Physica A 503 (2018) 410–423

Contents lists available at ScienceDirect

Physica A

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

A link clustering based memetic algorithm for overlapping community detection

Mingming Li, Jing Liu *

Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an, 710071, China

HIGHLIGHTS

- Propose a link clustering based memetic algorithm to detect overlapping communities.
- Meme-Link can successfully detect overlapping communities and nodes on both general and sparse networks.
- Meme-Link outperforms or performs similarly to the state-of-the-art algorithms in the experiments.

ARTICLE INFO

Article history:
Received 12 October 2017
Received in revised form 11 December 2017
Available online 5 March 2018

Keywords:
Community detection
Overlapping community
Memetic algorithm
Link community

ABSTRACT

Community detection has attracted plenty of attention in the field of complex networks recently, since communities often play important roles in networked systems. Overlapping communities are one of the characteristics of social networks, describing the phenomenon that a node may belong to more than one social group. Thus, it is necessary to find overlapping community structures for realistic social network analyses. In this paper, we propose a link clustering based memetic algorithm for detecting overlapping communities. Since links usually represent the unique relationships among nodes, link clustering can find link groups with the same characteristics. As a result, nodes are naturally partitioned into multiple communities. The proposed algorithm optimizes a modularity density function which is able to identify densely connected groups of links on the weighted line graph modeling the network, and then maps link communities to node communities based on a novel genotype representation. In our method, the number of communities can be automatically determined. Experimental results on general and sparse networks show that our method can successfully detect overlapping community structures and almost all the overlapping nodes.

1. Introduction

The community structure is considered to be an important attribute of real-world social networks because it usually represents for the functionality of a system. Community detection divides a network into groups of nodes, where nodes are densely connected inside but sparsely connected outside. However, it is well known that people in social networks naturally have the characteristics of multiple community members. For example, a person is usually associated with social groups such as family, friends, and college; researchers may be active in several areas. Kelley et al. [1] and Reid et al. [2] showed that the overlapping is indeed a significant feature of many real-world social networks. For this reason, there is growing interest

* Corresponding author.
E-mail address: neouma@163.com (J. Liu).
URL: http://see.xidian.edu.cn/faculty/liujing/ (J. Liu).

https://doi.org/10.1016/j.physa.2018.02.133
0378-4371/

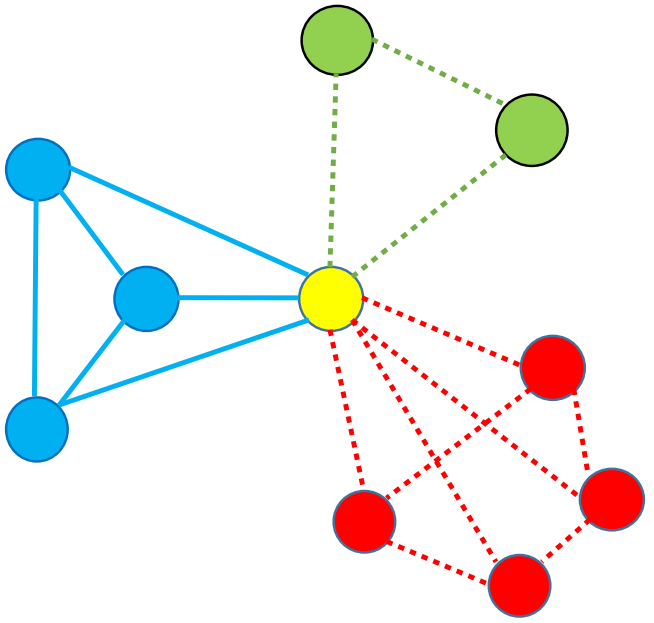

Fig. 1. An example of overlapping communities. In this example, a meaningful partition consists of dividing the links into three groups (blue lines, green lines and red lines). In that case, the central node (yellow one) belongs to the three communities because it is at the interface between these link communities. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

in studying overlapping community detection algorithms that can identify a set of clusters which are not necessarily disjoint. There could be nodes that belong to more than one cluster.

Until now, lots of overlapping community detection methods have been proposed, but most of them are node-based algorithms. The node-based overlapping community detection algorithms [3–7] divide the nodes directly into different communities. However, link-based communities can more naturally represent overlapping structures because an edge can only belong to one community, which greatly simplifies the difficulty of coding. After the edge clustering is complete, the nodes connected to the edge naturally belong to different communities. As shown in Fig. 1. It is clear that the blue edges, the green edges and the red edges are in different communities. The central node (yellow one) naturally belongs to three communities.

In this paper, we propose a new link-based community detection algorithm, named as Meme-Link, which uses the memetic algorithm to discover overlapping communities in the network. Evolutionary algorithms that interspersed the recombination of high quality solutions with periods of intensive individual optimization were named as memetic algorithms in [8,9]. Meme-Link uses the concept of modularity density [10] to measure the quality of communities of a network, and optimizes this quantity by running the memetic algorithm on the weighted line graph $\boldsymbol{W}(\boldsymbol{G})$ of graph $\boldsymbol{G}$ modeling the network, and then maps link communities to node communities based on a novel genotype representation. The number of communities found by Meme-Link can be automatically determined, without any prior information. Experimental results show that Meme-Link outperforms or performs similarly to the state-of-the-art algorithms on both general and sparse networks.

The rest of this paper is organized as follows. Section 2 briefly reviews the related work on community detection algorithms and the weighted line graph. The framework of Meme-Link is introduced in detail in Section 3. The experiments on both general and sparse networks are given in Section 4. Finally, conclusions are given in Section 5.

2. Related work

Many algorithms with different backgrounds, such as physics, statistics, and data mining, have been proposed to detect communities in complex networks [10–17]. A popular-used one was proposed by Newman and Girvan [11,15], which divides a network into separated clusters of nodes, where each node can belong to only one group. However, most real-world networks are made up of overlapped communities of nodes. Thus, there is a surge of developing methods that allow overlapping among the discovered communities in recent years [3–6,18–20].

2.1. Existing community detection algorithms

Clique Percolation Method (CPM): CMP is based on the assumption that a community consists of overlapping sets of fully connected subgraphs, and the community is detected by searching adjacent cliques. It first identifies all cliques of size

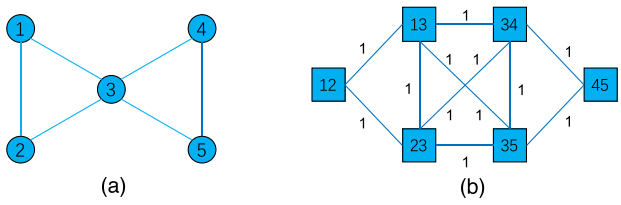

Fig. 2. Bow tie and its line graph. We project the bow tie graph to its line graph, if two edges have at least one common node in the original graph, then the two edges are connected in line graph. (a) Original graph. (b) Line graph.

k in a network. Once these have been determined, a new graph is constructed so that each vertex represents one of these k-cliques. Two nodes are connected if the k-cliques that represent them share $k-1$ members. The connected components in the new graph determine which cliques constitute the community. Overlapping between communities is possible since a vertex can locate in multiple k-cliques simultaneously. CFinder is the implementation of CPM, whose time complexity is polynomial in many applications [4].

Dynamical algorithms: Label propagation algorithms use labels to uncover communities. In COPRA [5], each node updates its belonging coefficients by averaging the coefficients from all its neighbors in a synchronous fashion. The time complexity is $O(vm\log(vm/n))$ per iteration, where parameter v controls the maximum number of communities that a node can associate with, m and n are the number of edges and number of nodes respectively. SLPA [3] is an extension of the label propagation algorithm, and each node can be a listener or a speaker. The role is switched according to whether the node is an information provider or an information consumer. Typically, a node can hold as many labels as possible, depending on what it is experiencing in the underlying process of the underlying network structure. A node accumulates the knowledge of repeating the observation label, rather than removing all the labels except one of them. In addition, the more nodes observe the label, the more likely it will spread this label to others.

Genetic algorithm has been adopted to detect overlapping community, such as GA-Net+ [18] proposed by Pizzuti. The algorithm first transmits the network to the line graph and encodes it according to the nodes on the line graph. GA-Net+ transforms the communities in the line graph into communities of the original network and uses community scores to evaluate community partitions in each iteration. Although GA-Net+ proposed the concept of line graph, it is still a node-based overlapping community detection algorithm.

Link partitioning: Partitioning links instead of nodes to discover communities has been explored, where the node partition of a link graph leads to an edge partition of the original graph. In Ahn et al. [21], single-linkage hierarchical clustering was used to construct a link dendrogram. In order to evaluate the quality of link community, Ahn et al. proposed a new criterion, partition density $\boldsymbol{D}$. This criterion emphasizes the community density, ignoring the connection among communities, which may lead to theoretical bias in small communities. Evans and Lambiotte [22] projected the network into a weighted line graph, whose nodes are the links of the original graph, and then applied disjoint community detection algorithms to finish detection task.

2.2. Line graph

Meme-Link is a link-based community detection algorithm, so we first get the line graph of the network, and then partition links instead of nodes to discover communities, finally map link communities to node communities based on a novel genotype representation method.

A network $\mathcal{N}$ can be modeled as a graph $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$, where $\boldsymbol{V}$ is a set of objects, called nodes, and $\boldsymbol{E}$ is a set of links, called edges, that connects two elements of $\boldsymbol{V}$. A_{ij} is an entry at position (i,j), which is 1 if there is an edge from node i to node j, and 0 otherwise. $\mathbf{A}$ ($N\times N$, N is the number of nodes) is the adjacency matrix of $\boldsymbol{G}$. The simplest way to get line graph $\boldsymbol{L}(\boldsymbol{G})$ from the graph $\boldsymbol{G}$ is that the elements $C_{\alpha\beta}$ of this $L\times L$ matrix (L is the number of edges) are equal to 1 if two edges have at least one common node and 0 otherwise, where $\boldsymbol{C}$ is the adjacency matrix of $\boldsymbol{L}(\boldsymbol{G})$.

Fig. 2 is an example of line graph modeling the bow tie graph. As can be seen from Fig. 2(b), $C_{12,23}$ is equal to 1, because edge e_{12} and edge e_{23} have the common node 2.

It is easy to know that this adjacency matrix is symmetric, and it is a simple graph with $\boldsymbol{L}$ nodes. By construction, each node i of degree k_i of the original graph $\boldsymbol{G}$ corresponds to a k_i fully connected clique in $\boldsymbol{L}(\boldsymbol{G})$. Thus it has $\sum_i \frac{k_i(k_i-1)}{2}=O(\langle k^2\rangle N)$ links. Line graphs have been studied extensively and many properties have been found, Whitney's uniqueness theorem states that the structure of $\boldsymbol{G}$ can be recovered completely from its line graph $\boldsymbol{L}(\boldsymbol{G})$ for any graph other than a triangle or a star network of four nodes [23]. This result implies that projecting the adjacency matrix $\boldsymbol{A}$ to $\boldsymbol{L}(\boldsymbol{G})$ does not lead to any loss of information from the network structure.

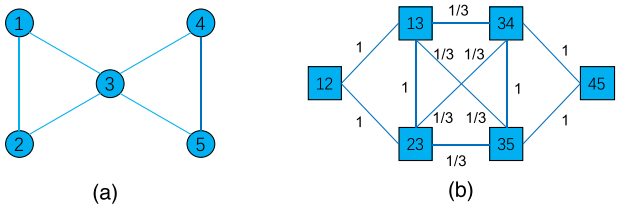

Fig. 3. Bow tie and its weighted line graph. We project the bow tie graph to its weighted line graph according (1). (a) Original graph. (b) Weighted line graph.

2.3. Weighted line graph

The previously obtained line graph is unweighted, which cannot represent the similarity of the two edges. As shown in Fig. 3(a), obviously, edge e_{13} and edge e_{23} have higher similarity than edge e_{13} and edge e_{35} as they have more common nodes. But $C_{13,23}$ and $C_{13,35}$ are both equal to 1 in Fig. 2(b). So, weighted line graph can more accurately represent edge similarity. Here we introduce a commonly used method that projects the adjacency matrix $\boldsymbol{A}$ into weighted line graph $\boldsymbol{W}(\boldsymbol{G})$.

Given a pair of edges e_{ik} and e_{jk} incident on a node k, a similarity can be computed via the Jaccard index defined as

$$S\left(e_{i,k}, e_{j,k}\right) = \frac{|N_i \cap N_j|}{|N_i \cup N_j|} \tag{1}$$

where N_i is the neighborhood of node i including i. See Fig. 3(b), $S(13,23)$ is equal to 1 and $S(13,35)$ equal to 1/3.

3. Meme-Link

In this section, we describe the proposed community detection algorithm Meme-Link, the representation adopted for partitioning the network, and the variation operators used. The Meme-Link employs the framework of memetic algorithm to detect link communities and then converts link communities to node communities. The details of Meme-Link are given in Algorithm 1.

Algorithm 1: Meme-Link

Input:

$\boldsymbol{G}$: Initial network;

Gen: Maximum number of generations;

Pop: Population size;

P_c: Crossover probability;

P_m: Mutation probability;

Output:

Community partition of $\boldsymbol{G}$;

$\boldsymbol{W} \leftarrow$NodeToWeightedEdge($\boldsymbol{G}$);
$\boldsymbol{P} \leftarrow$InitialPopulation(*Pop*, $\boldsymbol{W}$);
for i =1 ***to*** *Gen* ***do***
  $\boldsymbol{P}_{parent} \leftarrow$Selection($\boldsymbol{P}$);
  $\boldsymbol{P}_{child} \leftarrow$GeneticOperation($\boldsymbol{P}_{parent}, P_c, P_m$);
  $\boldsymbol{P}_{new} \leftarrow$LocalSearch($\boldsymbol{P}_{child}$);
  $\boldsymbol{P} \leftarrow$ Update($\boldsymbol{P}$, $\boldsymbol{P}_{new}$);
end for;
Community $\leftarrow$ Decode($\boldsymbol{P}$).

In Algorithm 1, the NodeToWeightedEdge() procedure is responsible for projecting graph $\boldsymbol{G}$ to weighted line graph $\boldsymbol{W}$. The InitialPopulation() procedure is responsible for creating the initial population. The Selection() procedure is used to select parental individuals for mating in GA, here we use the deterministic tournament selection. The GeneticOperation() procedure is used to perform crossover and mutation operation. The Update() procedure is used to generate the current population based on populations $\boldsymbol{P}$ and $\boldsymbol{P}_{new}$. Here, the current population is generated by taking the best *Pop* individuals from $\boldsymbol{P} \cup \boldsymbol{P}_{new}$.

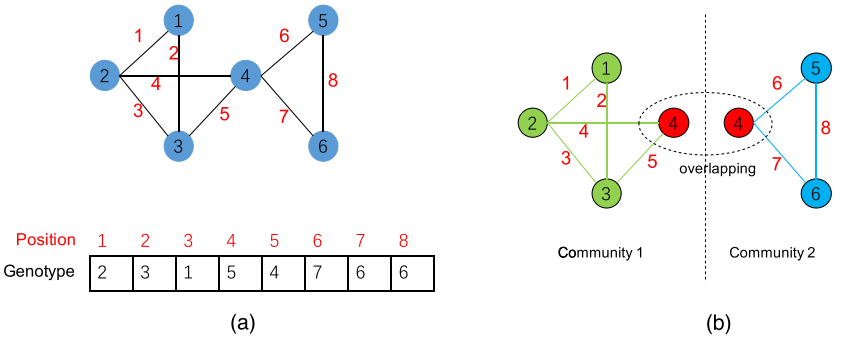

Fig. 4. Example of the genetic representation. (a) A simple network and its possible genotype. (b) The corresponding decoded partition.

As for the Decode() function, it is used to decode link community to node community. Clearly, Algorithm 1 shows that Meme-Link has two parts: the preprocessing part and the evolution part. Thus, the time complexity of Meme-Link is determined by these two parts. Suppose there are m edges and n nodes in a line graph. The operation in NodeToWeightedEdge() is related to the number of edges, so its time complexity is $O(m)$. The operation in InitialPopulation() is related to the number of individuals and the number of nodes, so its time complexity is $O(Popsize \times n)$. As for the evolution part, the time complexity of crossover and mutation operators is related to the number of nodes, and can be realized in linear time, namely, $O(n)$. The fitness evaluation function is the most time-consuming process. In every generation, we should calculate the objective value of every individual. The calculating procedure has the complexity $O(m)$, and the decoding process has the complexity $O(n)$. As a consequence, the time complexity of fitness evaluation is $O(Gen \times Popsize \times (m + n))$. The LocalSearch() procedure has the complexity $O(Gen \times n \times m \times T)$. (T is the number of neighbors of each node, usually very small.). In the following, detailed descriptions of every part are given.

3.1. Representation and initialization

Our clustering algorithm uses the locus-based adjacency representation proposed in [24]. Different from those node-based genotypes, Meme-Link encodes links of networks. In this graph-based representation, an individual of the population consists of L genes $g_1, \ldots, g_L$ and each gene can assume allele value j in the range $\{1, 2, \ldots, L\}$. Genes and alleles represent edges of the weighted line graph $\boldsymbol{W}(\boldsymbol{G})$ modeling a network $\mathcal{N}$, and a value j assigned to the ith gene is interpreted as edge i and j is connected. This means that in the clustering solution found i and j will be in the same cluster. A decoding step, however, is necessary to identify all the separate components of the corresponding graph. The edges located at the same component are assigned to one cluster. An example is shown in Fig. 4, using a simple backtracking scheme, this decoding step can be performed in a linear time [25]. After obtaining link communities, overlapping communities can be obtained by gathering the nodes incident to the edges in each link community. The main advantage of this representation is that the number of clusters k is automatically determined by the number of components contained in an individual in the decoding step.

The initialization process takes the effective connections of the links in the network into account. Conducting the roulette selection based on edge similarity, a random individual is generated. This can speed up the algorithm convergence and reduce the computational cost. See Fig. 3(b), edge e_{13} connects with four other edges, e_{12}, e_{23}, e_{35} and e_{34}, so the allele value of e_{13} can be e_{12}, e_{23}, e_{35} or e_{34}. The similarity is 1, 1, 1/3 and 1/3 respectively. We decide the allele value of e_{13} by conducting the roulette selection based on edge similarity.

3.2. Genetic operators

Uniform Crossover. Meme-Link uses a standard uniform crossover operator. First, a crossover mask of length L, i.e., the number of edges, is randomly generated. Each value on the mask is either 0 or 1. An offspring is generated by selecting the gene from the first parent when the mask value is 0, and from the second parent when the mask value is 1. The main motivation of using the uniform crossover is that it guarantees the maintenance of effective connections of nodes in the networks of child individuals. In fact, because of the biased initialization, if gene i contains value j, link (i, j) exists. Since the child at each position i contains value j coming from one of the two parents, then link (i, j) exists. Fig. 5 shows an example of uniform crossover. The mask is 1 at the first position, so the allele value of offspring at the first position chooses from parent2, which is 3. The mask is 0 at the second position, so the allele value of offspring at the second position chooses from parent1 , which is 3.

Mutation. The mutation operator that randomly changes the value of the ith gene causes a useless exploration of the search space, because the two edges may be not connected in the graph. Thus, in our mutation operator, an edge from the neighbors

Parent1	2	3	1	5	4	7	6	6
Parent2	3	5	4	5	6	7	8	6
Mask	1	0	1	1	0	0	0	1
Offspring	3	3	4	5	4	7	6	6

Fig. 5. An example of uniform crossover.

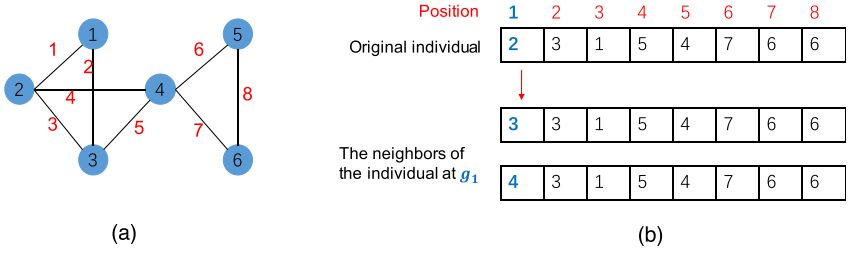

Fig. 6. Example of the neighbors at g_1. A value j assigned to the ith gene is interpreted as edge i and j is connected. (a) A simple network. (b) An individual and its neighbors at g_1.

is selected based on the roulette wheel selection according to their similarity. For example, considering the network of Fig. 4(a), the allowed allele values of the gene in the sixth position are 7, 8. This mutation guarantees the generation of a mutated child in which each node is linked only with one of its neighbors.

3.3. Objective function

Although several evaluation criteria for overlapping communities have been proposed, most of them are based on node communities. Here we give a brief introduction to the quantitative function of modularity density proposed by Li et al. [10]. The difference is that we use it for the link communities on weighted line graphs, not node communities. We consider a weighted line graph $\boldsymbol{W}(\boldsymbol{G}) = (\boldsymbol{V}, \boldsymbol{E})$ with $|\boldsymbol{V}| = L$ nodes and $|\boldsymbol{E}| = e$ edges. In fact, the nodes in the line graph correspond to the edges in the node graph. The adjacent matrix of the weighted line graph is $\boldsymbol{S}$. If V_1 and V_2 are two disjoint subsets of $\boldsymbol{V}$, we define $\boldsymbol{L}(V_1, V_2) = \sum_{\alpha\in V_1,\beta\in V_2} S_{\alpha\beta}$, $\boldsymbol{L}(V_1, V_1) = \sum_{\alpha\in V_1,\beta\in V_1} S_{\alpha\beta}$, and $\boldsymbol{L}\left(V_1, \overline{V_1}\right) = \sum_{\alpha\in V_1,\beta\in \overline{V_1}} S_{\alpha\beta}$, where $\overline{V_1} = V - V_1$. Given a partition $\Omega = \{V_1, V_2, \ldots, V_m\}$ of the graph, where V_i is the vertex set of subgraph $\boldsymbol{W}(\boldsymbol{G})_{\boldsymbol{i}}$ for $i = 1, 2, \ldots, m$, the modularity density is then defined as

$$D = \sum_{i=1}^{m} \frac{L(V_i, V_i) - L\left(V_i, \overline{V}_i\right)}{|V_i|}. \tag{2}$$

In this equation, each summand means the ratio between the difference of the internal and external degrees of subgraph $\boldsymbol{W}(\boldsymbol{G})_{\boldsymbol{i}}$ and the size of the subgraph. The larger the value of $\boldsymbol{D}$ is, the more accurate a partition is. So the community detection problem can be viewed as a problem of finding a partition of a network such that its modularity density $\boldsymbol{D}$ is maximized.

Li et al. also proved the equivalence of modularity density and kernel k means, and proposed a more general modularity density measure

$$D_\lambda = \sum_{i=1}^{m} \frac{2\lambda L(V_i, V_i) - 2(1-\lambda)L(V_i, \overline{V}_i)}{|V_i|}. \tag{3}$$

When $\lambda = 1$, $\boldsymbol{D}_\lambda$ is equivalent to the ratio association; when $\lambda = 0$, $\boldsymbol{D}_\lambda$ is equivalent to the ratio cut; when $\lambda = 0.5$, $\boldsymbol{D}_\lambda$ is equivalent to the modularity density $\boldsymbol{D}$. So the general modularity density $\boldsymbol{D}_\lambda$ can be viewed as a combination of the ratio association and the ratio cut. Generally, optimizing the ratio association algorithm often divides a network into small communities, while optimization of the ratio cut often divides a network into large communities. This general modularity density $\boldsymbol{D}_\lambda$, which is a convex combination of these two indexes, can avoid the resolution limits. In other words, by varying the value of λ, we can use this general function to analyze the topological structure and uncover more detailed and hierarchical organization of complex networks [10]. In this paper, Meme-Link employs $\boldsymbol{D}_\lambda$ as the objective function.

3.4. Local search

We first define the neighbors of an individual. Given an individual of the population consists of L genes $g_1, \ldots, g_L$, change the allele value j of g_i to other edges connected with g_i in turn. The new individuals are called neighbors of the original

individual. As shown in Fig. 6, edge e_{12} connects with edge e_{13}, e_{23}, and e_{24}, so the neighbors of the original individual are those shown in Fig. 6(b).

Algorithm 2: Local search procedure

Input: $\boldsymbol{P}_{child}$

Output: $\boldsymbol{P}_{child}$ after local search.

$\boldsymbol{P}_{now}$ ←FindBest($\boldsymbol{P}_{child}$);
for i =1 ***to*** L ***do***
  Neighbors←FindNeighbors($\boldsymbol{P}_{now}$, $\boldsymbol{g}_i$, γ);
  $\boldsymbol{P}_{next}$ ← FindBest (*Neighbors*);
  if (Evaluate($\boldsymbol{P}_{next}$) > Evaluate ($\boldsymbol{P}_{now}$)) ***then***
    $\boldsymbol{P}_{now}$ ←$\boldsymbol{P}_{next}$;
  end if;
end for.

The local search procedure used in Meme-Link is a hill-climbing strategy. The details of local search operator are shown in Algorithms 2. It is an iterative algorithm that begins with any problem-solving way and then attempts to find a better solution by gradually changing the individual elements of the solution. If the change produces a better solution, the new solution will be incrementally changed and repeated until further improvements are not found. Here we apply this optimization technique to a partition of the network. In Algorithm 2, the FindBest() procedure is responsible for evaluating the fitness of each chromosome in the input population, and returning the chromosome with the maximum fitness, on which the local search procedure will be performed. The Evaluate() procedure is used to evaluate the fitness of a solution. The FindNeighbors() function is responsible for finding the neighbors of the individual at g_i, parameter γ is used to control the number of neighbors according to their similarity. In some large-scale network optimization problems [26], this can reduce the consumption of computing resources. In this paper, we set $\gamma = 0.3$ when the number of neighbors at g_i is greater than 10, and $\gamma = 0.6$ otherwise.

4. Experiments and results

4.1. Benchmark networks

To validate the performance of Meme-Link for overlapping community detection, we conduct extensive experiments on the LFR benchmark [27], which is a special case of the planted l-partition model, but characterized by heterogeneous distributions of node degrees and community sizes.

In our experiments, we use both general and sparse networks. For general networks, we use the ones with $n = 1000$. The average degree is kept at $\overline{k} = 8$. The rest parameters are set as follows: node degrees and community sizes are governed by the power law distributions, with exponents 2 and 1, respectively; the maximum degree is 50; the community size varies between 20 and 80; the mixing parameter μ varies from $\{0.1, 0.3\}$, which is the expected fraction of links of a node connecting it to other communities. The degree of overlapping is determined by parameters O_n (i.e., the number of overlapping nodes) and O_m (i.e., the number of communities to which each overlapping node belongs). We fix the former to 30% of the total number of nodes. The latter, the most important parameter for our test, varies from 2 to 6 indicating the diversity of overlapping nodes. Harder detection tasks will be created by increasing the value of O_m.

Most of real networks are sparse [28–30], satisfying $\overline{k} \ll \ln N \ll N$ or $\overline{k} \sim \ln N$. $\overline{k}$ is the average degree of the network and N is the number of nodes in the network. So, sparse networks are very common and important. For sparse networks, we used network with size $n = 1000$. The average degree is kept at $\overline{k} = 5$. Node degrees and community sizes are governed by the power law distributions, with exponents 2 and 1, respectively; the maximum degree is 30; the community size varies between 20 and 50; the mixing parameter μ is 0.1. O_n is set to 10%, and O_m varies from 2 to 6.

We compare Meme-Link with three well-known algorithms, including CFinder (the implementation of clique propagation algorithm [4]), MOSES [6] (an algorithm expands a community from edges), and SLPA [3] (a label propagation algorithm). For algorithms with tunable parameters, the results with the best setting are reported. Parameters for those algorithms were set as follows. For CFinder, k varied from 3 to 10; for SLPA, parameter r varies from 0.05 to 0.5 in the step of 0.05. For Meme-Link, we set the population size to 200, the number of generations to 10, the crossover rate to 0.6, and the mutation rate to 0.4. λ in (3) is set to 0.3 in general networks and 0.1 in sparse networks.

4.2. Experiments on general networks

4.2.1. Identifying overlapping nodes

Community overlapping manifests itself as the existence of nodes with membership in multiple communities. Thus, we refer to nodes with multiple memberships as overlapping nodes. In real-world social networks, such nodes are important

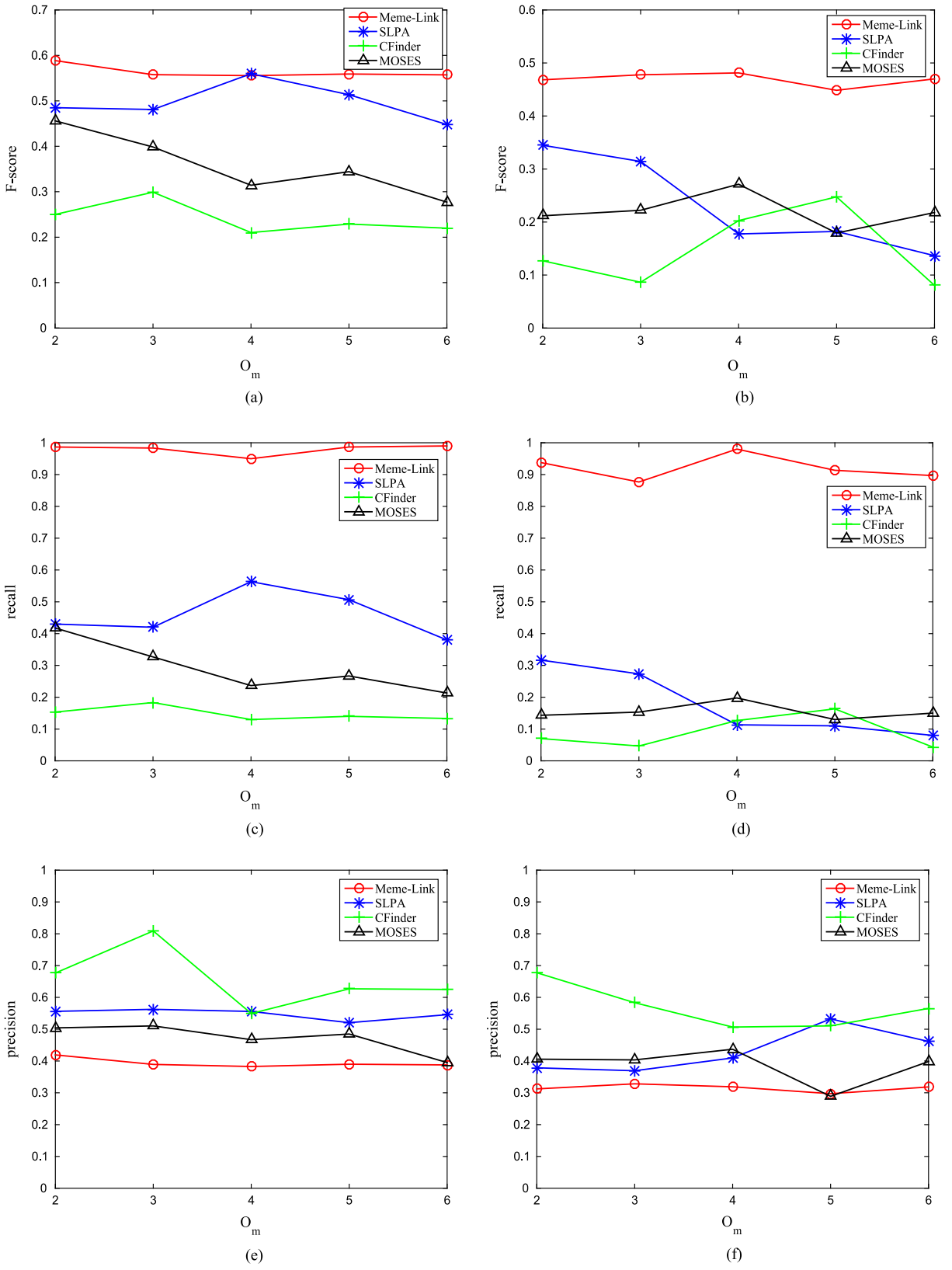

Fig. 7. Evaluations of overlapping node detection on general networks. (a) F-score for networks with $\mu = 0.1$. (b) F-score for networks with $\mu = 0.3$. (c) *Recall* for networks with $\mu = 0.1$. (d) *Recall* for networks with $\mu = 0.3$. (e) *Precision* for networks with $\mu = 0.1$. (f) *Precision* for networks with $\mu = 0.3$.

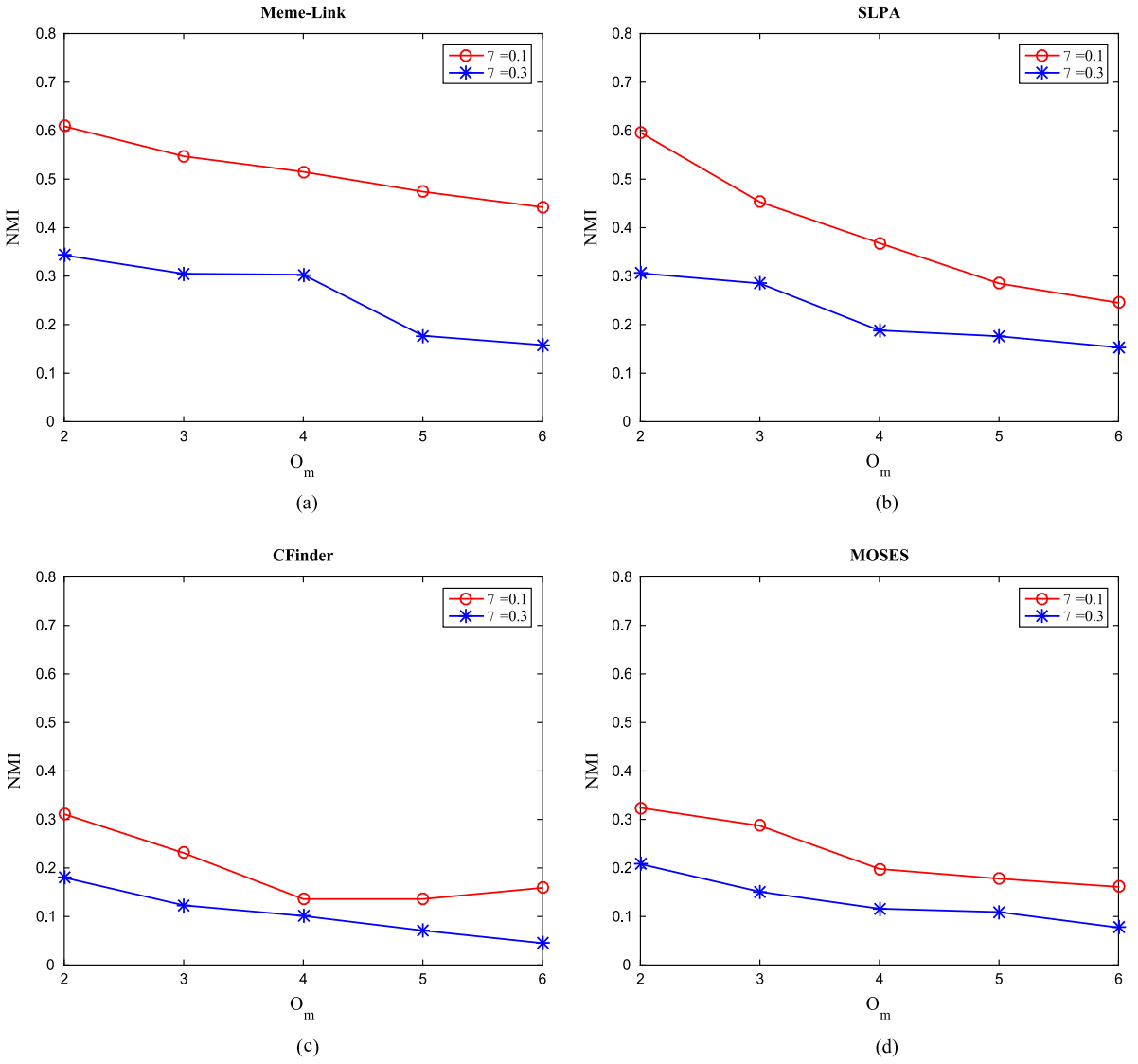

Fig. 8. The effects of mixing parameter μ on LFR networks. (a) *NMI* for Meme-Link with $\mu = 0.1$ and $\mu = 0.3$. (b) *NMI* for SLPA with $\mu = 0.1$ and $\mu = 0.3$. (c) *NMI* for CFinder with $\mu = 0.1$ and $\mu = 0.3$. (d) *NMI* for MOSES with $\mu = 0.1$ and $\mu = 0.3$.

because they usually represent bridges between communities. For this reason, the ability to identify overlapping nodes, although often neglected, is essential for assessing the accuracy of community detection algorithms.

Note that the number of overlapping nodes alone is not sufficient to quantify the detection performance. To provide more precise analysis, we define the identification of overlapping nodes as a binary classification problem. We use F-score as a measure of accuracy, which is the harmonic mean of *precision* (i.e., the number of correctly detected overlapping nodes divided by the total number of detected overlapping nodes) and *recall* (i.e., the number of correctly detected overlapping nodes divided by the true number of overlapping nodes). The experimental results are shown in Fig. 7.

Fig. 7 shows that Meme-Link achieves the largest F-score and *recall* in networks with different levels of mixture, as defined by μ. It is worth noting that Meme-Link almost finds all overlapping nodes. Link-based community detection algorithms are used exclusively for solving overlapping community problems, so they tend to find more overlapping nodes than expected [31]. This also leads to the low *Precision* of Meme-Link. Although CFinder has a higher *Precision*, only a small amount of overlapping nodes can be detected.

4.2.2. Identifying overlapping communities

Most measures for quantifying the quality of a partition are not suitable for a cover produced by overlapping detection algorithms. We adopt the extended normalized mutual information (*NMI*) proposed by Lancichinetti [19]. *NMI* yields the values between 0 and 1, with 1 corresponding to a perfect matching.

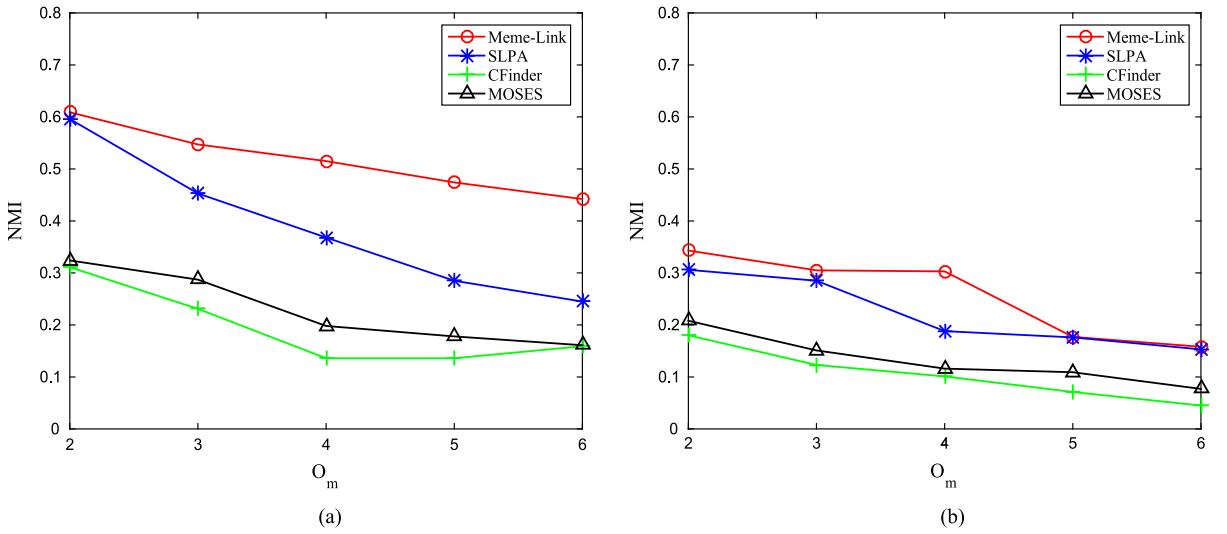

Fig. 9. Comparison in terms of *NMI*. (a) *NMI* for networks with $\mu = 0.1$. (b) *NMI* for networks with $\mu = 0.3$.

In general, changes in the network topology, especially the mixing value μ, have a similar impact on the disjoint community detection. That is, the larger the value of μ, the worse the results produced by detection algorithms are (i.e., blue curve $<$ red curve in Fig. 8) due to the fact that the connections inside communities are weak for larger μ. On the other hand, detection performance typically decays as the diversity of overlapping increases (i.e., O_m getting larger). Fig. 9 shows that Meme-Link outperforms or performs similarly to other algorithms on general networks. SLPA has a better performance than CFinder and MOSES. The decrease in *NMI* of Meme-Link is also relatively slow, indicating that Meme-Link is less sensitive to the diversity of O_m.

4.3. Identifying overlapping nodes and communities in sparse networks

The experimental results on sparse networks are similar to those on general networks. Fig. 10(a) shows that Meme-Link achieves the largest F-score in networks. CFinder and MOSES have close performance in the test. It is interesting that SLPA has a positive correlation with O_m while other algorithms typically demonstrate a negative correlation. The F-score of Meme-Link is stable, indicating that Meme-Link is less sensitive to the diversity of O_m in the node level. It is clear that Meme-Link has very high *recall* because it tends to find more overlapping nodes [31].

Fig. 11 shows that Meme-Link performs slightly better than SLPA, and obviously better than other algorithms over different networks. The better performance in the node level (i.e., F-score) helps understand the results. The decrease in *NMI* of Meme-Link is also relatively slow, indicating that Meme-Link is less sensitive to the diversity of O_m. Fig. 12 shows the obtained partitions on the benchmark network. Edges with different colors indicate different communities obtained by Meme-Link and the black nodes are overlapping nodes.

4.4. Experiments on real-life data sets

It is well known that overlapping communities generally exist in social networks and biological networks. From the theoretical point of view, the community of links could be more intuitive than the community of nodes in some real-life networks, since most individuals in the society belong to multiple communities, such as families, co-workers, and friends while the links between a pair of individuals usually exists for a dominant reason. Fig. 13 displays the American college football network. This network represents American football games between Division IA colleges during the regular fall season in 2000, as compiled by Girvan and Newman [11]. Different colors of the edges indicate different communities obtained by Meme-Link. Nodes connected to multiple edges of different colors are overlapping nodes. Fig. 14 displays a protein–protein interaction network which contains a human MAP kinase interactome published in [32]. Edges with different colors indicate different kinds of protein–protein interactions. Meme-Link can be used to predict the function of a single protein and to discover novel modules by clustering on the edges.

5. Conclusions

In this paper, we propose a link-based algorithm for overlapping community detection. Different from those node-based overlapping community detection algorithms, Meme-Link utilizes the property of the unique role of links and applies a novel

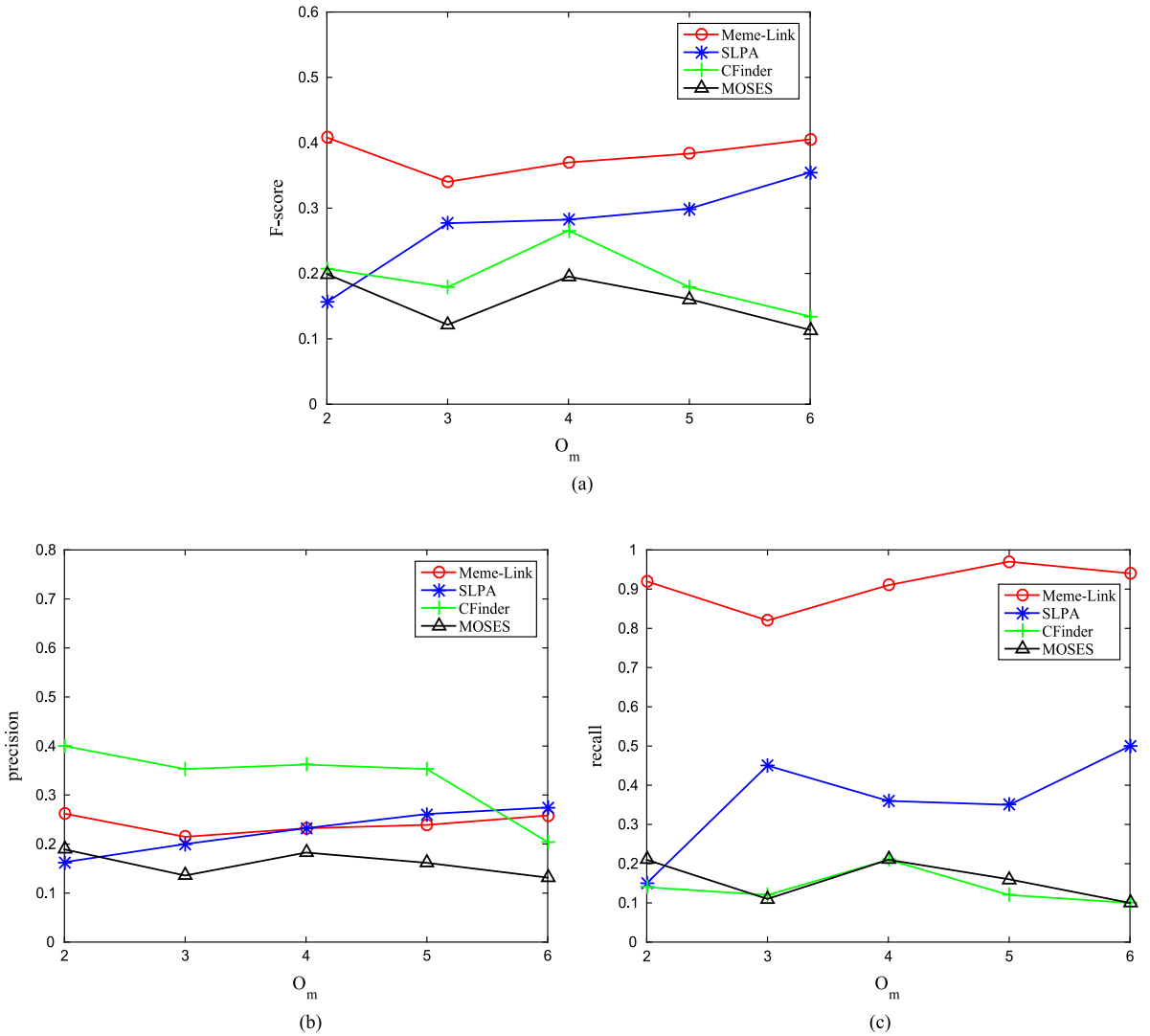

Fig. 10. Evaluations of overlapping node detection on sparse networks. (a) F-score for networks. (b) *Precision* for networks. (c) *Recall* for networks.

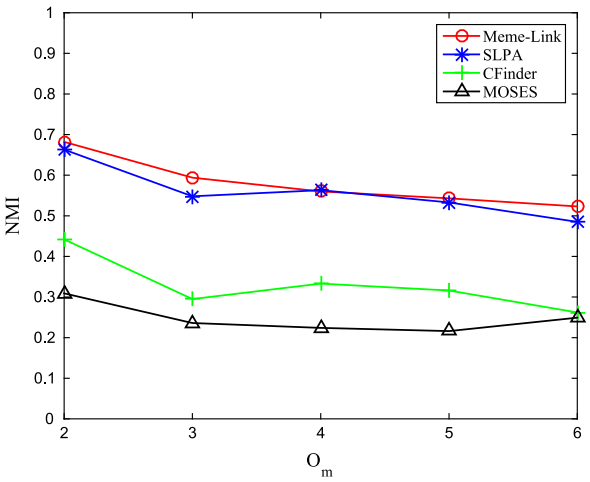

Fig. 11. *NMI* of different algorithms on sparse networks.

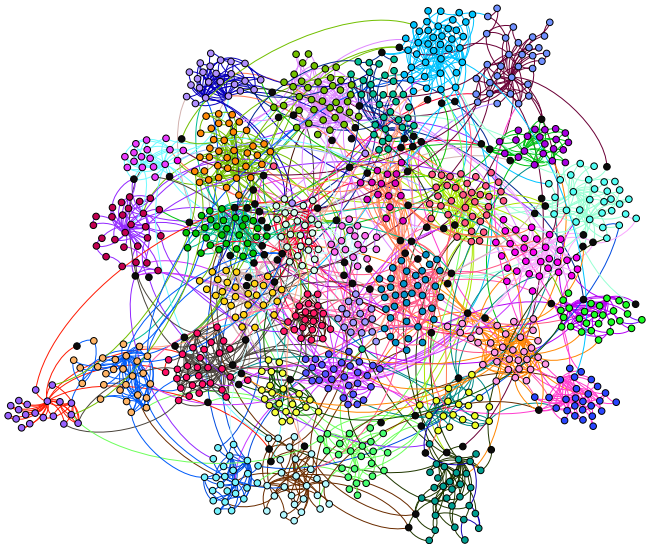

Fig. 12. The obtained partition on the benchmark network. Different colors represent different communities. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

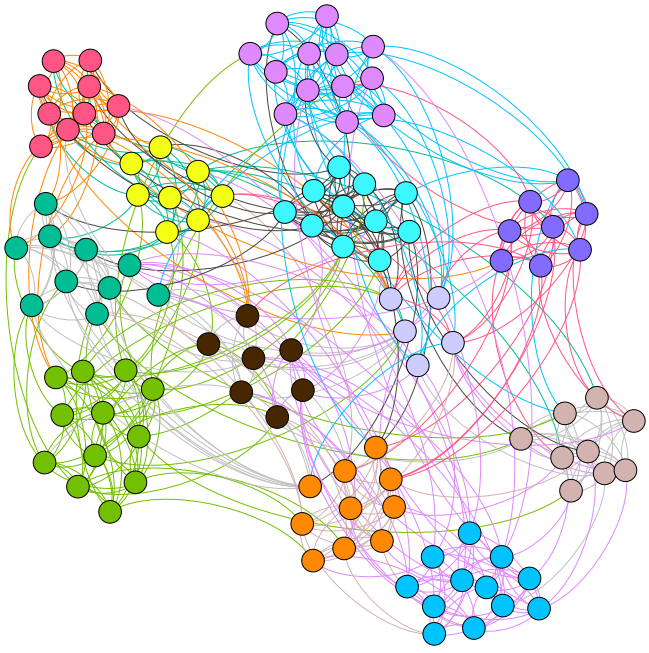

Fig. 13. The obtained partition on the American college football network.

memetic algorithm to cluster edges. The genetic representation and the corresponding operators effectively represent the link communities and determine the number of communities automatically. Experiments show that Meme-Link outperforms or performs similarly to the state-of-the-art algorithms on both general and sparse networks. However, it is known that memetic algorithms require high execution times when large populations of individuals are used. The number of edges is often several times over the number of nodes in the network, so the link community detection algorithms can be more time-consuming. On the other hand, memetic algorithms are naturally suited to be implemented on parallel architectures. It is necessary to realize the algorithm on a parallel machine, which will be studied in our future work.

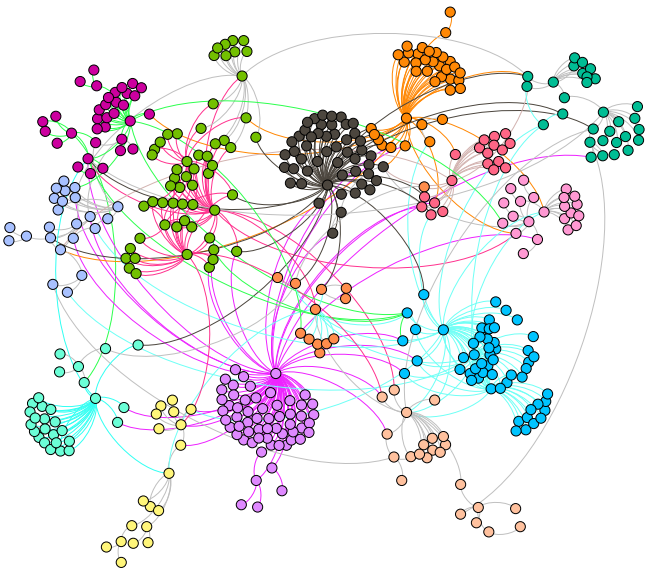

Fig. 14. The obtained partition on the protein–protein interaction network.

Acknowledgments

This work was supported in part by the Outstanding Young Scholar Program of National Natural Science Foundation of China (NSFC) under Grant 61522311, in part by the General Program of NSFC under Grant 61773300, and in part by the Key Program of Fundamental Research Project of Natural Science of Shaanxi Province, China under Grant 2017JZ017.

References

[1] S. Kelley, M. Goldberg, M. Magdon-Ismail, K. Mertsalov, A. Wallace, Defining and discovering communities in social networks, in: Handbook of Optimization in Complex Networks, Springer, US, 2012, pp. 139–168.
[2] F. Reid, A. McDaid, N. Hurley, Partitioning breaks communities, in: Mining Social Networks and Security Informatics, Springer, Netherlands, 2013, pp. 79–105.
[3] J. Xie, B.K. Szymanski, X. Liu, Slpa: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process, in: 2011 IEEE 11th International Conference on Data Mining Workshops, ICDMW, 2011, pp. 344–349.
[4] G. Palla, I. Derényi, I. Farkas, T. Vicsek, Uncovering the overlapping community structure of complex networks in nature and society, 2005. arXiv preprint physics/0506133.
[5] S. Gregory, Finding overlapping communities in networks by label propagation, New J. Phys. 12 (10) (2010) 103018.
[6] A. McDaid, N. Hurley, Detecting highly overlapping communities with model-based overlapping seed expansion, in: 2010 International Conference on IEEE Advances in Social Networks Analysis and Mining, ASONAM, 2010, pp. 112–119.
[7] C. Liu, J. Liu, Z. Jiang, A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks, IEEE Trans. Cybernet. 44 (12) (2014) 2274–2287.
[8] P. Moscato, On evolution search optimization genetic algorithms and martial arts: Towards memetic algorithms, Caltech concurrent computation program, C3P Report, 1989, p. 826.
[9] W. Du, B. Liang, G. Yan, O. Lordan, X. Cao, Identifying vital edges in Chinese air route network via memetic algorithm, Chinese J. Aeronaut. 30 (1) (2017) 330–336.
[10] Z. Li, S. Zhang, R.S. Wang, X.S. Zhang, L. Chen, Quantitative function for community detection, Phys. Rev. E 77 (3) (2008) 036109.
[11] M. Girvan, M.E. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. 99 (12) (2002) 7821–7826.
[12] Z. Jiang, J. Liu, S. Wang, Traveling salesman problems with PageRank distance on complex networks reveal community structure, Physica A 463 (2016) (2016) 293–302.
[13] M.E. Newman, Fast algorithm for detecting community structure in networks, Phys. Rev. E 69 (6) (2004) 066133.
[14] Z. Li, J. Liu, A multi-agent genetic algorithm for community detection in complex networks, Physica A 449 (2016) 336–347.
[15] M.E. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.
[16] Y. Li, J. Liu, C. Liu, A comparative analysis of evolutionary and memetic algorithms for community detection from signed social networks, Soft Comput. 18 (2) (2014) 329–348.
[17] M.E. Newman, Modularity and community structure in networks, Proc. Natl. Acad. Sci. 103 (23) (2006) 8577–8582.
[18] C. Pizzuti, Overlapped community detection in complex networks, in: Proceedings of the 11th Annual conference on Genetic and evolutionary computation, ACM, 2009, pp. 859–866.
[19] A. Lancichinetti, S. Fortunato, J. Kertész, Detecting the overlapping and hierarchical community structure in complex networks, New J. Phys. 11 (3) (2009) 033015.
[20] I. Derényi, G. Palla, T. Vicsek, Clique percolation in random networks, Phys. Rev. Lett. 94 (16) (2005) 160202.
[21] Y.Y. Ahn, J.P. Bagrow, S. Lehmann, Link communities reveal multiscale complexity in networks, 2009. arXiv preprint arXiv:0903.3178.
[22] T.S. Evans, R. Lambiotte, Line graphs, link partitions, and overlapping communities, Phys. Rev. E 80 (1) (2009) 016105.

[23] H. Whitney, Congruent graphs and the connectivity of graphs, in: Hassler Whitney Collected Papers, Birkhäuser, Boston, 1992, pp. 61–79.
[24] Y. Park, M. Song, A genetic algorithm for clustering problems, in: Proceedings of the Third Annual Conference on Genetic Programming, 1998, pp. 568–575.
[25] J. Handl, J. Knowles, An evolutionary approach to multiobjective clustering, IEEE Trans. Evol. Comput. 11 (1) (2007) 56–76.
[26] X. Peng, Y. Wu, Large-scale cooperative co-evolution using niching-based multi-modal optimization and adaptive fast clustering, Swarm Evol. Comput. (2017).
[27] A. Lancichinetti, S. Fortunato, F. Radicchi, Benchmark graphs for testing community detection algorithms, Phys. Rev. E 78 (4) (2008) 046110.
[28] M. Andrecut, S. Huang, S.A. Kauffman, Heuristic approach to sparse approximation of gene regulatory networks, J. Comput. Biol. 15 (9) (2008) 1173–1186.
[29] A. Arenas, A. Diaz-Guilera, C.J. Pérez-Vicente, Synchronization reveals topological scales in complex networks, Phys. Rev. Lett. 96 (11) (2006) 114102.
[30] M. Boguná, R. Pastor-Satorras, Class of correlated random networks with hidden variables, Phys. Rev. E 68 (3) (2003) 036112.
[31] J. Xie, S. Kelley, B.K. Szymanski, Overlapping community detection in networks: The state-of-the-art and comparative study, ACM Comput. Surveys 45 (4) (2013) 43.
[32] Bandyopadhyay Sourav, et al., A human MAP kinase interactome, Nat. Methods 7 (10) (2010) 801–805.