun. 2017.

[32] Y. Li, Y. Wang, J. Chen, L. Jiao, and R. Shang, "Overlapping community detection through an improved multi-objective quantum-behaved particle swarm optimization," *J. Heuristics*, vol. 21, no. 4, pp. 549–575, 2015.

[33] L. Angelini, S. Boccaletti, D. Marinazzo, M. Pellicoro, and S. Stramaglia, "Identification of network modules by optimization of ratio association," *Chaos*, vol. 17, no. 2, 2007, Art. no. 023114.

[34] M. Tasgin, A. Herdagdelen, and H. Bingol, "Community detection in complex networks using genetic algorithms," *ArXiv Preprint ArXiv:0711.0491*, 2007.

[35] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, Orlando, FL, USA, 1997, pp. 4104–4108.

[36] T. Zhang and B. Wu, "A method for local community detection by finding core nodes," in *Proc. Int. Conf. Adv. Soc. Netw. Anal. Min.*, 2012, pp. 1171–1176.

[37] J. Yang and J. Leskovec, "Overlapping community detection at scale: A nonnegative matrix factorization approach," in *Proc. 6th ACM Int. Conf. Web Search Data Min.*, Rome, Italy, 2013, pp. 587–596.

[38] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropol. Res.*, vol. 33, no. 4, pp. 452–473, 1977.

[39] D. Lusseau, "The emergent properties of a dolphin social network," *Proc. Roy. Soc. B Biol. Sci.*, vol. 270, no. 2, pp. 186–188, 2003.

[40] M. E. J. Newman, "Modularity and community structure in networks," *Proc. Nat. Acad. Sci. USA*, vol. 103, no. 23, pp. 8577–8582, 2006.

[41] N. Zaki, J. Berengueres, and D. Efimov, "Prorank: A method for detecting protein complexes," in *Proc. Genet. Evol. Comput. Conf.*, Philadelphia, PA, USA, 2012, pp. 209–216.

[42] H. Yu *et al.*, "High quality binary protein interaction map of the yeast interactome network," *Science*, vol. 322, no. 5898, pp. 104–110, 2008.

[43] P. M. Gleiser and L. Danon, "Community structure in jazz," *Adv. Complex Syst.*, vol. 6, no. 4, pp. 565–573, 2003.

[44] M. E. Cusick *et al.*, "Literature-curated protein interaction datasets," *Nat. Methods*, vol. 6, no. 1, pp. 39–46, 2009.

[45] S. Gregory, "Finding overlapping communities in networks by label propagation," *New J. Phys.*, vol. 12, no. 10, pp. 2011–2024, 2009.

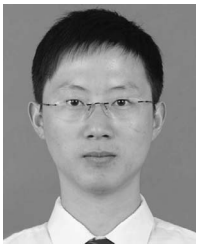
[46] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the definition of modularity to directed graphs with overlapping communitie," *J. Stat. Mech. Theory Exp.*, vol. 3, no. 3, pp. 459–480, 2009.

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

[48] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Comput. Surveys*, vol. 45, no. 4, pp. 1–35, 2013.

[49] F. Folino and C. Pizzuti, "An evolutionary multiobjective approach for community discovery in dynamic networks," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1838–1852, Aug. 2014.

[50] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization," *IEEE Trans. Evol. Comput.*, to be published.
[51] X. Zhang, Y. Tian, and Y. Jin, "A knee point-driven evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 761–776, Dec. 2015.
[52] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.
[53] R. Cheng, Y. Jin, K. Narukawa, and B. Sendhoff, "A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 838–856, Dec. 2015.
[54] X. Qiu, J.-X. Xu, K. C. Tan, and H. A. Abbass, "Adaptive cross-generation differential evolution operators for multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 232–244, Apr. 2016.



Lei Zhang received the B.Sc. degree from Anhui Agriculture University, Hefei, China, in 2007, and the Ph.D. degree from the University of Science and Technology of China, Hefei, in 2014.

He is currently a Lecturer with the School of Computer Science and Technology, Anhui University, Hefei. He has published more than 20 papers in refereed conferences and journals, such as ACM International SIG Conference on Knowledge Discovery and Data Mining, ACM International Conference on Information and Knowledge Management, the IEEE International Conference on Data Mining, ACM Transactions on Knowledge Discovery from Data, the *IEEE Computational Intelligence Magazine*, and *Information Sciences*. His current research interests include multiobjective optimization and applications, data mining, social network analysis, and pattern recommendation.

Dr. Zhang was a recipient of the ACM CIKM'12 Best Student Paper Award.

Hebin Pan received the B.Sc. degree from Anhui Architecture University, Hefei, China, in 2012. He is currently pursuing the master's degree with the School of Computer Science and Technology, Anhui University, Hefei.

His current research interests include multiobjective optimization methods and their application in complex network clustering.

Yansen Su received the B.Sc. degree from Tangshan Normal University, Tangshan, China, in 2007, the M.Sc. degree from the Shandong University of Science and Technology, Qingdao, China, in 2010, and the Ph.D. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2014.

She is currently a Lecturer with the School of Computer Science and Technology, Anhui University, Hefei. Her current research interests include complexes networks, computational biology, and multiobjective optimization.

Xingyi Zhang (M'15) received the B.Sc. degree from Fuyang Normal College, Fuyang, China, in 2003, and the M.Sc. and Ph.D. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2009, respectively.

He is currently a Professor with the School of Computer Science and Technology, Anhui University, Hefei, China. He has published over 40 papers in refereed conferences and journals, such as the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the *IEEE Computational Intelligence Magazine*, and *Information Sciences*. His current research interests include unconventional models and algorithms of computation, multiobjective optimization, and membrane computing.

Yunyun Niu received the B.S. degree in applied mathematics from Qufu Normal University, Jining, China, in 2004, the M.S. degree in electric machines and electric apparatus from the Zhengzhou University of Light Industry, Zhengzhou, China, in 2007, and the Ph.D. degree in systems analysis and integration from the Huazhong University of Science and Technology, Wuhan, China, in 2012.

She is currently an Associate Professor with the School of Information Engineering, China University of Geosciences, Beijing, China. Her current research interests include membrane computing and artificial intelligence.