IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 8, 2019, accepted August 28, 2019, date of publication September 2, 2019, date of current version September 18, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2938982

A Memetic Algorithm for Community Detection in Bipartite Networks

SHIWEI CHE, WU YANG, AND WEI WANG
Information Security Research Center, Harbin Engineering University, Harbin 150001, China

Corresponding author: Wu Yang (yangwu@hrbeu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61572459 and Grant 61672180.

ABSTRACT Bipartite networks belong to the category of complex networks, whose vertices can be divided into two separated vertex sets, so that there are no edges between vertices in the same set, and edges only exist between vertices in different sets. In the past decades, although community discovery in one-mode networks has been deeply explored, the detection of communities in bipartite networks has not been widely studied. In this paper, we present a new memetic algorithm named MATMCD-BN for community detection in bipartite networks with two types of nodes in the community. Firstly, we put forward a new initialization method for population initialization of memetic algorithm for bipartite network communities discovery, which can expedite the convergence speed of this algorithm. Secondly, besides using traditional mutation operator, we propose a new crossover operator (called two-way random crossover operator in this paper) and a new mutation operator (called mutation operator 2 in this paper), which are helpful to improve the accuracy of the solution and accelerate the convergence speed of the proposed algorithm. Finally, we develop a local search method, which can make the solution approach the global optimal solution quickly and jump out of the local optimal solution with a certain probability. As far as we know, the proposed MATMCD-BN is the first memetic algorithm (MA) applied to community detection in bipartite networks with two types of nodes in the community. In order to confirm the performance of this algorithm, we have done a lot of experiments using synthetic and real social networks. The experimental results demonstrate that the presented method is effective and promising for bipartite network community identification.

INDEX TERMS Genetic algorithm, social network, bipartite network, community detection.

I. INTRODUCTION

Many complex systems in the real world can be represented by complex networks [26], such as computer networks, information networks, collaborative networks, the Internet, the world wide web, technology networks, transportation networks. A complex network is usually composed of vertices (or nodes) and edges (or links). A node represents a component of a complex network, and an edge between two nodes stands for the interaction between two components of a complex network. Network features like small world phenomenon and scale-free characteristic have attracted much attention. In recent years, community structure as another important feature of the network has been widely studied [9]. At present, there is no uniform definition of community. This definition usually depends on the current application or the specific system to be processed. A common definition is that a community is a collection of vertices, in which vertices are closely connected, and between collections, vertices are sparsely connected [4].

So far, many algorithms and methods for community structure detection have been proposed. For example, splitting algorithm, agglomerative algorithm, optimization algorithm, random walks, statistical mechanics, spectral clustering and graph partitioning [11]. Pothen *et al.* presented a community structure discovery algorithm on the strength of hierarchical clustering for complex networks [13]. In the literature, one of the most famous optimization algorithms is the algorithm named GN presented by Girvan and Newman [9], [19]. It is a split hierarchical clustering algorithm on the strength of iterative deletion of network edges, which divides the entire network into several communities. In order to identify communities, many performance evaluation criteria have been developed. The modularity presented by Newman and Girvan

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

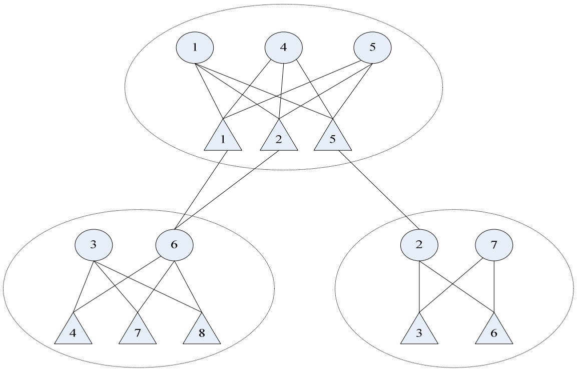


FIGURE 1. An example of a bipartite network with three communities.

is an important evaluate function to assess the accuracy of clustering in complex networks [19]. On the basis of the concept of modularity, community discovery can be considered as a modularity optimization problem. Based on modularity optimization, many community detection methods have been proposed [9]. They all encounter the resolution limit problem [17]. None of them can detect the communities in the network whose size is less than $\sqrt{2L}$, where L is the total number of edges in the network. The resolution limit problem can be settled by different community structure quality evaluation functions, like the density-based bipartite modularity function used in the algorithm in this paper.

As is known to all, individuals in social networks are often members of multiple communities. In [14] and [20], the author pointed out that overlap is a common feature of many real complex networks. Therefore, more and more researchers are interested in overlapping community detection. So far, many overlapping community detection algorithms have been presented. Most of them are node-based rather than link-based approaches. After the link-based community structure is obtained, the link-based community can be easily transformed into a node-based community by converting the edge into two nodes attached to it [16].

A number of complex systems in the real world can be modeled as bipartite networks. A bipartite network, also called a two-mode network, is comprised of two types of vertices, and edges can only connect different types of vertices. The authors of [19] believe that the communities of bipartite networks can only consist of identical types of vertices. However, the authors of [17] and [20] argued that the bipartite communities should be made up of two types of vertices, making the edges in the communities are denser than those between communities. Fig. 1 shows an example of a bipartite network with three communities.

Some algorithms have been presented to find communities included in bipartite networks. Until now, researchers have proposed two types of community detection algorithms for bipartite networks. One method is to first transform the initial bipartite network into a one-mode network by projection. After that, the mature unipartite network community detection methods are applied to find the communities on the projection network. However, the projection will lose some valuable information of the initial bipartite network. The second method directly analyses the original bipartite network and detects the communities on it.

Because the community detection problem is NP-hard, people have used various mature methods to solve this problem. Therefore, approximate algorithms such as swarm intelligence algorithms and evolutionary algorithms (EAs) are applied to community detection. Evolutionary algorithm is an algorithm inspired by the natural evolutionary process. Evolutionary algorithms use the evolutionary principles of the natural world to solve community detection problems, such as selecting the best individuals for the next generation population from the current population, performing crossover operations, and performing mutation operations. When selecting individuals, fitness functions are used to evaluate individuals. Genetic algorithm (GA) belongs to the category of evolutionary algorithm. It is an effective global search technology, but it often spends a lot of time to converge to the global optimal solution. Genetic algorithms use selection, crossover and mutation operators to optimize a solution population.

An evolutionary algorithm which combines a genetic algorithm with a local search process is called memetic algorithm (MA). Because MA introduces local search into genetic algorithm, it solves the problem of high computational complexity of genetic algorithm, and also overcomes the problem of easily falling into local optimum. Compared with the algorithm for searching the exact best solution, the genetic algorithm is more suitable for finding the good approximate solutions rapidly, and local search is more efficient for finding the exact best solution. Local learning further optimizes some of the best solutions in the population, thus speeding up the convergence rate of the population. For solving combinatorial optimization problems, MA is a very powerful algorithm. MA is suitable for solving NP-hard community detection problem. Many precedents have shown that MA has been successfully applied to this problem. Since finding the optimal density-based bipartite modularity is an NP-hard problem, we use MA to optimize the fitness function. In the process of discovering community structure, for optimize the fitness of chromosomes in the population of genetic algorithm and speed up the convergence of the algorithm, it is necessary to move nodes to the most suitable community. In order to find the most suitable community for each vertex, we only need to check the vertex's direct neighbor vertex. Local search combines knowledge of community discovery with a genetic algorithm, which is why the memetic algorithm is more effective than the genetic algorithm in the field of community discovery. Due to the local search function can accelerate the convergence of the memetic algorithm, it is vital for the performance of the memetic algorithm.

This paper presents a memetic algorithm for two-mode community (i.e., containing two types of nodes in the community) discovery in bipartite networks. The main contributions of this paper are summarized as follows.

1. As far as we know, the proposed MATMCD-BN is the first MA algorithm applied to community detection in bipartite networks with two types of nodes in the community.
2. For accelerating the convergence rate (reduce the numbers of iteration), a new memetic algorithm population initialization mechanism is proposed.
3. We propose a new crossover operator (named two-way random crossover operator in this paper). It can better inherit the genetic characteristics of parent chromosomes, improve the quality of solution, and accelerate the convergence speed of the algorithm.
4. In addition to using the traditional mutation operator, we also propose a new mutation operator (named mutation operator 2 in this paper). Similarly, it can improve the quality of the solution and accelerate the convergence of the algorithm. Moreover, it can significantly improve the diversity of the population.
5. We also propose a local search function to make the best solution of the child population of the algorithm closer to the global best solution in the solutions space. This function can make the algorithm jump out of local optimum and achieve global optimum with a certain probability.

To check the performance of MATMCD-BN, we conducted a lot of experiments on five synthetic and six real-world social networks. We also compare MATMCD-BN with three existing famous bipartite network community detection methods. The experimental results show that the presented method is superior to the existing methods.

The rest of this paper is organized as follows: Section II states the related work of our research. Section III gives some basic concepts and knowledge related to our research. In Section IV, the proposed MATMCD-BN algorithm for community discovery in the bipartite network is presented. This section describes in detail the memetic algorithm population representation, population initialization, fitness function, chromosome selection algorithm, crossover operator, mutation operator, local search function and so on. The experimental results on synthetic and real networks are shown in Section V. Finally, Section VI gives some conclusions.

II. RELATED WORKS

The main goal of community discovery is to partition the network into communities. A large number of algorithms based on different disciplines, such as mathematics, biology, computer science and sociology, have been presented to discover communities in the network [9], [15], [29], [30]–[32].

In the past decades, with the emergence of a good deal of community discovery problems, many methods have been presented to solve community discovery problems. Newman and Girvan proposed a splitting algorithm named GN method. Newman also presented an algorithm using modularity Q on the strength of GN algorithm, which belongs to the category of agglomeration algorithms [31]. This algorithm is named FN algorithm. In FN algorithm, at first, each vertex in the network forms a unique community. Then, at each step, the algorithm iteratively merges the community pairs with the greatest modularity gain. Based on the optimization of modularity, Clauset *et al.* [34] presented an algorithm to discover communities in networks, named CNM. Compared with FN algorithm, this algorithm is faster and suitable for detecting the community structure of large scale networks. In [35], Newman also presented a spectral algorithm using modularity matrix.

As described in section I, the resolution limit problem is a common defect of modularity optimization-based algorithms. By using different quality metric functions, many algorithms have been presented to settle this problem [36], [29]. The density-based bipartite modularity function Q_D introduced in section IV-C of this paper is one of these functions. In the literature, another existing method to overcome the resolution limit problem is to transform the community discovery problem into a multi objective optimization problem [37], [38]. Multi objective optimization algorithm seeks the best solution of the problem by optimizing multiple optimization functions (also known as objective functions) simultaneously. These optimization functions assess the community partition found from different perspectives.

Shi *et al.* [37] presented a multi objective evolutionary algorithm named MOCD, which is used in community discovery of complex networks. In the MOCD algorithm, two model choice methods are used to choose better solutions from Pareto optimal set, which can produce more precise network partition.

Arenas *et al.* [36] presented a multi objective evolutionary algorithm MOEAD-Net to solve community discovery problems of complex networks. This algorithm optimizes two contradictory optimization functions at the same time. In [40], an algorithm MOEA/D-Net which also belongs to the category of multi objective evolutionary algorithm (MOEA) is proposed. The experimental consequences in [40] prove that MOEA/D-Net is a promising and high-efficiency algorithm for accurately discovering network community structures. In the literature, two other MOEA algorithms have been also proposed, named MODTLBO/D [41] and MODBSA/D [42].

Pizzuti [39] presented a genetic algorithm named GA-Net to detect community partitions in networks. This algorithm identifies communities in the network by optimizing an objective function. Because the mutation operators in GA-Net only consider the real connections between nodes, the algorithm is fast.

Pizzuti [38] also developed a network community structure identification algorithm MOGA-Net, which belongs to the category of multi objective genetic algorithm. MOGA-Net optimizes two optimization functions at the same time, namely community score and community fitness. MOGA-Net can generate a hierarchical structure of communities. On the basis of modularity and an improved genetic algorithm, Shang *et al.* [44] presented a community discovery method named MIGA. MIGA uses modularity as the optimization function, which simplifies this algorithm. In addition, MIGA uses prior information (the number of communities) in the initialization process of the algorithm, which enhances the stability and accuracy of the algorithm.

Based on genetic algorithm, a memetic algorithm named Meme-Net was proposed by the authors of [46]. Its local search function used a hill climbing strategy. This algorithm optimizes the modularity density function, which contains an adjustable parameter to enable one to find community partitions at different resolutions. In [60], Li *et al.* presented a new link-based community discovery algorithm called Meme-Link. Meme-Link first transforms the original network into the corresponding weighted line graph. Then, the disjoint communities based on links are detected on the line graph. Finally, the link-based disjoint communities of the line graph are transformed into overlapping communities of the original network. Based on node-entropy learning procedure, Žalik *et al.* [47] presented a community detection algorithm, called Node Entropy MA for Networks (NE-Net). NE-Net uses modularity as the objective function and adopts two new genetic operators: modularity based group crossover and mutation. A modified version of NE-Net, called Entropy based MA for Networks (E-Net), was also proposed by Žalik et al. In order to discover communities in the networks, Mu *et al.* [48] proposed a memetic algorithm called MA-SAT. MA-SAT uses simulated annealing (SA) and tightness greedy optimization (TGO) as two local search procedures. SA can avoid the algorithm falling into local optimum, while TGO is beneficial to the diversity of the population. In [49], Said *et al.* developed a multi objective memetic algorithm for community discovery. In [50], a modularity based memetic algorithm for undirected and unweighted networks community detection is proposed. In [61], a memetic particle swarm optimization algorithm for networks community discovery is proposed, which is called MPSOA. In [51], Mirsaleh and Meybodi proposed an algorithm called MLAMA-NET based on Michigan memetic algorithm to solve community discovery problems.

Generally, community detection algorithms in bipartite networks can be fallen into two categories: one-mode projection algorithms and direct processing algorithms for bipartite networks [33]. In [21], a weighted projection method is proposed, which projects the bipartite network into a unipartite network, and then detects the community using the classical one-mode community discovery algorithm. In order to discover one-mode communities in bipartite networks, Cui *et al.* [22] proposed an algorithm. Firstly, this algorithm projects the initial bipartite network to a one-mode network. Then, an algorithm based on weighted clustering threshold is used to find one-mode community. This algorithm can identify overlapping nodes.

For the second bipartite network community detection algorithm, the optimization algorithm is usually applied to find the communities. In [24], Chen *et al.* proposed a bipartite modularity extended from unipartite modularity, and then proposed a BRIM algorithm to detect communities directly on bipartite networks. In [25], the BRIM algorithm is extended and an algorithm combining label propagation (LP) and BRIM is proposed, which is called LP BRIM. Based on distance dynamics, Sun *et al.* [59] proposed a new method to discover two-mode communities in large bipartite networks. This method is inspired by the interaction between people in human society. Zhou *et al.* [27] developed a community detection algorithm for bipartite networks. It detected bipartite communities using the formula of bipartite modularity gain. Based on two asymmetric parameters, Wang *et al.* [5] proposed a bipartite network community discovery algorithm. This algorithm can detect overlapping community partitions. On the basis of the concept of average bipartite modularity degree, Xu *et al.* [55] developed a quantitative function to assess the community partitions in bipartite networks. Li *et al.* [10] developed a quantitative measure named bipartite partition density to assess the community partitions in bipartite networks. They also developed an algorithm called BiLPA to find overlapping community structures in bipartite networks using bipartite partition density. Cui *et al.* [12] presented an algorithm to identify overlapping community structures directly in bipartite networks. Based on memetic algorithm, Wang *et al.* [52] presented an algorithm called MACD-BN to find communities in bipartite networks. However, MACD-BN can only detect communities that contain one type of node, that is, one-mode rather than two-mode communities.

In this paper, the MATMCD-BN algorithm we developed belongs to the second category. It is a memetic algorithm that combines genetic algorithm with a local search function. Unlike MACD-BN, it can detect two-mode communities. And two-mode community detection is more in line with the needs of many practical applications.

III. BASIC CONCEPTS AND KNOWLEDGE

A. DEFINITION OF BIPARTITE NETWORK AND ITS COMMUNITY

A bipartite network can be modeled as an undirected graph $\mathrm{G} = (\mathrm{V}, \mathrm{E})$, where V is the set of nodes or vertices and $\mathrm{V} = \{\mathrm{U} \cup \mathrm{W}\}$, U and W represent the nodes of type-U and type-W respectively, E is the set of edges or links. $\mathrm{U} \cap \mathrm{W} = \emptyset$, $|\mathrm{U}| = \mathrm{p}$, $|\mathrm{W}| = \mathrm{q}$, and $\mathrm{p} + \mathrm{q} = \mathrm{n}$. Edge $\mathrm{e_{ij}}$ can only connect different types of nodes, that is, $\mathrm{e_{ij}} \in \mathrm{E}\left(\mathrm{u_i} \in \mathrm{U}, \mathrm{w_j} \in W\right)$. $|\mathrm{E}| = \mathrm{m}$ is the number of edges in a bipartite network G. The adjacency matrix A of G can be expressed as follows:

$$\mathrm{A} = \begin{vmatrix} 0_{p*p} & A'_{p*q} \\ A'^{T}_{q*p} & 0_{q*q} \end{vmatrix}$$

where 0_{p*p} and 0_{q*q} are all-zero matrices, and A'_{p*q} is a non-zero matrix. A'_{p*q}, a simplified version of A, can also be used to represent the adjacency matrix of G. In A'_{p*q}, the row stands for the nodes of U and the column stands for the nodes of W.

Community discovery in a bipartite graph $\mathrm{G} = (\mathrm{V}, \mathrm{E}) = (\mathrm{U} \cup \mathrm{W}, \mathrm{E})$ is executed to partition G into s sub-graphs

$G_i = (U_i \cup W_i, E_i)$, $i = 1, 2, \ldots, s$, where s is the number of the communities, $U_i \subset U, W_i \subset W, \cup_{i=1}^{s} U_i = U$ and $\cup_{i=1}^{s} W_i = W$. This paper studies the community structure of connected bipartite networks.

B. A BIPARTITE MODULARITY FORMULA FOR BIPARTITE NETWORKS

Here, we will introduce a bipartite modularity formula of bipartite network used in experiments later in this paper. It is defined as follows:

$$Q_b = \frac{1}{m}\sum_{i=1}^{p}\sum_{j=1}^{q}\left(A'_{ij} - \frac{d_i g_j}{m}\right)\delta\left(r_i, s_j\right) \tag{1}$$

where d_i is the degree of the ith type-U node, g_j is the degree of the jth type-W node, and r_i, s_j represent the communities to which node i and j are assigned. When $r_i = s_j$, $\delta\left(r_i, s_j\right) = 1$, otherwise, $\delta\left(r_i, s_j\right) = 0$. See section III-A for the meaning of p, q, A'_{ij} and m.

IV. PRESENTED ALGORITHM

In this part, we will present a memetic algorithm for two-mode community detection in bipartite networks, referred to as MATMCD-BN. Firstly, the representation method of individuals in a population are given. At the same time, we propose a new population initialization method, which is helpful to accelerate the convergence of the population. Then, a fitness function for evaluating individuals in a population and a selection operator for reproductive operations are introduced. Next, a crossover operator and two mutation operators used in MATMCD-BN are proposed. Finally, a local search function proposed by us is given. The following sections will discuss the above contents in detail. The flow chart of MATMCD-BN algorithm is shown in Fig. 2.

A. INDIVIDUAL REPRESENTATION

In memetic algorithm, each possible community partition is represented by an individual, also known as a chromosome or a solution. A set of individuals is called a population of memetic algorithm, that is, population $P = \{C_1, C_2, \ldots, C_M\}$, where C_i is the ith individual in the population and M is the size of the population. Classical individual representation includes locus-based representation and string-based representation. This paper uses string-based representation. The ith individual in the population can be expressed as: $C_i = [g_1, g_2, \ldots, g_n]$. Here, g_j is the jth gene of chromosome C_i, and n is the number of nodes in the network. Each gene may take a value in the range $\{1, 2, \ldots, p\}$, where p is the number of nodes in node set U of bipartite network G. This is determined by our proposed initialization algorithm. Genes represent the nodes in graph $G = (V, E)$ that models a network, and the value assigned to the ith gene stands for the community containing node i. In this representation, if the gene $g_i = g_j$, it means that nodes i and j belong to the same community.

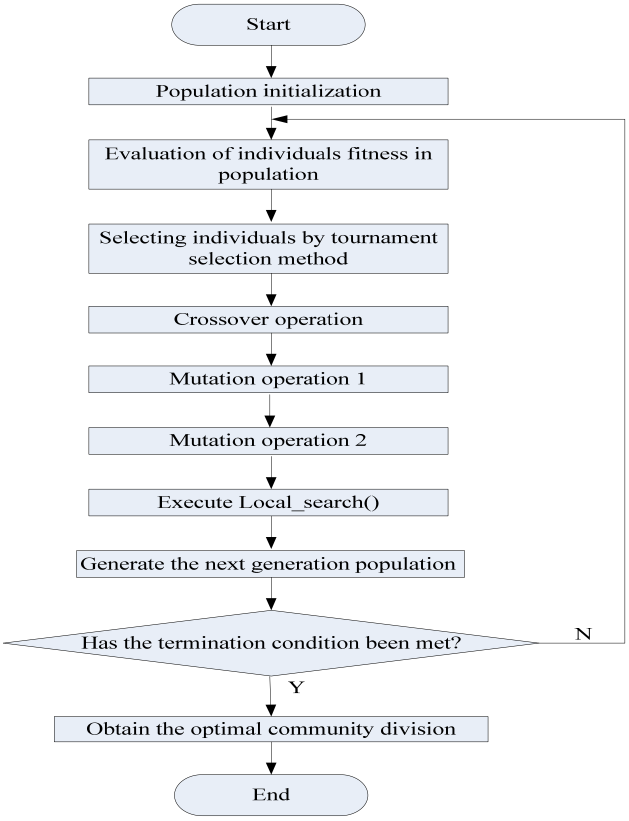

Start
Population initialization
Evaluation of individuals fitness in population
Selecting individuals by tournament selection method
Crossover operation
Mutation operation 1
Mutation operation 2
Execute Local_search()
Generate the next generation population
Has the termination condition been met?
N
Y
Obtain the optimal community division
End

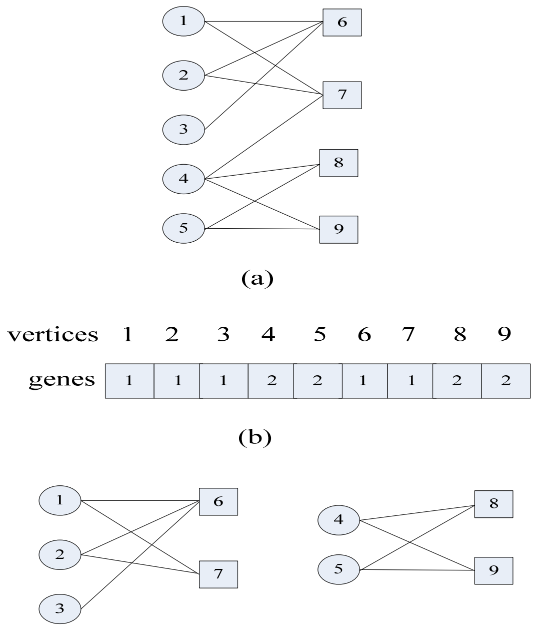
FIGURE 2. The flow chart of MATMCD-BN Algorithm.

Fig. 3(b) shows a string-based representation of the graph shown in Fig. 3(a). The bipartite network is made up of eight nodes numbered from 1 to 8. As shown in Fig. 3 (b), the gene values of nodes 1, 2, 3, 6 and 7 are all 1, while those of nodes 4, 5, 8 and 9 are all 2. This means that the network consists of two communities, where nodes 1, 2, 3, 6, 7 belong to one community and nodes 4, 5, 8, 9 belong to another community. Fig. 3 (c) shows the community structure transformed by chromosomes in Fig. 3 (b).

B. POPULATION INITIALIZATION

For the development of high quality MA, the generation of the initial population is of great significance, because the characteristics of the initial individuals will affect the accuracy of the final result of the algorithm and the convergence rate of the algorithm. Therefore, we present a new population initialization algorithm to enhance the quality of the initial population and accelerate the convergence rate of the algorithm. The initialization process of the proposed MATMCD-BN algorithm is shown in algorithm 1.

In algorithm 1, the outer loop of line 1 to 12 is responsible for generating the initial population. The inner loop of lines 2 to 8 is responsible for generating a chromosome. In line 4, a random community label is initialized for the 1 to p nodes in the chromosome. In line 6, the labels of p+1 to p+q nodes are initialized to the community labels owned by most of their neighbors. The inner loop of lines 9 to 11 is responsible for



(a)

(b)

(c)

FIGURE 3. (a) Simple bipartite network consisting of 8 vertices; (b) string-based representation of one chromosome; (c) graph representation of the chromosome shown in (b).

Algorithm 1 Pseudocode of Initialization Step of MATMCD-BN Algorithm

Parameters: population size popsize;
Input: a adjacent matrix A of a bipartite networks;
Output:initial population;

```
1:  for i = 1; i <= popsize; i + + do
2:      for j = 1; j <= n; j + + do
3:          if j <= p then
4:              pop[i][j] = A random integer in the range of 1-p
                generated randomly;
5:          else
6:              pop[i][j] = The label shared by most of the
                neighbor nodes of node j. If there are more than
                one label satisfy the above condition, randomly
                select one of them.
7:          end if
8:      end for
9:      for j = 1; j <= p; j + + do
10:         pop[i][j] = The label shared by most of the neighbor
            nodes of node j. If there are more than one label
            satisfy the above condition, randomly select one
            of them.
11:     end for
12: end for
13: return pop[];
```

modifying the community labels of nodes 1 to p in chromosomes to those owned by most of their neighbors. Finally, in line 13, the algorithm returns the generated population.

We compare our initialization method with two commonly used population initialization methods. One is the random initialization method, the other is the initialization method using the label propagation algorithm. Detailed descriptions of these two methods are presented in the appendix. The experimental results on various real data sets show that the Q_b values of the optimal individuals of the initial population obtained by algorithm 1 are significantly improved compared with the other two algorithms in the appendix. Compared with algorithm 5, the increase is about 35%. Compared with algorithm 6, the increase is about 28%. For the final convergence time of the algorithm, the convergence time of MATMCD-BN using algorithm 1 as the initialization method has also been significantly reduced. Compared with algorithm 5, the reduction is about 38%. Compared with algorithm 6, the reduction is about 25%.

C. FITNESS FUNCTION

In this subsection, we will introduce a new evaluation function called density-based bipartite modularity presented by Xu et al. [55] to evaluate the good and bad of community partitions in bipartite networks. It is defined as follows:

$$Q_D = \sum_{i=1}^{s} \frac{D\left(U_{C_i}, W_{C_i}\right) - D\left(U_{C_i}, \overline{W}\right) - D(\overline{U}, W_{C_i})}{\left|U_{C_i}\right| \times \left|W_{C_i}\right|} \tag{2}$$

where s is the total number of communities in the bipartite network, C_i is the ith community in community partition. $D\left(U_{C_i}, W_{C_i}\right) = \sum_{j \in U_{C_i}} \sum_{k \in w_{C_i}} A'_{jk}$is the number of edges in community C_i. $D\left(U_{C_i}, \overline{W}\right) = \sum_{j \in U_{C_i}} \sum_{k \in \overline{W}} A'_{jk}$ is the number of edges between the nodes of type U in community C_i and all nodes of type W outside C_i.$\overline{W} = W - W_{C_i}$. Similarly $\overline{U}$= $U - U_{C_i}$. $\left|U_{C_i}\right|$ is the number of nodes of type U contained in C_i. $\left|W_{C_i}\right|$ is the number of nodes of type W contained in C_i. The bigger the value of Q_D is, the higher the quality of community partition. Xu et al. claim that this metric is correlated with the density of connections in the community, thus overcoming the resolution limit problem.

D. SELECTION ALGORITHM

In order to select chromosomes for crossover and mutation operations, we need to give a selection algorithm. The selection methods of traditional MA include random selection method, tournament selection method and roulette selection method. In order that individuals with fitness values that are not as high as the fitness values of the fittest individuals can also exist in the next generation of population, we chose an algorithm that kept elitism controlled. This method is called the tournament selection method. The most prominent feature of this method is that individuals in the population have equal opportunities to be selected for subsequent genetic operations. The implementation steps of this method are as follows: firstly, k ($k < M$, M is the size of the population) individuals are randomly selected from the population set, and then an individual with the greatest fitness value is selected from the k individuals as an individual in the selected set of individuals, repeating the above process until the required number of chromosomes is obtained.

E. CROSSOVER

Crossover operations use very simple methods to generate new individuals. The crossover operator acts on two parent individuals at the same time, and generates new individuals by exchanging information between the parent individuals. Therefore, the individuals generated by crossover operation have the characteristics of two parent individuals at the same time [53]. The process of crossover operation is usually split into three stages: (1) selecting two individuals from the parent population; (2) combining them according to certain rules; (3) outputting two new individuals. A crossover position is randomly determined. Then, based on the pre-specified probability of execution crossover, the corresponding fragments of the two parent individuals around this position are exchanged. Classical crossover operators include: one-point crossover, the two-point crossover, the uniform crossover, the one-way crossover, the two-way crossover, and so on.

In order to make full use of the community structure information of the parent chromosomes, improve the quality of the descendants generated by crossover operator and accelerate the convergence of population, we propose a new crossover operator, which is called two-way random crossover operator. This operator first marks the two chromosomes selected by the tournament selection method as the primary chromosome and the secondary chromosome respectively. Then, an offspring chromosome is generated by the following process.

(1) Mark all the genes of the primary chromosome and the secondary chromosome as unvisited, and set the counter $t = 1$.

(2) Randomly select an unvisited gene in the primary chromosome, find all the gene positions with the same gene value as the selected gene in the primary chromosome, and then fill the current t value into all the corresponding gene positions of the offspring chromosome. If the position of the gene to be filled has been filled by previous operations, it will not be changed and its original value will be retained. In the primary chromosome, all genes with the same value as the selected gene are marked as visited. If all the gene positions to be filled have been filled by previous operations, another unvisited gene is randomly selected from the primary chromosome to perform the above operations.

(3) Let $t = t + 1$.

(4) Randomly select an unvisited gene in the secondary chromosome, find all the gene positions with the same gene value as the selected gene in the secondary chromosome, and then fill the current t value into all the corresponding gene positions of the offspring chromosome. If the position of the gene to be filled has been filled by previous operations, it will not be changed and its original value will be retained. In the secondary chromosome, all genes with the same value as the selected gene are marked as visited. If all the gene positions to be filled have been filled by previous operations, another unvisited gene is randomly selected from the secondary chromosome to perform the above operations.

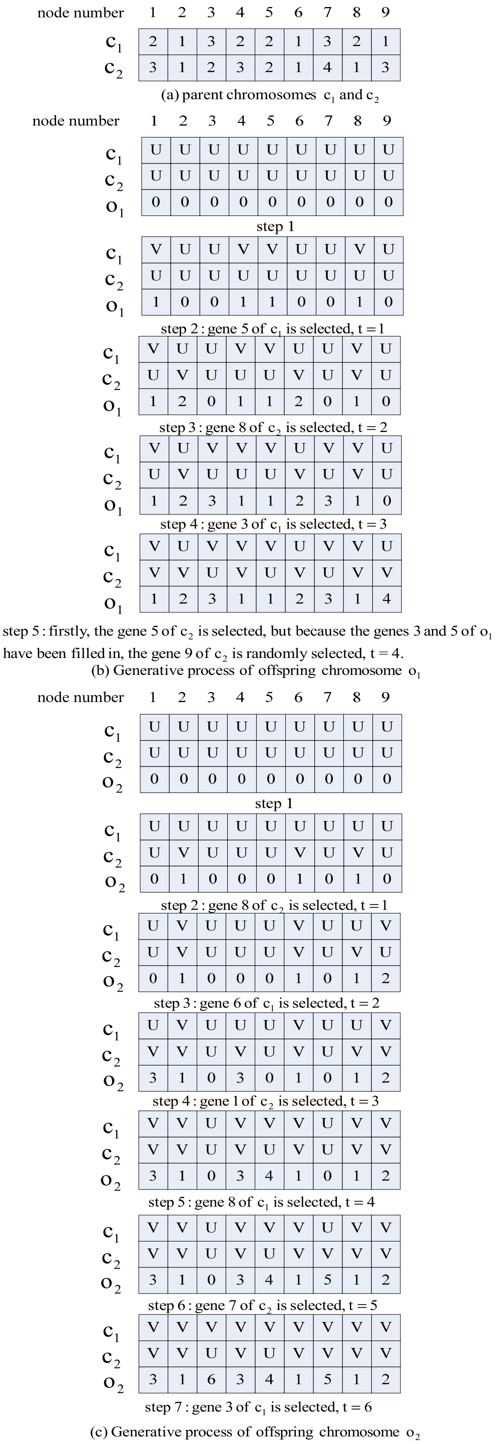
(5) Let $t = t + 1$.

(6) Step (2) - (5) are performed iteratively until the gene positions of the offspring chromosome are filled. In this way, we get an offspring chromosome.

Next, the primary chromosome and secondary chromosome are exchanged. Then, the above steps (1)-(6) are performed again on the two exchanged chromosomes to obtain another offspring chromosome. Fig. 4 shows a schematic diagram of the crossover operation of the MATMCD-BN algorithm.

In Fig. 4, c_1 and c_2 are two parent chromosomes selected from the previous generation population using the tournament selection algorithm. First, let c_1 be the primary chromosome and c_2 be the secondary chromosome. The process of obtaining an offspring chromosome o_1 from the parent chromosomes c_1 and c_2 are as follows. Step 1: Mark all genes of the primary chromosome c_1 and secondary chromosome c_2 as unvisited. Step 2: Randomly select an unvisited gene 5 in the primary chromosome c_1, and find all genes with the same gene value as gene 5 in c_1, i.e. genes 1, 4, 5 and 8. Then, let $t = 1$, and fill the current t value into the genes 1, 4, 5 and 8 of the offspring chromosome o_1. Subsequently, genes 1, 4, 5 and 8 in c_1 are marked as visited. Step 3: Randomly select an unvisited gene 8 in the secondary chromosome c_2, and find all genes with the same gene value as gene 8 in c_2, i.e. genes 2, 6 and 8. Then, let $t = t + 1 = 2$, and fill the current t value into the genes 2 and 6 of the offspring chromosome o_1. Because gene 8 of o_1 has been filled by the previous operation, this operation does not fill it. Subsequently, genes 2, 6, and 8 in c_2 are marked as visited. Step 4: Randomly select an unvisited gene 3 in the primary chromosome c_1, and find all genes with the same gene value as gene 3 in c_1, i.e. genes 3 and 7. Then, let $t = t + 1 = 3$, and fill the current t value into the genes 3 and 7 of the offspring chromosome o_1. Subsequently, genes 3 and 7 in c_1 are marked as visited. Step 5: Randomly select an unvisited gene 5 in the secondary chromosome c_2, and find all genes with the same gene value as gene 5 in c_2, i.e. genes 3 and 5. Because the genes 3 and 5 of o_1 have been filled, another unvisited gene 9 is randomly selected in c_2, and all genes with the same gene value as gene 9, namely genes 1, 4 and 9, are found in c_2. Then, let $t = t + 1 = 4$, and fill the current t value into the genes 9 of the offspring chromosome o_1. Because genes 1 and 4 of o_1 has been filled by the previous operation, this operation does not fill it. Subsequently, genes 1, 4, and 9 in c_2 are marked as visited. Because chromosome o_1 has been filled at this time, the process of generating offspring chromosome o_1 is over, and we get offspring chromosome o_1. Now, let c_2 be the primary chromosome and c_1 be the secondary chromosome to produce another offspring chromosome o_2. The generative process of o_2 is similar to that of o_1. The detailed steps are shown in Fig. 4 (c).

We compare our two-way random crossover operator with three famous crossover operators (i.e., uniform crossover



node number	1	2	3	4	5	6	7	8	9
c_1	2	1	3	2	2	1	3	2	1
c_2	3	1	2	3	2	1	4	1	3

(a) parent chromosomes c_1 and c_2

node number	1	2	3	4	5	6	7	8	9
c_1	U	U	U	U	U	U	U	U	U
c_2	U	U	U	U	U	U	U	U	U
o_1	0	0	0	0	0	0	0	0	0

step 1

c_1	V	U	U	V	V	U	U	V	U
c_2	U	U	U	U	U	U	U	U	U
o_1	1	0	0	1	1	0	0	1	0

step 2 : gene 5 of c_1 is selected, $t=1$

c_1	V	U	U	V	V	U	U	V	U
c_2	U	V	U	U	U	V	U	V	U
o_1	1	2	0	1	1	2	0	1	0

step 3 : gene 8 of c_2 is selected, $t = 2$

c_1	V	U	V	V	V	U	V	V	U
c_2	U	V	U	U	U	V	U	V	U
o_1	1	2	3	1	1	2	3	1	0

step 4 : gene 3 of c_1 is selected, $t = 3$

c_1	V	U	V	V	V	U	V	V	U
c_2	V	V	U	V	U	V	U	V	V
o_1	1	2	3	1	1	2	3	1	4

step 5 : firstly, the gene 5 of c_2 is selected, but because the genes 3 and 5 of o_1 have been filled in, the gene 9 of c_2 is randomly selected, $t = 4$.

(b) Generative process of offspring chromosome o_1

node number	1	2	3	4	5	6	7	8	9
c_1	U	U	U	U	U	U	U	U	U
c_2	U	U	U	U	U	U	U	U	U
o_2	0	0	0	0	0	0	0	0	0

step 1

c_1	U	U	U	U	U	U	U	U	U
c_2	U	V	U	U	U	V	U	V	U
o_2	0	1	0	0	0	1	0	1	0

step 2 : gene 8 of c_2 is selected, $t=1$

c_1	U	V	U	U	U	V	U	U	V
c_2	U	V	U	U	U	V	U	V	U
o_2	0	1	0	0	0	1	0	1	2

step 3 : gene 6 of c_1 is selected, $t=2$

c_1	U	V	U	U	U	V	U	U	V
c_2	V	V	U	V	U	V	U	V	V
o_2	3	1	0	3	0	1	0	1	2

step 4 : gene 1 of c_2 is selected, $t=3$

c_1	V	V	U	V	V	V	U	V	V
c_2	V	V	U	V	U	V	U	V	V
o_2	3	1	0	3	4	1	0	1	2

step 5 : gene 8 of c_1 is selected, $t=4$

c_1	V	V	U	V	V	V	U	V	V
c_2	V	V	U	V	U	V	V	V	V
o_2	3	1	0	3	4	1	5	1	2

step 6 : gene 7 of c_2 is selected, $t=5$

c_1	V	V	V	V	V	V	V	V	V
c_2	V	V	U	V	U	V	V	V	V
o_2	3	1	6	3	4	1	5	1	2

step 7 : gene 3 of c_1 is selected, $t=6$

(c) Generative process of offspring chromosome o_2

FIGURE 4. A schematic diagram of the crossover operation of the MATMCD-BN algorithm. Among them, U indicates that the corresponding gene has been unvisited, and V indicates that the corresponding gene has been visited. The number in the square stands for the number of the community to which the corresponding vertex belongs. The number 0 indicates that the corresponding gene has not been filled. (a) Parent chromosomes c_1 and c_2. (b) Generative process of offspring chromosome o_1. (c) Generative process of offspring chromosome o_2.

operator, one-way crossover operator and two-way crossover operator). The experimental results on various real data sets show that the Q_b value of the final result of MATMCD-BN algorithm using two-way random crossover operator is significantly higher than that of MATMCD-BN algorithm using uniform crossover operator, one-way crossover operator and two-way crossover operator. Compared with the uniform crossover operator, the increase is about 47%. Compared with the one-way crossover operator, the increase is about 45%. Compared with the two-way crossover operator, the increase is about 42%. Two-way random crossover operator also contributes to reducing the final convergence time of the algorithm. Compared with the uniform crossover operator, the convergence time of the two-way random crossover operator is reduced by about 26%. Compared with the one-way crossover operator, the convergence time of the two-way random crossover operator is reduced by about 23%. Compared with the two-way crossover operator, the convergence time of the two-way random crossover operator is reduced by about 19%.

F. MUTATION

In the MATMCD-BN algorithm, we use two mutation operators, one is the traditional mutation operator, we call it mutation operator 1, the other is the mutation operator proposed by us, we call it mutation operator 2.

As shown in Fig. 3, in string-based representations, each gene represents a vertex and the gene value represents the community to which the vertex belongs. In the community discovery algorithms, the traditional mutation operator (mutation operator 1) first randomly selects a node v_i, then randomly selects a neighbor node v_j of v_i (v_i and v_j are not in a community), and finally replaces the gene value (community label) of v_i with the gene value of v_j [38].

An example of traditional mutation operation is shown in Fig. 5. As shown in Fig. 5, an individual X is chosen first, and then a node 7 on individual X is randomly selected. The neighbors of node 7 include nodes 1, 2 and 4. Assume that node 2 is randomly selected. Therefore, the community label of node 7 is set as the community label 1 of node 2.

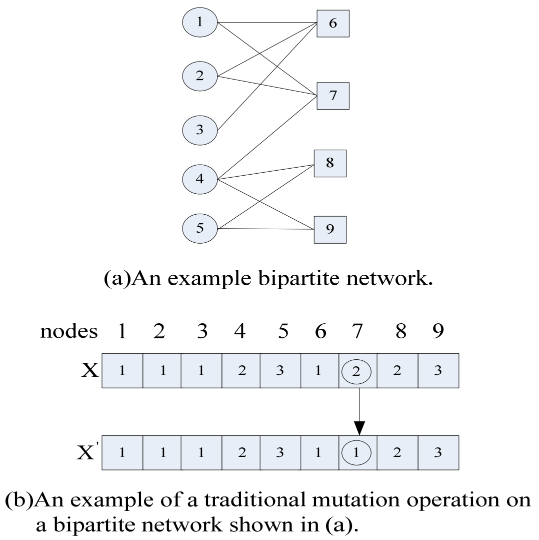
Before introducing mutation operator 2, let's first introduce the definition of community separability [54].

Definition 1 (Community Separability(Sep)): Let C be a community in the graph. The separability of C is the ratio between the number of internal edges and external edges in C. It is defined as follow.

$$\mathrm{Sep}\,(C) = \frac{|\{(u, w) \in E : u \in C, w \in C\}|}{|\{(u, w) \in E : u \in C, w \notin C\}|} \tag{3}$$

The higher the value of Sep (C) is, the more obvious the community structure of community C is.

The mutation operator 2 is introduced below. When Sep (C) is below a threshold θ, we believe that the nodes in C cannot form a community. In this case, mutation operator 2 is needed to assign nodes in C to other communities. The pseudocode of mutation operator 2 of the MATMCD-BN algorithm is shown in algorithm 2. An example of the operation of mutation operator 2 of the MATMCD-BN algorithm is shown in Fig. 6.



(a)An example bipartite network.

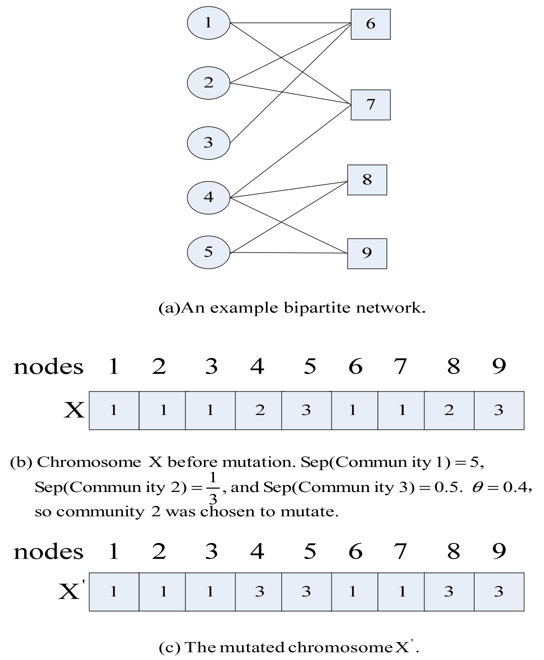
(b)An example of a traditional mutation operation on a bipartite network shown in (a).

FIGURE 5. An example of traditional mutation operation.

Algorithm 2 Pseudocode of Mutation Operator 2 of the MATMCD-BN Algorithm

Parameters: parent chromosome X, a threshold value ρ;
Input: an adjacent matrix A of a bipartite network;
Output: mutated chromosome X';

```
C = The community with the smallest community separability in X was found by equation (3);
if C's community separability < $\rho$ then
    ns = set of nodes in C;
    $X' = X$;
    for each node $i \in$ ns do
        Randomly select a neighbor j of node i (node i and j are not in the same community), assign the community label of node j to i, and modify the corresponding gene value in X'. If node i and all its neighbors belong to the same community, then the community label of node i will not change;
    end for
    $q_1$ = The $Q_D$ value of X calculated by equation (2);
    $q_2$ = The $Q_D$ value of $X'$ calculated by equation (2);
    if $q_2 > q_1$ then
        return $X'$;
    end if
end if
return X;
```



(a)An example bipartite network.

(b) Chromosome X before mutation. Sep(Commun ity 1) = 5, Sep(Commun ity 2) = $\frac{1}{3}$, and Sep(Commun ity 3) = 0.5. $\theta = 0.4$, so community 2 was chosen to mutate.

(c) The mutated chromosome X'.

FIGURE 6. An example of the operation of mutation operator 2 of the MATMCD-BN algorithm.

Fig. 6 (a) shows a bipartite network. Fig. 6 (b) shows a chromosome generated by the network in Fig. 6 (a). It consists of three communities, namely community 1, 2 and 3. Using equation (3), we can calculate Sep (1) = 5, Sep (2) = $\frac{1}{3}$, Sep (3) = 0.5. Let θ = 0.4. Therefore, community 2 is selected to mutate. In the neighborhood of node 4, node 7 and 9 are not in the same community as node 4. Assume that node 9 is randomly selected. Therefore, the community label of node 4 is changed to the community label 3 of node 9. In the neighborhood of node 8, node 4 and node 5 are not in the same community as node 8. Because, at the moment, the community labels of nodes 4 and 5 are all 3, so the community label of node 8 is changed to 3.

We also compare the performance of our MATMCD-BN algorithm using mutation operator 1 (i.e., traditional mutation operator) and mutation operator 2 with that of the MATMCD-BN algorithm using mutation operator 1 only. The experimental results on various real data sets show that the Q_b value of the final result of our MATMCD-BN algorithm has been significantly improved compared with that of the MATMCD-BN algorithm using only mutation operator 1. Compared with the MATMCD-BN algorithm which only uses mutation operator 1, the Q_b value increased by about 29%. Similarly, by combining mutation operator 1 with mutation operator 2, our MATMCD-BN algorithm also reduces the convergence time of the algorithm. Compared with the MATMCD-BN algorithm which only uses mutation operator 1, the convergence time is reduced by about 13%.

G. LOCAL SEARCH FUNCTION

Local search function is an important component of memetic algorithm. It has an important influence on the good and bad of the final solution and the convergence rate of the algorithm. To this end, we propose a local search function named Local_search(), which can enhance the quality of the final solution and accelerate the convergence rate of the algorithm. Given the optimal solution best_of_children of the child population, for each node j, we store in the set commSet the communities to which most neighbors of node j belong (There may be two or more eligible communities.). If we can improve the fitness function by moving node i to a community in commSet, then we do this, otherwise we do not.

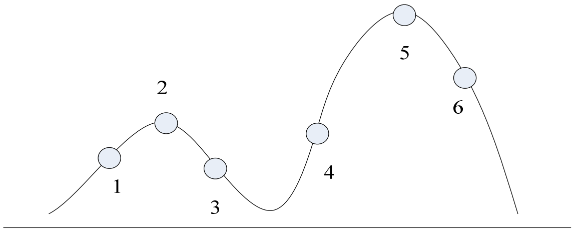


FIGURE 7. A schematic diagram of local search.

Algorithm 3 gives detailed information about the function Local_search().

Algorithm 3 Local Search().

Parameters: optimal chromosome best_of_children in the child population, a threshold value θ;
Input: an adjacent matrix A of a bipartite network;
Output: improved chromosome best_of_children;

```
1:  Produce a random arrangement of the integers from
        1 to n and store the arrangement in an array
        named o;
2:  for i = 1 to n do
3:      j = $o$[i];
4:      commSet = Ø;
5:      Store the communities to which most of the
        neighbors of node j belong in the set commSet.
        (there may be more than one eligible communities);
6:      for k = 1 to |commSet| do
7:          c = commSet[k];
8:          newPartition=Change the community label of node
                j of best_of_children to c to form
                a new chromosome;
9:          $q_1$ =The $Q_D$ value of chromosome
                best_of_children was calculated by equation
                (2);
10:         $q_2$ = The $Q_D$ value of
                chromosome newPartition was calculated by
                equation (2);
11:         if $q_2 > q_1$ then
12:             best_of_children = newPartition;
13:         else if rand(0, 1) < $\theta$
14:             best_of_children = newPartition;
15:         end if
16:     end for
17: end for
18: return best_of_children;
```

In algorithm 3, rand (0, 1) represents a real number between 0 and 1 randomly generated. One of the main advantages of our local search function, Local_search(), is that it can adopt a poorer solution with a certain probability. This advantage enables it not only to improve the results in the iteration process, but also to escape from the local optimum solution. This is achieved by the conditional statement of line 13 in algorithm 3.

As shown in Fig. 7, assuming that the algorithm currently searches to point 1 and the local optimal solution point 2 is found, some other strategies will stop searching. Because some other strategies are incapable of discovering the global optimal solution by moving in a small range around point 2. However, our proposed Local_search() can move around with a certain probability, that is, it can accept a solution with a specified probability that is poorer than the current solution. This enables the function to escape from the local optimal solution and reach the global optimal solution after several algorithm steps. As shown in Fig. 7, when Local_search() discovers point 2 is a local optimum point, it can move to point 3 with a specified probability that is worse than point 2. After a few moving steps, it may reach point 4, which escapes from local optimal solution point 2. Then, the function can seek out the global optimal solution point 5. Thus, the function can discover the global optimal solution with a certain probability. Therefore, our proposed Local_search() function can escape from the local optimum and attain the global optimum.

In algorithm 4, the MATMCD-BN algorithm proposed by us is described in detail.

V. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETTINGS

Here, we will assess MATMCD-BN in detail through experiments, and compare it with three existing famous algorithms in five synthetic bipartite networks and six real bipartite networks. These three algorithms are described as follows.

1) ASYMINTIMACY

In [5], two parameters are defined to show the relations between the same type of vertices and heterogeneous vertices respectively. In this algorithm, two different types of vertices are tackled independently on the basis of different closeness. In the first place, the same type of vertices are clustered into subsets based on the asymmetric intimacy. Subsequently, in order to form core communities, the second type of vertices is divided into the corresponding set. In this way, a set of core communities have been acquired. If the overlap rate of the two core communities exceeds the threshold, then this pair of communities are merged. This process is repeated as long as there are core communities that can be combined in the core communities set. The time complexity of this algorithm is $O(2n^2 + mn)$, where m and n are the number of links and vertices in the graph respectively.

2) LP BRIM

By combining label propagation (LP) with BRIM, the authors of [3] proposed a bipartite network community detection method named LP BRIM, which extended the work of BRIM. The worst case time complexity of this algorithm is $O(n^2)$, where n is the number of nodes in the graph. In real bipartite networks, this worst-case time complexity is acceptable.

Algorithm 4 MATMCD-BN Algorithm

Parameters: population size popsize, crossover rate P_c, first mutation rate P_{m1}, second mutation rate P_{m2}, number of generations without improvement t, a threshold value θ;
Input: an adjacent matrix A of a bipartite network;
Output: a community division C of the network;

```
pop = The initial population obtained by using
        algorithm 1;
repeat
    Using equation (2) to assess the fitness of each
    chromosome in pop;
    parents = The parent chromosomes for genetic
              operations were selected from pop by
              tournament selection method;
    children = Using $P_c$ to perform two-way random
               crossover operator on chromosomes
               in parents;
    children = Using $P_{m1}$ to perform mutation
               operator 1 on chromosomes in children;
    children = Using $P_{m2}$ to perform mutation
               operator 2 on chromosomes in children;
    best_of_children = Chromosome with the highest
                       fitness in children;
    best_of_children = Local_search
                       (best_of_children, $\theta$);
    pop = pop$\cup$children$\cup$
          best_of_children;
    pop = The first M chromosomes with the
          highest fitness in pop were selected as the next
          generation population;
until number of generations without improvement in the
best chromosome in pop >= t
C = Community division of chromosome with the
    highest fitness in pop;
return C.
```

3) ADAPTIVE BRIM

The authors of [2] propose bipartite, recursively induced modules (BRIM) algorithm on the basis of the iterative optimization idea of modularity measure Q_b in bipartite networks. At each iteration of the algorithm, Q_b is non-decreasing. However, this algorithm often finds the local optimal solution rather than the global optimal solution. At the same time, the number of modules does not need to be specified in advance.

The experimental data of algorithms BRIM, LP BRIM and AsymIntimac in this paper are from [59]. MATMCD-BN is implemented by C# 4.0 using Microsoft Visual Studio 2010. In the experiment, we set the parameters popsize = 1000, $P_c = P_{m1} = P_{m2} = 0.3$, $\rho = 0.2$, $t = 3$, and $\theta = 0.15$.

4) MEASUREMENTS

For compare the performance of different algorithms, two types of metrics are generally used. If the community structures are known beforehand, the normalized mutual information (NMI) formula is used to produce a score between 0 and 1. If the opposite is true, then modularity [35] is used to compare algorithms. Modularity was originally proposed for one-mode networks. Barber modifies the formula of modularity as Q_b for bipartite networks [2]. A higher Q_b value from [0,1] indicates a better community structure. A Q_b value close to 0 represents a poor community structure.

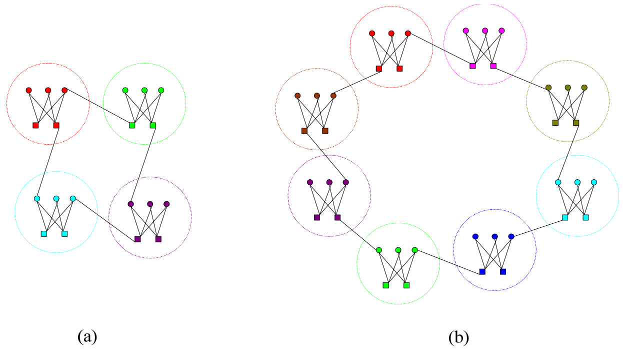


FIGURE 8. A ring consisting of bicliques. (a) A ring consisting of four bicliques. Each biclique is made up of two square nodes and three circular nodes, and the square nodes are fully connected to the circular nodes. (b) A ring is made up of eight bicliques with the same linking rules used in (a).

B. SYNTHETIC NETWORKS

Many existing community detection methods lean upon maximization of modularity. However, these methods may suffer from resolution limitation problem [17]. They usually cannot find communities smaller than $\sqrt{2L}$, where L is the number of links of the network. To prove the effectiveness of MATMCD-BN, a series of synthetic networks consisting of different numbers of biclique were designed. As shown in Fig. 8, one synthetic network consists of 4 sequentially connected bicliques, and the other consists of 8 in turn connected bicliques. Each biclique is made up of two modes of nodes, and the nodes of different modes are fully interconnected. The basic topological characteristics of all synthetic networks are shown in Table 1. The following experiments are performed on synthetic networks with different numbers of bicliques. The experimental results are shown in Table 2 and Fig. 9.

As can be seen from Table 2, for networks with only 4 bicliques and 8 bicliques, MATMCD-BN, LP BRIM and Adaptive BRIM can correctly detect communities. Further experiments show that when the network has 16 bicliques, Adaptive BRIM detects 15 communities and obtains NMI = 0.934. LP BRIM detects 13 communities and gets NMI = 0.802. However, MATMCD-BN can still obtain an accurate solution, which has been significantly improved compared with the other three algorithms. Similar results have been obtained on 64 bicliques network. For Adaptive BRIM, MATMCD-BN achieved about 2% improvement, and about 13% improvement for LP BRIM. Experiments on 128 bicliques network show that MATMCD-BN improves about 1% for Adaptive BRIM and about 12% for LP BRIM. Compared with the other three algorithms, AsymIntimacy is

TABLE 1. The basic topological characteristics of rings of bicliques used in the experiment in this paper.

Network	*p*	*q*	*n*	*m*	*<k>*	*C*	*r*
4 bicliques	12	8	20	28	2.800	0.482	-0.5
8 bicliques	24	16	40	56	2.800	0.482	-0.5
16 bicliques	48	32	80	112	2.800	0.482	-0.5
64 bicliques	192	128	320	448	2.800	0.482	-0.5
128 bicliques	384	256	640	896	2.800	0.482	-0.5

TABLE 2. Performance summary of MATMCD-BN, AsymIntimacy, LP BRIM and Adaptive BRIM on rings of bicliques. NMI is the detection accuracy of different bipartite community discovery algorithms. N_C is the number of communities detected by different algorithms.

Network	*MATMCD-BN*		*LP BRIM*		*Adaptive BRIM*		*AsymIntimacy*	
	(NMI)	*(N_C)*	*(NMI)*	*(N_C)*	*(NMI)*	*(N_C)*	*(NMI)*	*(N_C)*
4 bicliques	**1.000**	**4**	**1.000**	4	**1.000**	4	0.714	4
8 bicliques	**1.000**	**8**	**1.000**	8	**1.000**	8	0.759	8
16 bicliques	**1.000**	**16**	0.802	13	0.934	15	0.785	16
64 bicliques	**1.000**	**64**	0.887	56	0.986	63	0.816	64
128 bicliques	**1.000**	**128**	0.900	113	0.993	127	0.826	128

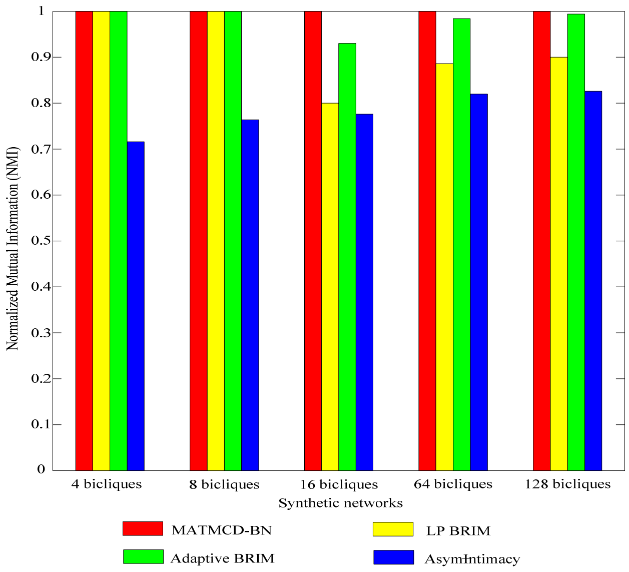


FIGURE 9. Performance summary of MATMCD-BN, AsymIntimacy, LP BRIM and Adaptive BRIM on rings of bicliques. NMI is the detection accuracy of different bipartite community discovery algorithms.

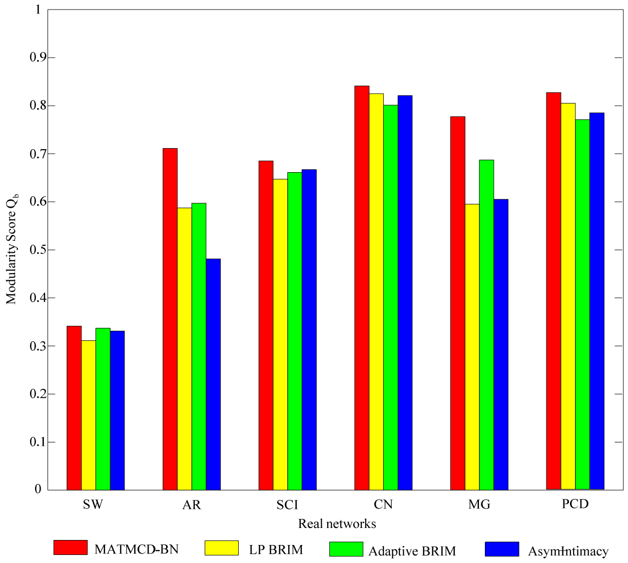


FIGURE 10. Performance comparison of MATMCD-BN, AsymIntimacy, LP BRIM and Adaptive BRIM on real bipartite networks. Q_b indicates the modularity score in each two-mode networks.

always the worst according to the NMI values on all synthetic networks used in this experiment. But compared with LP BRIM and Adaptive BRIM, AsymIntimac always gets the right number of communities. Therefore, due to the resolution limit problem, the other three algorithms can not accurately detect small communities, but MATMCD-BN can precisely detect such small communities.

C. REAL NETWORKS

Several experiments in this section are performed on real networks, that is to say, the modular structure is unknown, and Q_b is used to verify the accuracy of the algorithm. The real bipartite networks used in our experiments include Southern Women Events Participation (SW), America Revolution (AR), Scotland Corporate Interlock (SCI), Crime Network (CN), Malaria and var Genes (MG) and Protein Complex and Drug network (PCD). The basic topological characteristics of all real networks are shown in Table 3. The following experiments are performed on real networks. The experimental results are shown in Table 4 and Fig. 10.

1) SOUTHERN WOMEN EVENTS PARTICIPATION (SW)

The Southern Women dataset was produced by Davis *et al.* [6] in the 1930s in the southern United States. It indicates the interaction among 18 women who took part in 14 informal social activities. The original goal of the survey is to explore the corresponding relationship between social strata and informal touches. Because this dataset forms a natural bipartite network with small amount of data, researchers have done a lot of research on it. This network is a connected

TABLE 3. The basic topological characteristics of the real-world bipartite network used in the experiment in this paper.

Network	*p*	*q*	*n*	*m*	*<k>*	*C*	*r*
SW	18	14	32	89	5.563	0.328	-0.337
AR	136	5	141	160	2.270	0.781	-0.743
SCI	108	136	244	358	3.140	0.303	-0.171
CN	829	551	1380	1476	2.139	0.427	-0.166
MG	297	806	1103	2965	5.376	0.227	-0.300
PCD	680	739	1419	3690	1.746	0.407	-0.140

TABLE 4. Performance comparison of MATMCD-BN, AsymIntimacy, LP BRIM and Adaptive BRIM on real bipartite networks. Q_b indicates the modularity score in each two-mode networks. $N_c$is the number of bipartite clusters (i.e. bipartite communities) discovered by different methods.

Network	*MATMCD-BN*		*AsymIntimacy*		*LP BRIM*		*Adaptive BRIM*	
	(Q_b)	(N_c)	(Q_b)	(N_c)	(Q_b)	(N_c)	(Q_b)	(N_c)
SW	**0.356**	4	0.333	4	0.313	2	0.345	4
AR	**0.708**	5	0.480	3	0.591	5	0.602	5
SCI	**0.689**	52	0.668	40	0.648	36	0.660	24
CN	**0.841**	165	0.821	142	0.823	203	0.798	104
MG	**0.786**	146	0.604	48	0.592	60	0.687	28
PCD	**0.826**	158	0.784	79	0.806	107	0.770	113

and unweighted network. The network contains 32 vertices and 89 links. The properties of this network are shown in Table 3. Four different community discovery algorithms are compared to detect the cluster structure of this network. The comparison results are shown in Table 4.

In this experiment, MATMCD-BN partitions SW Network into 4 different sizes of bipartite communities. Among them, the largest community consists of 8 women and 6 events. The second largest community is made up of 4 women and 4 events. The third largest community is comprised of 4 women and 3 activities. The last community consists of 2 women and 1 event.

2) AMERICAN REVOLUTION (AR)

This dataset is made up of membership information for 136 members of 5 organizations. The founding time of these organizations can be traced back to before the American Revolution [7]. The dataset includes a large number of American celebrities. The relationship between members and organizations may be represented as a bipartite network. A link between an individual and an organization indicate that the individual is a member of this organization. In Table 3, the basic topological characteristics of this network are described in detail. Table 3 indicates that the network is comprised of 141 nodes and 160 edges. Next, we will compare four different bipartite network community discovery methods to find the community partitions in this network. The comparison results are shown in Table 4.

As can be seen from Table 4, the optimal modularity score ($Q_b = 0.708$) is obtained from MATMCD-BN algorithm, which is superior to the other three comparison algorithms.

MATMCD-BN found five different communities in the American Revolution network. These five communities have the same organizational model, and each community consists of a specific organization and its members. From the results of the detection, we see that all organizations are located in the core of their respective communities, encircled by members of the organization. We also see that a small quantity of individuals are members of different organizations, they are overlapping nodes between different clusters.

3) SCOTLAND CORPORATE INTERLOCK (SCI)

This dataset is the third dataset used in our experiment. This dataset reveals the Scottish business chain network between 1904 and 1905 [8]. The network consists of 136 directors who hold multiple directorship positions in 108 share-holding corporations. If a person belongs to the board of directors of a company, there is an unweighted edge between him and the corporation. But this bipartite network is not connected, it consists of several connected components.

As can be seen from Table 4, among the four comparison algorithms, MATMCD-BN obtains the best community partition with ($Q_b = 0.689$).

4) CRIME NETWORK (CN)

This dataset includes individuals who were documented in at lowest one criminal incident. The individual is either a victim, a witness or as a suspect in the incident [7].

The relations between criminal individuals and criminal incidents naturally constitute a bipartite network with 1476 edges. These edges connect 829 criminal individuals and 551 criminal incidents. As can be seen from Table 4, MATMCD-BN is superior to the other three methods in precision ($Q_b = 0.841$).

5) MALARIA AND VAR GENES (MG)

By a protein camouflage encoded in var genes, parasites can escape the human immune mechanism [56]. In order to evade the human immune system, the var gene is often recombined to create new camouflages, which naturally produces community structure [33]. Therefore, the var genes and their genetic subsequences constitute a two-mode network consisting of two types of nodes. This network has a natural community partition. As shown in Table 3, the vertices of two types of MG network contain 297 genes and

806 subsequences, respectively. There are 2965 edges connecting different kinds of nodes in this two-mode network.

In this real network, four different detection algorithms detect community structure of the genes and their genetic substrings respectively. As can be seen from Table 4, compared with AsymIntimacy, LP BRIM and Adaptive BRIM, MATMCD-BN achieves the best community partition with modularity value $Q_b = 0.786$.

6) THE PROTEIN COMPLEX AND DRUG NETWORK (PCD)

Recently, the studies in the field of biology have found contacts between some protein complexes and corresponding illness. The bipartite networks analyzed by Nacher and Schwartz contain two types of vertices: drug and protein complexes. This network consists of 680 drugs and 739 protein complexes [58]. It reveals the interrelationship between molecules and human diseases. The basic topological characteristics of PCD networks are described in Table 3. As shown in Table 3, PCD is made up of 1419 nodes and 3690 edges. As shown in Table 4, MATMCD-BN is superior to the other three algorithms, and its modularity value is $Q_b = 0.826$. LP BRIM takes second place, and its modularity value is $Q_b = 0.806$.

VI. CONCLUSION

In the study of complex networks including bipartite networks, community structure is a crucial network property. For better research and make use of such a network, it is very important to detect its community structure. This paper presents a memetic algorithm called MATMCD-BN for community discovery in two-mode networks. The chromosome representation of MATMCD-BN algorithm uses the classical string-based representation method. We proposed a novel population initialization method for bipartite network community detection to speed up the convergence rate of this algorithm. The fitness function of MATMCD-BN algorithm uses the Q_D function proposed in [55], which solves the resolution limit problem of traditional modularity. For selecting parent chromosomes for crossover operator and mutation operators, we use the tournament selection method, which gives individuals in the population equal opportunities to be selected for subsequent genetic operations. Besides using the traditional mutation operator, we also propose a new crossover operator and a new mutation operator. The new two-way random crossover operator may better inherit the genetic characteristics of parent chromosomes, and the new mutation operator can significantly improve the multiformity of the population. In addition, we propose a new local search function, which can improve the quality of the final solution and the convergence rate of the algorithm, and make the algorithm can jump away from the local optimum with a certain probability and reach the global optimum. In order to check the performance of the presented method, a lot of experiments have been carried out on five synthetic networks and six real networks. The experimental results were compared with three famous bipartite network community discovery algorithms. The comparison results indicate that the MATMCD-BN algorithm is superior to the other three algorithms, which shows that our algorithm is a good algorithm to find community structures in bipartite networks.

APPENDIX

Algorithm 5 Pseudocode of Random Initialization Method

Parameters: population size popsize;
Input: an adjacent matrix A of a bipartite network;
Output: initial population;

```
for i = 1; i <= popsize; i++ do
    Generate an random integer permutation of
    integers 1-n and store it in the array o (n is the
    amount of vertices in a bipartite graph);
    for j = 1; j <= n; j++ do
        pop[i][j] = o[j];
    end for
end for
return pop[];
```

Algorithm 6 Pseudocode of Initialization Method Using the Label Propagation Algorithm

Parameters: population size popsize;
Input: an adjacent matrix A of a bipartite network;
Output: initial population;

```
for i = 1; i <= popsize; i++ do
    Generate an random integer permutation of integers
    1-n and store it in the array o (n is the amount of
    vertices in a bipartite graph);
    for j = 1; j <= n; j++ do
        pop[i][j] = o[j];
    end for
for j = 1; j <= n; j++ do
        pop[i][j] = The label shared by most of the
        neighbor nodes of node j. If there are more than one
        labels satisfy the above condition, randomly select
        one of them.
        end for
    end for
return pop[];
```

REFERENCES

[1] J. Xie, B. K. Szymanski, and X. Liu, "SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process," in *Proc. IEEE 11th Int. Conf. Data Mining Workshops (ICDMW)*, Dec. 2011, pp. 344–349.

[2] M. J. Barber, "Modularity and community detection in bipartite networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 76, no. 6, 2007, Art. no. 066102.

[3] X. Liu and T. Murata, "Community detection in large-scale bipartite networks," *Trans. J. Jpn. Soc. Artif. Intell.*, vol. 25, no. 1, pp. 16–24, 2010.

[4] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying communities in networks," *Proc. Nat. Acad. Sci. USA*, vol. 101, no. 9, pp. 2658–2663, 2004.

[5] X. Wang and X. Qin, "Asymmetric intimacy and algorithm for detecting communities in bipartite networks," *Phys. A, Stat. Mech. Appl.*, vol. 462, pp. 569–578, Nov. 2016.

[6] A. Davis, B. B. Gardner, and M. R. Gardner, *Deep South: A Social Anthropological Study of Caste and Class*. Columbia, SC, USA: Univ. South Carolina Press, 1941.

[7] J. Kunegis, "Konect: The koblenz network collection," in *Proc. 22nd Int. Conf. World Wide Web*, Rio de Janeiro, Brazil, May 2013, pp. 1343–1350.

[8] J. Scott and M. Hughes, *The Anatomy of Scottish Capital: Scottish Companies and Scottish Capital*. Montreal, QC, Canada: McGill-Queen's Press-MQUP, 1981, p. 291.

[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] Z. Li, R.-S. Wang, S. Zhang, and X.-S. Zhang, "Quantitative function and algorithm for community detection in bipartite networks," *Inf. Sci.*, vols. 367–368, pp. 874–889, Nov. 2016.

[11] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell Syst. Tech. J.*, vol. 49, pp. 291–307, Feb. 1970.

[12] Y. Cui and X. Wang, "Uncovering overlapping community structures by the key bi-community and intimate degree in bipartite networks," *Phys. A, Stat. Mech. Appl.*, vol. 407, pp. 7–14, Aug. 2014.

[13] A. Pothen, H. D. Simon, and K.-P. Liou, "Partitioning sparse matrices with eigenvectors of graphs," *SIAM J. Matrix Anal. Appl.*, vol. 11, no. 3, pp. 430–452, 1990.

[14] S. Kelley, M. Goldberg, M. Magdon-Ismail, K. Mertsalov, and A. Wallace, "Defining and discovering communities in social networks," in *Handbook of Optimization in Complex Networks*. Boston, MA, USA: Springer, 2012, pp. 139–168.

[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, pp. 814–818, Jun. 2005.

[16] G. Bello-Orgaz, S. Salcedo-Sanz, and D. Camacho, "A multi-objective genetic algorithm for overlapping community detection based on edge encoding," *Inf. Sci.* vol. 462, pp. 290–314, Sep. 2018.

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

[18] A. McDaid and N. Hurley, "Detecting highly overlapping communities with model-based overlapping seed expansion," in *Proc. Int. Conf. IEEE Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2010, pp. 112–119.

[19] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, Feb. 2004, Art. no. 026113.

[20] F. Reid, A. McDaid, and N. Hurley, "Partitioning breaks communities," in *Mining Social Networks and Security Informatics*. Dordrecht, The Netherlands: Springer, 2013, pp. 79–105.

[21] T. Zhou, J. Ren, M. Medo, and Y.-C. Zhang, "Bipartite network projection and personal recommendation," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 76, Oct. 2007, Art. no. 046115.

[22] Y. Cui and X. Wang, "Detecting one-mode communities in bipartite networks by bipartite clustering triangular," *Phys. A, Stat. Mech. Appl.*, vol. 457, pp. 307–315, Sep. 2016.

[23] R. Guimerà, M. Sales-Pardo, and L. A. Amaral, "Module identification in bipartite and directed networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 76, Sep. 2007, Art. no. 036102.

[24] B.-L. Chen, L. Chen, S. R. Zou, and X.-L. Xu, "Detecting community structure in bipartite networks based on matrix factorisation," *Int. J. Wireless Mobile Comput.*, vol. 6, pp. 599–607, Nov. 2013.

[25] P. Zhang, J. Wang, X. Li, M. Li, Z. Di, and Y. Fan, "Clustering coefficient and community structure of bipartite networks," *Phys. A, Stat. Mech. Appl.*, vol. 387, pp. 6869–6875, Dec. 2008.

[26] W.-D. Pei, W. Xia, X.-R. Ma, and L.-F. Jiang, "Robustness and statistical characters of a class of complex network models," in *Recent Advances in Computer Science and Information Engineering* (Lecture Notes in Electrical Engineering), vol. 129. Berlin, Germany: Springer, 2012, pp. 747–752.

[27] C. Zhou, L. Feng, and Q. Zhao, "A novel community detection method in bipartite networks," *Phys. A, Stat. Mech. Appl.*, vol. 492, pp. 1679–1693, Feb. 2018.

[28] A. F. Liu, C. H. Fu, Z. P. Zhang, H. Chang, and D. R. He, "An empirical statistical investigation on Chinese mainland movie network," *Complex Syst. Complex. Sci.*, vol. 4, no. 3, pp. 10–16, 2007.

[29] Z. Li, S. Zhang, R.-S. Wang, X.-S. Zhang, and L. Chen, "Quantitative function for community detection," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 77, no. 3, 2008, Art. no. 036109.

[30] Z. Jiang, J. Liu, and S. Wang, "Traveling salesman problems with PageRank distance on complex networks reveal community structure," *Phys. A, Stat. Mech. Appl.*, vol. 463, no. 2016, pp. 293–302, 2016.

[31] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 6, 2004, Art. no. 066133.

[32] Z. Li and J. Liu, "A multi-agent genetic algorithm for community detection in complex networks," *Phys. A, Stat. Mech. Appl.*, vol. 449, pp. 336–347, May 2016.

[33] D. B. Larremore, A. Clauset, and A. Z. Jacobs, "Efficiently inferring community structure in bipartite networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 90, Jul. 2014, Art. no. 012805.

[34] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 70, Dec. 2004, Art. no. 066111.

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

[36] A. Arenas, A. Fernández, and S. Gómez, "Analysis of the structure of complex networks at different resolution levels," *New J. Phys.*, vol. 10, May 2008, Art. no. 053039.

[37] C. Shi, Z. Yan, Y. Cai, and B. Wu, "Multi-objective community detection in complex networks," *Appl. Soft Comput.*, vol. 12, pp. 850–859, Feb. 2012.

[38] C. Pizzuti, "A multiobjective genetic algorithm to find communities in complex networks," *IEEE Trans. Evol. Comput.*, vol. 16, no. 2, pp. 418–430, Jun. 2012.

[39] C. Pizzuti, "GA-Net: A genetic algorithm for community detection in social networks," in *Parallel Problem Solving From Nature—PPSN X*, vol. 5199, G. Rudolph, T. Jansen, N. Beume, S. Lucas, and C. Poloni, Eds. Berlin, Germany: Springer, 2008, pp. 1081–1090.

[40] 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.

[41] D. Chen, F. Zou, R. Lu, L. Yu, Z. Li, and J. Wang, "Multi-objective optimization of community detection using discrete teaching–learning-based optimization with decomposition," *Inf. Sci.*, vol. 369, pp. 402–418, Nov. 2016.

[42] F. Zou, D. Chen, S. Li, R. Lu, and M. Lin, "Community detection in complex networks: Multi-objective discrete backtracking search optimization algorithm with decomposition," *Appl. Soft Comput.*, vol. 53, pp. 285–295, Apr. 2017.

[43] Y. Tian, H. Wang, X. Zhang, and Y. Jin, "Effectiveness and efficiency of non-dominated sorting for evolutionary multi- and many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 4, pp. 247–263, Dec. 2017.

[44] R. Shang, J. Bai, L. Jiao, and C. Jin, "Community detection based on modularity and an improved genetic algorithm," *Phys. A, Stat. Mech. Appl.*, vol. 392, pp. 1215–1231, Mar. 2013.

[45] Y. Zhao, W. Jiang, S. Li, Y. Ma, G. Su, and X. Lin, "A cellular learning automata based algorithm for detecting community structure in complex networks," *Neuro-Comput.*, vol. 151, pp. 1216–1226, Mar. 2015.

[46] M. Gong, B. Fu, L. Jiao, and H. Du, "Memetic algorithm for community detection in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 84, Nov. 2011, 056101.

[47] K. R. Žalik and B. Žalik, "Memetic algorithm using node entropy and partition entropy for community detection in networks," *Inf. Sci.*, vols. 445–446, pp. 38–49, Jun. 2018.

[48] C.-H. Mu, J. Xie, Y. Liu, F. Chen, Y. Liu, and L.-C. Jiao, "Memetic algorithm with simulated annealing strategy and tightness greedy optimization for community detection in networks," *Appl. Soft Comput.*, vol. 34, pp. 485–501, Sep. 2015.

[49] A. Said, R. A. Abbasi, O. Maqbool, A. Daud, and N. R. Aljohani, "CC-GA: A clustering coefficient based genetic algorithm for detecting communities in social networks," *Appl. Soft Comput.*, vol. 63, pp. 59–70, Feb. 2018.

[50] L. Ma, M. Gong, J. Liu, Q. Cai, and L. Jiao, "Multi-level learning based memetic algorithm for community detection," *Appl. Soft Comput.*, vol. 19, pp. 121–133, Jun. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494614000623. doi: 10.1016/j.asoc.2014.02.003.

[51] M. R. Mirsaleh and M. R. Meybodi, "A Michigan memetic algorithm for solving the community detection problem in complex network," *Neurocomputing*, vol. 214, pp. 535–545, Nov. 2016.

[52] X. Wang and J. Liu, "A comparative study of the measures for evaluating community structure in bipartite networks," *Inf. Sci.*, vols. 448–449, pp. 249–262, Jun. 2018.

[53] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA, USA: Addison-Wesley, 1989.

[54] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," *Knowl. Inf. Syst.*, vol. 42, no. 1, pp. 181–213, 2015.

[55] Y. Xu, L. Chen, B. Li, and W. Liu, "Density-based modularity for evaluating community structure in bipartite networks," *Inf. Sci.*, vol. 317, pp. 278–294, Oct. 2015.

[56] D. B. Larremore, A. Clauset, and C. O. Buckee, "A network approach to analyzing highly recombinant malaria parasite genes," *PLoS Comput. Biol.*, vol. 9, no. 10, 2013, Art. no. e1003268.

[57] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," 2009, *arXiv:0903.3178*. [Online]. Available: https://arxiv.org/abs/0903.3178

[58] J. C. Nacher and J.-M. Schwartz, "Modularity in protein complex and drug interactions reveals new polypharmacological properties," *PLoS ONE*, vol. 7, no. 1, 2012, Art. no. e30028.

[59] H.-L. Sun, E. Ch'ng, X. Yong, J. M. Garibaldi, S. See, and D.-B. Chen, "A fast community detection method in bipartite networks by distance dynamics," *Phys. A, Stat. Mech. Appl.*, vol. 496, pp. 108–120, Apr. 2018.

[60] M. Li and J. Liu, "A link clustering based memetic algorithm for overlapping community detection," *Phys. A, Stat. Mech. Appl.*, vol. 503, pp. 410–423, Aug. 2018.

[61] C. Zhang, X. Hei, D. Yang, and L. Wang, "A memetic particle swarm optimization algorithm for community detection in complex networks," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 30, no. 2, 2016, Art. no. 1659003.

SHIWEI CHE received the M.E. degree from the Department of Computer Science and Technology, Xinjiang University, Xinjiang, China, in 2010. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Harbin Engineering University. His current research interests include social networks and community detection.

WU YANG received the Ph.D. degree in computer system architecture specialty from the Computer Science and Technology School, Harbin Institute of Technology. He is currently a Professor and a Doctoral Supervisor with Harbin Engineering University. His current research interests include wireless sensor networks, peer-to-peer networks, and information security. He is a member of ACM and a Senior Member of CCF.

WEI WANG received the Ph.D. degree in computer system architecture specialty from Computer Science and Technology School, Harbin Institute of Technology. He is currently a Professor with Harbin Engineering University. His current research interests include social networks and community detection.