A Multiobjective Evolutionary Algorithm Based on Similarity for Community Detection from Signed Social Networks

Chenlong Liu, Jing Liu, *Member, IEEE,* and Zhongzhou Jiang

***Abstract*—Various types of social relationships, such as friends and foes, can be represented as signed social networks (SNs) that contain both positive and negative links. Although many community detection (CD) algorithms have been proposed, most of them were designed primarily for networks containing only positive links. Thus, it is important to design CD algorithms which can handle large-scale SNs. To this purpose, we first extend the original similarity to the signed similarity based on the social balance theory. Then, based on the signed similarity and the natural contradiction between positive and negative links, two objective functions are designed to model the problem of detecting communities in SNs as a multiobjective problem. Afterward, we propose a multiobjective evolutionary algorithm, called MEA$_s$-SN. In MEA$_s$-SN, to overcome the defects of direct and indirect representations for communities, a direct and indirect combined representation is designed. Attributing to this representation, MEA$_s$-SN can switch between different representations during the evolutionary process. As a result, MEA$_s$-SN can benefit from both representations. Moreover, owing to this representation, MEA$_s$-SN can also detect overlapping communities directly. In the experiments, both benchmark problems and large-scale synthetic networks generated by various parameter settings are used to validate the performance of MEA$_s$-SN. The experimental results show the effectiveness and efficacy of MEA$_s$-SN on networks with 1000, 5000, and 10 000 nodes and also in various noisy situations. A thorough comparison is also made between MEA$_s$-SN and three existing algorithms, and the results show that MEA$_s$-SN outperforms other algorithms.**

***Index Terms*—Community detection problems, direct representation, indirect representation, multiobjective evolutionary algorithms, signed social networks, similarity.**

I. Introduction

NETWORKS are employed in many fields to represent various kinds of complex systems [1]–[2], and social networks have attracted much attention. To understand and utilize the information in social networks, research has found many distinctive network properties like the small-world and scale-free ones [3]–[5], and developed various methods to capture the network structure characteristics from different perspectives. The research on analyzing the community structure has drawn a great deal of attention during the past decade [6]–[28]. Although many community detection (CD) methods have been proposed, most of them can only handle networks without negative links, namely, unsigned networks. However, many complex systems in social world can be modeled as networks with both positive and negative links, namely, signed networks (SNs).

In fact, SNs have been widely used to represent various types of social relationships. For example, the technology news website Slashdot lets its users tag other users as friends and foes, as well as the product site Epinions that allows users to trust and distrust each other. Guha *et al*. [29] addressed the problem of predicting the trust between any two people in a social network connected by trust/distrust scores. In addition to social networks, many biological networks are also signed. For instance, the interactions between genes in gene regulatory networks can be enhanced or repressed. Therefore, SNs can represent more general relationships between individuals in social or biological networks. Positive links denote friendship, cooperation, trust, enhancing, etc., while negative links denote hostility, dislike, distrust, repressing, etc. Moreover, Kunegis *et al*. [30] in their recent work showed that negative links have a measurable added value for social networks.

Due to the importance of community structure as a topological property of social networks, methods that can detect communities from SNs are hardly needed. In unsigned networks, community structure is defined as a group of nodes or vertices which have dense connections within groups and sparse connections between groups, whereas for SNs, communities are defined not only by the density of links but also by the signs of links. That is, within communities, the links should be positive and dense, and between communities, the links should be negative or positive and sparse. But this problem is by no means straightforward since it is natural to have some negative links within groups and, at the same time, some positive links between groups. Also, nodes connected by positive links do not belong to the same community, either. Thus, more robust community partitions should properly disregard and retain some positive and negative links to identify more natural communities [31].

Manuscript received May 19, 2013; revised October 11, 2013 and January 7, 2014; accepted January 29, 2014. Date of publication March 4, 2014; date of current version November 13, 2014. This work was supported in part by the National Natural Science Foundation of China under Grant 61103119 and Grant 61271301, in part by the Research Fund for the Doctoral Program of Higher Education of China under Grant 20130203110010, and in part by the Fundamental Research Funds for the Central Universities under Grant K5051202052. This paper was recommended by Associate Editor Y. Jin.

The authors are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an 710071, China. For additional information regarding this paper, please contact Jing Liu (corresponding author, e-mail: neouma@163.com).

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

Digital Object Identifier 10.1109/TCYB.2014.2305974

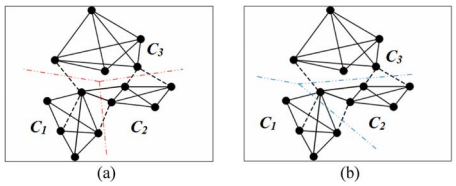


Fig. 1. Two partitions over one SN. Solid lines are positive and dashed lines are negative. (a) Partition which has more positive links within communities. (b) Partition which has more negative links between communities.

A. Our Contribution

With the intrinsic properties of detecting communities from SNs in mind, we first model this problem as a multiobject problem (MOP) considering both link density and signs. Two objectives are proposed to reflect the contradiction between positive and negative links. One is target to put all positive links in communities while the other is target to keep all negative links between communities. Fig. 1 shows a simple SN which can be divided into two kinds of community structures. The partition in Fig. 1(a) has 23 positive links within communities and four negative links between communities while that in Fig. 1(b) has 22 positive links within communities and five negative links between communities. So the former has more positive links within communities and the latter has more negative links between communities. In fact, both of these two partitions are of significance and acceptable. Thus, these two objectives should be optimized together.

Evolutionary algorithms (EAs) are the most popular method for solving MOPs, which are a kind of stochastic global optimization methods inspired by the biological mechanism of evolution and heredity, and have been successfully used to solve various problems [32]–[37]. Especially, Shi *et al.* [38] concluded that multiobjective EAs optimizing over a pair of negatively correlated objectives usually perform better than the single-objective EAs optimizing over either of the original objectives. Therefore, we propose an MOEA based on the signed similarity for detecting communities from SNs (MEA$_s$-SN). The signed similarity is extended from the original similarity based on the social balance theory so that it can deal with the effect of introducing negative links.

Especially, a direct and indirect combined representation is designed so that MEA$_s$-SN can switch between different representations during the evolutionary process, and thus benefit from both representations. Also, owing to this representation, MEA$_s$-SN can detect both separated and overlapping communities. A set of rigorous experiments are conducted on both benchmark and large-scale synthetic networks with 1000, 5000, and 10 000 nodes. The results show the effectiveness and efficacy of MEA$_s$-SN. A thorough comparison is also made between MEA$_s$-SN and three existing algorithms, and the results show that MEA$_s$-SN outperforms other algorithms. Moreover, MEA$_s$-SN requires no prior knowledge on the community structure, such as the number of communities and a good initial partition. Thus, MEA$_s$-SN is easy to use.

The rest of this paper is organized as follows. Section II introduces the objective functions designed for CD problems in SNs. Section III describes the details of MEA$_s$-SN. The experiments on benchmark and synthetic networks are given in Section IV. Section V discusses the related work on social network analysis and community detection methods. Finally, the conclusions are given in Section VI.

II. Objective Functions for CD from SNs

Given a signed network $G=(V, E, w)$. $V=\{v_1, v_2, ..., v_n\}$ is the set of nodes, $E\subseteq V\times V=\{(v_i, v_j) \mid v_i, v_j \in V \text{ and } i\neq j\}$ is the set of edges, and $w(v_i, v_j)$ is the weight of the edge between nodes v_i and v_j. The weight can be larger than 0 (positive relationship) or smaller than 0 (negative relationship). Let $\boldsymbol{C}=\{C_1, C_2, ..., C_m\}$ be a set of communities in G; that is, $C_i \subset V$ for $i=1, 2, \ldots, m$. The problem of CD from SNs can be accurately expressed in (1), shown at the bottom of the page.

Huang *et al.* [14] used a structural similarity to denote the local connectivity density of any two adjacent nodes in a weighted undirected network. Given a weighted undirected network, the similarity $s(u, v)$ for $u, v\in V$ is defined as follows [14]:

$$s(u,v)=\frac{\sum\limits_{x\in\Gamma(u)\cap\Gamma(v)} w(u,x)\cdot w(v,x)}{\sqrt{\sum\limits_{x\in\Gamma(u)} w^2(u,x)}\cdot\sqrt{\sum\limits_{x\in\Gamma(v)} w^2(v,x)}} \tag{2}$$

where $\Gamma(y)$, $y\in V$ is defined as the set of node y and y's neighbors; that is

$$\Gamma(y)=\{v\in V \mid (y,v)\in E\}\cup\{y\}. \tag{3}$$

However, this structural similarity is designed only for unsigned networks, which can not handle negative links. Therefore, based on the social balance theory [39], [40], we first extend this similarity to a signed similarity. The social balance theory suggests that people in a social network tend to form into a balanced network structure, that is, for a triad, either all three of these users are friends or only one pair of them is friends. Thus, based on this theory, suppose we already know the relationship between two pairs of users, we can infer the relationship between the third pair of users.

Given three users, labeled as a, b, and c, and the relationships between a and c, b and c. Fig. 2 shows the three possible cases. The links are used to indicate the relationship between two users. Solid lines mean friends (positive links) and dashed lines mean foes (negative links). For the case in Fig. 2(a), both a and c, b and c are friends, to be balanced, a and b should be friends. For the case in Fig. 2(b), a and c are friends, but b and c are foes, to be balanced, a and b should be foes. For the case in Fig. 2(c), both a and c, b and c are foes, to be balanced, a and b should be friends. But this will be the same to the case in Fig. 2(b), so we just suppose no relationship between a and b. Therefore, the signed similarity measure should be larger than 0 for the case in Fig. 2(a), smaller than 0 for the case in

$$\begin{cases} w(v_i,v_j)>0, & (v_i,v_j)\in E\wedge(v_i\in C_l)\wedge(v_j\in C_l) \\ w(v_i,v_j)<0, & (v_i,v_j)\in E\wedge(v_i\in C_l)\wedge(v_j\in C_k)\wedge(l\neq k) \end{cases}, l,k=1,2,...,m. \tag{1}$$

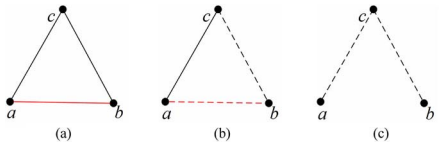


Fig. 2. Example for three users' relationships. Solid lines denote friends and dashed lines denote foes. (a) Both a, c and b, c are friends. (b) a, c are friends and b, c are foes. (c) Both a, c and b, c are foes.

Fig. 2(b), and equal to 0 for the case in Fig. 2(c). Thus, the signed similarity is defined as

$$s_{\text{signed}}(u,v)=\frac{\sum\limits_{x\in\Gamma(u)\cap\Gamma(v)}\psi(x)}{\sqrt{\sum\limits_{x\in\Gamma(u)}w^2(u,x)}\cdot\sqrt{\sum\limits_{x\in\Gamma(v)}w^2(v,x)}} \tag{4}$$

where

$$\psi(x)=\begin{cases}0 & \text{if } w(u,x)<0 \text{ and } w(v,x)<0\\ w(u,x)\cdot w(v,x) & \text{otherwise.}\end{cases} \tag{5}$$

Effective partitions for SNs tend to have more positive links within communities and more negative links between communities. To realize these purposes, the objective functions should be appropriately designed. Usually, the objective functions for MOPs should contradict to each other [41], which is also true for multiobjective CD [38]. Shi *et al.* [38] concluded that multiobjective EAs optimizing over a pair of negatively correlated objectives usually perform better than the single-objective EAs optimizing over either of the original objectives, and even better than other well-established CD approaches. Therefore, with the purposes of CD in SNs in mind, the following two conflict objective functions are designed respectively for maximizing the sum of positive similarities within communities and maximizing the sum of negative similarities between communities

$$\begin{cases}\text{Maximize } f_{\text{pos-in}}(\boldsymbol{C}=\{C_1,C_2,...,C_m\})=\dfrac{1}{m}\displaystyle\sum_{i=1}^{m}\frac{P_{\text{in}}^{C_i}}{P_{\text{in}}^{C_i}+P_{\text{out}}^{C_i}}\\ \text{Maximize } f_{\text{neg-out}}(\boldsymbol{C}=\{C_1,C_2,...,C_m\})=\dfrac{1}{m}\displaystyle\sum_{i=1}^{m}\frac{N_{\text{out}}^{C_i}}{N_{\text{in}}^{C_i}+N_{\text{out}}^{C_i}}\end{cases} \tag{6}$$

where

$$P_{\text{in}}^{C_i}=\sum_{u,v\in C_i\wedge(u,v)\in E}\max\left(s_{\text{signed}}(u,v),0\right) \tag{7}$$

$$P_{\text{out}}^{C_i}=\sum_{u\in C_i\wedge v\in C_j\wedge i\neq j\wedge(u,v)\in E}\max\left(s_{\text{signed}}(u,v),0\right) \tag{8}$$

$$N_{\text{in}}^{C_i}=\sum_{u,v\in C_i\wedge(u,v)\in E}\min\left(s_{\text{signed}}(u,v),0\right) \tag{9}$$

$$N_{\text{out}}^{C_i}=\sum_{u\in C_i\wedge v\in C_j\wedge i\neq j\wedge(u,v)\in E}\min\left(s_{\text{signed}}(u,v),0\right). \tag{10}$$

Equations (7) and (8) define the positive internal and external similarities of a community, while (9) and (10) define the negative internal and external similarities of a community. Although $f_{\text{pos-in}}$ is only related to positive similarities while $f_{\text{neg-out}}$ is only related to negative similarities, these two objectives contradict to each other since one node may connect to both positive and negative links. For example, Fig. 3(a) and (b) gives two different partitions of a network. Both of them divide the network into three communities. In Fig. 3(a), node 10 belongs to C_1 while in Fig. 3(b), it belongs to C_2. Since $f_{\text{neg-out}}$ tries to put all negative links between communities, it prefers the partition in Fig. 3(b) over that in Fig. 3(a). However, when node 10 is put in C_2, the similarity between nodes 10 and 12, 10 and 11 will contribute to P_{out} of C_1 instead of P_{in} of C_1. Thus, it leads to a drop in $f_{\text{pos-in}}$.

III. MEA$_s$-SN

A. Direct and Indirect Combined Representation

The representations have a great effect on both the evolutionary operators can be used and the overall efficiency of the resulting EAs. Usually, in the field of EAs, representations fall into two broad categories: direct and indirect. A direct representation is the natural representation, and can be evaluated easily. An indirect representation is not complete in itself, and a decoder which transforms the solution in the indirect representation into one in the direct representation is required.

In existing literature for CD based on EAs, the character string representation [21] and the locus-based adjacency representation [22] are usually used. These are two direct representations and can be evaluated easily. To detect separated and overlapping communities simultaneously, an indirect representation is proposed in our previous work [13]. Since the decoder implies a heuristic search, this indirect representation can find better candidate solutions. However, since the decoder needs to be executed before evaluating each individual, the computational cost is high. To overcome this defect, we design a direct and indirect combined representation based on the character string representation and the indirect representation proposed in [13]. In this new combined representation, each individual is defined as follows.

Definition 1: An individual, $\boldsymbol{A}$, consists of two components. The first component is a permutation of all nodes in V, labeled as $\boldsymbol{A}\langle\boldsymbol{P}\rangle$

$$\boldsymbol{A}\langle\boldsymbol{P}\rangle=\{v_{\pi_1},v_{\pi_2},...,v_{\pi_n}\} \tag{11}$$

where $(\pi_1, \pi_2, \ldots, \pi_n)$ is a permutation of $(1, 2, \ldots, n)$. The second component is a vector with n elements, labeled as $\boldsymbol{A}\langle\boldsymbol{C}\rangle$

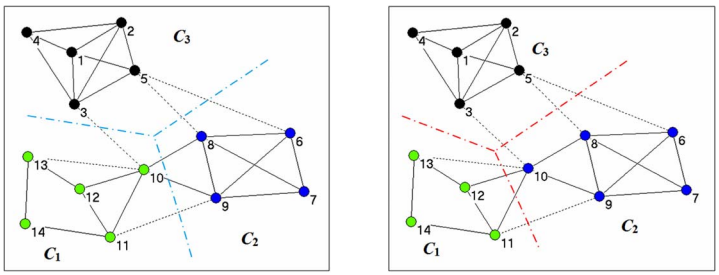
$$\boldsymbol{A}\langle\boldsymbol{C}\rangle=(c_1,c_2,...,c_n) \tag{12}$$

where c_i, $1\le i\le n$ denotes that node v_i belongs to community C_{c_i}. □

Clearly, $\boldsymbol{A}\langle\boldsymbol{P}\rangle$ is the indirect representation part and a decoder is required to transform it to the actual community structure. $\boldsymbol{A}\langle\boldsymbol{C}\rangle$ is the character string representation. Nodes v_i and v_j are in the same community if $c_i=c_j$.

In [13], the concept of community fitness proposed in [15] was used to design the decoder. Here, since we try to find communities based on the signed similarity, we designed a new decoder. In [14], based on the similarity, a quality function of a local community C, namely the tightness, was defined as follows:

$$T(C)=\frac{S_{\text{in}}^{C}}{S_{\text{in}}^{C}+S_{\text{out}}^{C}} \tag{13}$$



(a) (b)

Fig. 3. Two different partitions of a network. $f_{\text{neg-out}}$ prefers (b) since it tries to put all negative links between communities, however, this leads to a drop in $f_{\text{pos-in}}$.

where

$$S_{\text{in}}^{C} = \sum_{u,v \in C \wedge (u,v) \in E} s(u, v) \tag{14}$$

is the internal similarity of community C which is equal to two times of the sum of similarities between any two adjacent vertices both inside C, and

$$S_{\text{out}}^{C} = \sum_{u \in C \wedge v \in C' \wedge C \neq C' \wedge (u,v) \in E} s(u, v) \tag{15}$$

is the external similarity of community C which is equal to the sum of similarities between vertices inside C and vertices outside C. Here, we extend T to the signed situation and propose the signed tightness, which is labeled as T_{signed}

$$T_{\text{signed}}(C) = \frac{P_{\text{in}}^{C} - N_{\text{in}}^{C}}{P_{\text{in}}^{C} - N_{\text{in}}^{C} + P_{\text{out}}^{C}}. \tag{16}$$

Obviously, T_{signed} is equal to T when the network has positive links only.

Let $\boldsymbol{C} = \{C_1, C_2, ..., C_m\}$ be a set of communities and the decoder first initializes $\boldsymbol{C}$ to empty. Then, according to the order of $\boldsymbol{A}\langle \boldsymbol{P} \rangle$, for each node, check whether this node can increase T_{signed} of communities in current $\boldsymbol{C}$; that is, whether (17) can be satisfied

$$T_{\text{signed}}(C_j \cup v_i) > T_{\text{signed}}(C_j) \tag{17}$$

where C_j is one of existing communities and v_i is one node in $\boldsymbol{A}\langle \boldsymbol{P} \rangle$. For detecting separated communities, as long as one community is found to satisfy (17), the process is stopped and this node is added to this community. For detecting overlapping communities, this node is added to all communities that satisfy (17). If no existing community satisfies (17), this node itself forms a new community and this new community is added to $\boldsymbol{C}$. Further, for detecting overlapping communities, two communities with more than half identical nodes; that is, the condition in (18) is satisfied, are merged

$$\forall C_i \neq C_j, \frac{|C_i \cap C_j|}{|C_i|} > \frac{1}{2} \text{or} \frac{|C_i \cap C_j|}{|C_j|} > \frac{1}{2}. \tag{18}$$

Algorithm 1 summarizes the details of this decoder.

Algorithm 1 The decoder

Input: $\boldsymbol{A}\langle \boldsymbol{P} \rangle = \{v_{\pi_1}, v_{\pi_2}, ..., v_{\pi_n}\}$;
Output: $C = \{C_1, C_2, \ldots, C_m\}$;
1: **begin**
2: $C \leftarrow \emptyset$;
3: **for** $i = 1$ **to** n **do**
4: **begin**
5: **for** $j = 1$ **to** $|\boldsymbol{C}|$ **do** //$|\boldsymbol{C}|$ denotes the number of communities in $\boldsymbol{C}$
6: **begin**
7: **if** (C_j and v_{π_i} satisfy (17)) **then** $C_j \leftarrow C_j \cup v_{\pi_i}$ and update $T_{\text{signed}}(C_j)$;
8: **if** (detect separated communities) **then** break;
9: **end**;
10: **if** (v_{π_i} has not been added to any community) **then** $\boldsymbol{C} \leftarrow \boldsymbol{C} \cup \{v_{\pi_i}\}$;
11: **end**;
12: **if** (detect overlapping communities) **then**
13: **while** (there are two communities in $\boldsymbol{C}$ satisfying (18)) **do**
14: Merge these two communities;
15: **end**.

B. Evolutionary Operators

For individuals represented by the direct and indirect combined coding, evolutionary operators can be conducted on both $\boldsymbol{A}\langle \boldsymbol{P} \rangle$ and $\boldsymbol{A}\langle \boldsymbol{C} \rangle$. For $\boldsymbol{A}\langle \boldsymbol{P} \rangle$, the partially matched crossover (PMX) proposed in [42] is employed. This operator was designed for the representation of array of individuals to solve traveling salesman problems. In addition, a mutation operator, which randomly selects two elements in the permutation to swap, is also used.

For $\boldsymbol{A}\langle \boldsymbol{C} \rangle$, the one-way crossover operator introduced in [21] is employed, and a new tightness based mutation operator is designed as follows. Let $\boldsymbol{A}\langle \boldsymbol{C} \rangle = (c_1, c_2, \ldots, c_n)$, then for each c_i, $1 \leq i \leq n$, if $U(0, 1)$, which is a uniformly distributed random number in the range of [0, 1], is larger than T_{signed} of community C_{c_i}, then select a node v_j from the neighbors with positive similarities of v_i based on the roulette wheel selection according to their similarity, and then assign c_j to c_i. If no neighbor has positive similarity, then v_j will be the neighbor with the largest similarity.

C. Implementation of MEA$_s$-SN

In the past few years, many studies have been devoted to apply EAs to MOPs [43]–[46]. Among existing EAs for MOPs, MOEA/D [46] showed an excellent performance. Thus, MEA$_s$-SN is implemented under the framework of MOEA/D, with the designed representation and evolutionary operators introduced in Section III-B. Moreover, an improvement is

also made by making use of the advantages of the designed representation, which is explained in details as follows.

To make use of the advantages of both $A\langle P\rangle$ and $A\langle C\rangle$, in MEA$_s$-SN, the population is first initialized to $A\langle P\rangle$; that is, *Popsize* permutations are randomly generated. Then, for detecting separated communities, each permutation is transformed to a set of communities by the decoder. This set of communities is further transformed to $A\langle C\rangle$, which can be realized easily in linear time. In the following evolutionary process, each individual is represented as $A\langle C\rangle$, and the corresponding operators are conducted. In this way, the initial $A\langle P\rangle$ population can generate a better population through the decoder, and the following operations on $A\langle C\rangle$ can be realized time efficiently. In this way, the algorithm can benefit from the advantages of both $A\langle P\rangle$ and $A\langle C\rangle$. For detecting overlapping communities, $A\langle P\rangle$ is used during the whole evolutionary process since $A\langle C\rangle$ can not handle overlapping communities.

Since MOEA/D decomposes an MOP into a number of scalar optimization subproblems and optimizes them simultaneously, the decomposition approach is important. In MEA$_s$-SN, the Tchebycheff approach is employed. Thus, based on the two objective functions designed in Section II, the scalar optimization problem is in the following form:

$$\begin{aligned}\text{minimize } g^{te}&\left(\boldsymbol{C}|\left(\lambda_{\text{pos-in}},\lambda_{\text{neg-out}}\right),\left(f^*_{\text{pos-in}},f^*_{\text{neg-out}}\right)\right)\\&=\max\left\{\lambda_{\text{pos-in}}\cdot\left|f_{\text{pos-in}}(\boldsymbol{C})-f^*_{\text{pos-in}}\right|,\lambda_{\text{neg-out}}\cdot\right.\\&\quad\left.\left|f_{\text{neg-out}}(\boldsymbol{C})-f^*_{\text{neg-out}}\right|\right\}\end{aligned}\tag{19}$$

where g^{te} denotes the obtained scalar objective function, $\boldsymbol{C}$ is a possible set of communities, and $(\lambda_{\text{pos-in}}, \lambda_{\text{neg-out}})$ is a weight vector which satisfies $\lambda_{\text{pos-in}}\geq 0$, $\lambda_{\text{neg-out}}\geq 0$, and $\lambda_{\text{pos-in}}+\lambda_{\text{neg-out}}=1$. $\left(f^*_{\text{pos-in}}, f^*_{\text{neg-out}}\right)$ is the reference point, and each element is the maximum value of the corresponding objective function. For more details of MOEA/D, please refer to [46]. The details of MEA$_s$-SN are given in Algorithm 2.

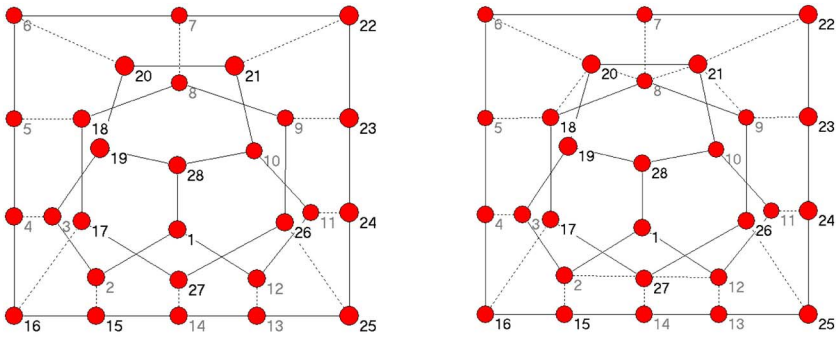
Clearly, Algorithm 2 shows that MEA$_s$-SN has two parts: the preprocessing part (Lines 2–7) and the evolutionary part (Lines 8–25). Thus, the time complexity of MEA$_s$-SN is determined by these two parts. Suppose there are m edges and n nodes in a network. The operation in Line 2 is related to the number of edges, so its time complexity is $O(m)$. The operation in Line 3 is related to the number of individuals, so its time complexity is O (*Popsize*). The operation in Line 4 is related to the number of individuals and the number of nodes, so its time complexity is $O(max\ \{Popsize, n\})$. The time complexity of Line 6 is determined by the decoder. To decode one permutation to a community structure, the time complexity is bounded by $O(\text{n}^2)$. Thus, the time complexity of Line 6 is $O(Popsize \times n^2)$. The operation in Line 7 can be finished in constant time. Therefore, the time complexity of the preprocessing part is $O(Popsize \times n^2)$.

As for the evolutionary part, the time complexity of crossover and mutation operators is related to the number of nodes, and can be realized in linear time, namely, $O(n)$. Therefore, the time complexity of the evolutionary part is bounded by $O(Gen \times Popsize \times n)$. As can be seen, the general time complexity of MEA$_s$-SN is determined by three factors, namely *Gen*, *Popsize*, and n. Since *Gen* and *Popsize* are normal parameters in EAs, in the experiments, we will further analyze

Algorithm 2 MEA$_s$-SN

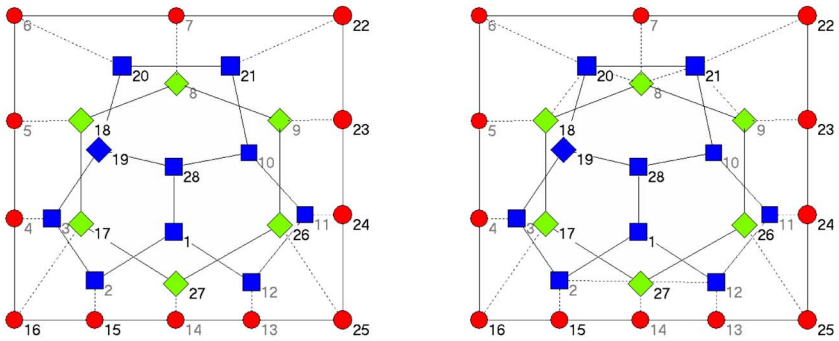
Input: $G=(V, E, w)$;
Popsize: the size of population;
A uniform spread of *Popsize* weight vectors: $\left(\lambda^1_{\text{pos-in}},\lambda^1_{\text{neg-out}}\right)$, $\left(\lambda^2_{\text{pos-in}},\lambda^2_{\text{neg-out}}\right)$, ..., $\left(\lambda^{Popsize}_{\text{pos-in}},\lambda^{Popsize}_{\text{neg-out}}\right)$;
T: the number of weight vectors in the neighborhood of each weight vector;
Gen: the number of generations;
Output: The final population.

1: **begin**
2: Calculate the signed similarity of any two connected nodes;
3: Calculate the Euclidean distances between any two weight vectors and then calculate the T closest weight vectors to each weight vector. For each $i=1, 2, \ldots,$ *Popsize*, set $B(i)=\{i_1,i_2, 2, \ldots, i_T\}$, where $\left(\lambda^{i_1}_{\text{pos-in}},\lambda^{i_1}_{\text{neg-out}}\right)$, $\left(\lambda^{i_2}_{\text{pos-in}},\lambda^{i_2}_{\text{neg-out}}\right)$, ... , $\left(\lambda^{i_T}_{\text{pos-in}},\lambda^{i_T}_{\text{neg-out}}\right)$ are the T closest weight vectors to $\left(\lambda^{i}_{\text{pos-in}},\lambda^{i}_{\text{neg-out}}\right)$;
4: Initialize the population in the form $A\langle P\rangle$: randomly generate *Popsize* permutations of n nodes, labeled as $A\left\langle \boldsymbol{P}^1_1\right\rangle$, $A\left\langle \boldsymbol{P}^1_2\right\rangle$, ..., $A\left\langle \boldsymbol{P}^1_{Popsize}\right\rangle$, where the superscript denotes the current generation;
5: **If** (Detecting separated communities)
6: Decode $A\left\langle \boldsymbol{P}^1_1\right\rangle, A\left\langle \boldsymbol{P}^1_2\right\rangle$, ..., $A\left\langle \boldsymbol{P}^1_{Popsize}\right\rangle$ to $A\left\langle \boldsymbol{C}^1_1\right\rangle, A\left\langle \boldsymbol{C}^1_2\right\rangle$, ..., $A\left\langle \boldsymbol{C}^1_{Popsize}\right\rangle$;
7: Initialize the reference idea point $\left(f^*_{\text{pos-in}}, f^*_{\text{neg-out}}\right)$;
8: **for** $i=1$ **to** *Gen* **do**
9: **begin**
10: **for** $j=1$ **to** *Popsize* **do**
11: **begin**
12: **if** (Detecting separate communities)
13: **begin**
14: Randomly select another individual, and conduct the one-way crossover operator on this individual and $A\left\langle \boldsymbol{C}^i_j\right\rangle$;
15: Conduct the tightness based mutation operator on $A\left\langle \boldsymbol{C}^i_j\right\rangle$;
16: **end**;
17: **if** (Detecting overlapping communities)
18: **begin**
19: Randomly select another individual, and conduct the PMX crossover operator on this individual and $A\left\langle \boldsymbol{P}^i_j\right\rangle$;
20: Conduct the swap mutation operator on $A\left\langle \boldsymbol{P}^i_j\right\rangle$;
21: **end**;
22: Update the reference idea point $\left(f^*_{\text{pos-in}}, f^*_{\text{neg-out}}\right)$;
23: Update the neighborhood solutions based on $B(i)=\{i_1, i_2, \ldots, i_T\}$;
24: **end**;
25: **end**;
26: **end**.



(a) (b)

Fig. 4. Original topological structure of the two illustrative SNs from [31]. Solid edges denote positive links and dashed edges denote negative links. (a) Illustrative SN 1. (b) Illustrative SN 2.



(a) (b)

Fig. 5. Partitions of the two illustrative SNs obtained by MEA$_s$-SN. Nodes with the same color and shape belong to the same community. (a) Illustrative SN 1. (b) Illustrative SN 2.

how the time complexity of MEA$_s$-SN increases with the number of nodes.

IV. Experiments

In this section, the performance of MEA$_s$-SN is validated on both benchmark and synthetic networks, and compared with those of the three existing algorithms, namely, FEC [31], Louvain method [47], [48], and CSA$_{HC}$-SN [49]. In the following experiments, *Popsize* and *Gen* of MEA$_s$-SN are set to 100. The original literature showed that MOEA/D performs well if the neighborhood size T is greater than 3, thus T is set to 20.[1] Next, the measures used to evaluate the performance of different algorithms and a synthetic network generator are first introduced. Then, three groups of experiments are conducted, which are respectively on: 1) benchmark networks; 2) separated; and 3) overlapping CD from synthetic SNs.

A. Evaluation Measures and Synthetic Network Generator

In general, each nondominated solution in the final generation is valuable for MOPs, which will be illustrated in Section IV-C. However, for the CD problem here, we need to select one suitable individual. Thus, Q_{signed} [16], which is extended from the popular measure Q [10] for the signed case, is employed for separated communities

$$Q_{\text{signed}} = \frac{1}{2w^+ + 2w^-} \sum_i \sum_j \left[w_{ij} - \left(\frac{w_i^+ w_j^+}{2w^+} - \frac{w_i^- w_j^-}{2w^-} \right) \right] \delta\left(C_i, C_j\right) \tag{20}$$

where w_{ij} is the weight of the adjacency matrix, $w_i^+(w_j^+)$ denotes the sum of all positive weights of node $v_i(v_j)$, and $w_i^-(w_j^-)$ denotes the sum of all negative weights of node $v_i(v_j)$. $w^+(w^-)$ represents the total positive(negative) strength of the SN, and $C_i(C_j)$ represents the community which node $v_i(v_j)$ belongs to, and $\delta(C_i, C_j)$ is 1 if nodes v_i and v_j are in same community; otherwise 0.

For the overlapping case, we propose a new function Q_{os} to evaluate the final results. Shen *et al.* [17] extended the popular Q [10] to the overlapping situation

$$Q_{\text{ov}} = \frac{1}{2w} \sum_i \sum_j \frac{1}{O_i O_j} \left(w_{ij} - \frac{w_i w_j}{2w} \right) \delta\left(C_i, C_j\right) \tag{21}$$

where $w_i = \sum_j w_{ij}$ denotes the strength of node v_j and $2w = \sum_i w_i = \sum \sum_{ij} w_{ij}$ stands for the total strength of the network. O_i is the number of communities which vertex v_i belongs to. In order to evaluate signed overlapping community

[1]The source codes of MEA$_s$-SN can be downloaded at http://see.xidian.edu.cn/faculty/liujing/.

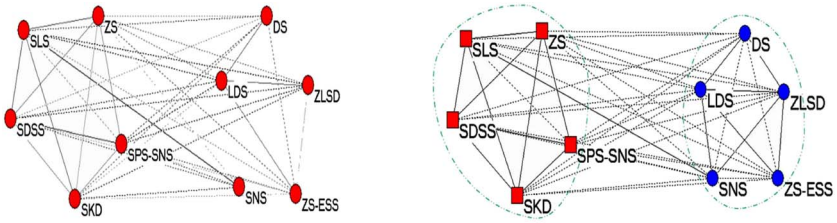


Fig. 6. (a) Topological structure of the Slovene parliamentary party network. (b) Community structure obtained by MEA$_s$-SN.

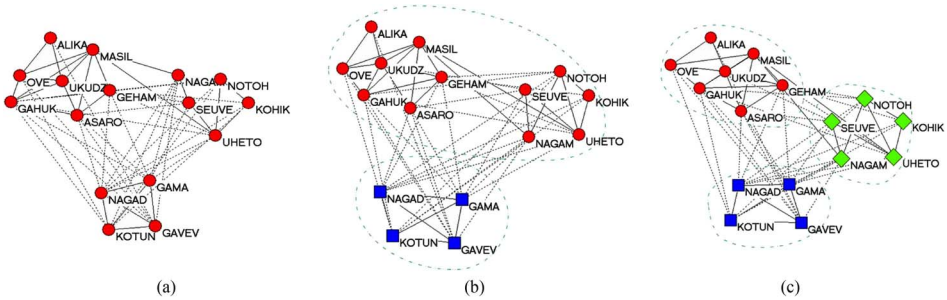


Fig. 7. (a) Topological structure of the Gahuku-Gama subtribes network. (b) and (c) Two community structures obtained by MEA$_s$-SN.

structures, combining the above Q_{signed} and Q_{ov}, we propose Q_{os} as follows:

$$Q_{\text{os}} = \frac{1}{2w^+ + 2w^-} \sum_i \sum_j \frac{1}{O_i O_j} \left[w_{ij} - \left(\frac{w_i^+ w_j^+}{2w^+} - \frac{w_i^- w_j^-}{2w^-} \right) \right] \delta\left(C_i, C_j\right). \tag{22}$$

Therefore, in the final generation, the individual with maximum Q_{signed} or Q_{os} is selected as the final community structure found. Further, to measure the partitioning quality, the normalized mutual information (*NMI*) [18] is adopted to estimate the similarity between true partitions and detected ones for separated cases, and the generalize NMI (GNMI) [15] is used for the overlapping case.

Real-world networks often have some limitations on their sizes or community structures. To systematically test the performance of the new algorithm, a synthetic network generator is needed. However, very few studies are available for generating SNs. Yang *et al*. [31] designed a generator for SNs, but this generator has two defects: 1) each node has the same degree and 2) cannot generate overlapping networks. One popular generator to generate networks with community structures is the Lancichinetti–Fortunato–Radicchi (*LFR*) benchmark [19], [20], which is suitable for both separated and overlapping situations.

Therefore, combining the LFR benchmark with the one proposed in [31], we design a new SNs generator, which is labeled as *SRN*(n, k, *maxk*, t_1, t_2, *minc*, *maxc*, *on*, *om*, μ, P_-, P_+). Here, n is the number of nodes, and k and *maxk* are the average and maximum degree of each node. t_1 and t_2 are the minus exponents for the degree and community size distributions, both of which are power laws. *minc* and *maxc* are the minimum and maximum community size. *on* and *om* are respectively the number of overlapping nodes and the number of memberships of overlapping nodes.

The last three parameters are important. μ is the fraction of links that each node shares with nodes in other communities, which controls the cohesiveness of the communities inside the generated SNs. Thus, the higher the value of μ is, the more ambiguous the community structure is. P_- is the fraction of negative links within communities, while P_+ is the fraction of positive links between communities. Ideally, negative links should be between communities and positive links should be within communities. Thus, P_- and P_+ are two parameters used to adjust the noise level. Being the same with μ, the larger the values of P_- and P_+ are, the more ambiguous the community structure is. As can be seen, this benchmark poses severe and flexible tests to algorithms. Given a fixed μ, we can control the noise level by adjusting P_- and P_+. In general, it will be more difficult to extract communities correctly when the values of P_- and P_+ are large. In the following experiments, the capability of MEA$_s$-SN in handling different μ, P_-, and P_+ are systematically tested.

B. Experiments on Benchmark Networks

In this subsection, four benchmark SNs widely used, including two illustrative SNs and two real social SNs, are employed to validate the performance of MEA$_s$-SN. Moreover, although MEA$_s$-SN is designed for SNs, it can also be used to unsigned networks. Therefore, three popular unsigned benchmark networks, namely, the Zachary karate club network, the bottlenose dolphin network, and the American football network, are tested.

The two illustrative SNs came from [31] and each has 28 nodes. Their topological structures are shown in Fig. 4. The community structures obtained by MEA$_s$-SN are given in Fig. 5. As can be seen, they can be divided into three communities. The links within communities are positive and those between communities are negative. MEA$_s$-SN found the correct partitions successfully for both networks, and $NMI = 1$.

The first real social network is the Slovene Parliamentary Party Network, which is the relation network of ten parties of the Slovene Parliamentary in 1994 [50]. Positive links mean the two parties' Parliament activities are similar, while nega-

tive links mean their activities are dissimilar. Fig. 6(a) shows the original topological structure, and the community structure obtained by MEA$_s$-SN is shown in Fig. 6(b). As can be seen, all parties are separated into two opponent communities. This result is the same as that given by Kropivnik and Mrvar [50].

The second real social network is the Gahuku-Gama Subtribes network, which was generated by Read on the cultures of highland New Guinea proposed in [51]. It regards the political alliances and oppositions among 16 Gahuku-Gama subtribes, which were distributed in a particular area and were involved in warfare with each other in 1954. Positive and negative links represent the political arrangements with positive and negative ties, respectively. Fig. 7(a) shows the original topological structure, and the community structures obtained by MEA$_s$-SN are shown in Fig. 7(b) and (c). As can be seen, two meaningful partitions were found. Fig. 7(b) has a higher value of $f_{\text{pos}-\text{in}}$ while Fig. 7(c) has a higher value of $f_{\text{neg}-\text{out}}$, and these results are identical to that reported in [51] and [52].

To further evaluate the effectiveness of MEA$_s$-SN on unsigned networks, Fig. 8 presents the community structures found by MEA$_s$-SN for the Zachary karate club network, the bottlenose dolphin network, and the American football network. As can be seen, for the Zachary karate club network, the nodes are divided into two groups. For the bottlenose dolphin network, the nodes are divided into four groups. For the American football network, the nodes are divided into 11 groups. The obtained values of Q are 0.372, 0.521, and 0.600, respectively, and the community structures found are meaningful.

C. Experiments on Separated Community Detection from Synthetic SNs

In this subsection, synthetic SNs are used to test the performance of MEA$_s$-SN in detecting separated communities. Parameters *on* and *om* in the generator are not needed. We first give the comprehensive results of MEA$_s$-SN over SNs with 1000, 5000, and 10 000 nodes, and then make a comparison between MEA$_s$-SN and three existing algorithms. At last, an experiment is also conducted to illustrate the advantage of using the multiobjective framework.

1) *Experimental Results of MEA$_s$-SN:* In this experiment, the number of nodes is set to 1000, 5000, and 10 000, respectively. k and *maxk* are set to 20 and 50 for networks with 1000 nodes, 40 and 100 for networks with 5000 and 10 000 nodes. t_1, t_2, *minc*, *maxc* are set to 2, 1, 20, and 50, respectively. Then, the effect of μ, P_+, and P_- is systematically tested. μ increases from 0.1 to 0.5 in the step of 0.1, P_+ increases from 0 to 1 in the step of 0.2, and P_- increases from 0 to 0.8 in the step of 0.2. For each combination of these three parameters, 30 independent runs of MEA$_s$-SN were conducted, and the averaged *NMI* and Q_{signed} are reported in Figs. 9 and 10.

As can be seen, for different combinations of μ, P_+, and P_-, the performance of MEA$_s$-SN on networks with 1000, 5000, or 10 000 nodes is similar, and MEA$_s$-SN works well on these networks. For example, for networks with 10 000 nodes, even when μ increases from 0.1 to 0.5, the *NMI* is almost always higher than 0.8 when P_- is in the range of [0, 0.2]. Even at high level of noise when P_- is in the range of [0.2, 0.6], the *NMI* is still almost higher than 0.5. The obtained Q_{signed} decreases with μ naturally, since the larger the value of μ is, the more ambiguous the community structure is.

TABLE I
COMPUTATIONAL TIME OF MEA$_s$-SN

Number of nodes	1000	5000	10000
Time (second)	9.9	75.7	328.1

The results also indicate that when P_+ and P_- increase, the values of *NMI* and Q_{signed} decrease, and P_- has a larger impact on the performance than P_+. In fact, as P_- and P_+ increase, the noise level within and between communities increases. Thus, the community structure becomes more ambiguous, and the clustering accuracy decreases accordingly. Actually, when P_+ increases from 0 to 1, the positive links between communities become denser. In such cases, the community structure will be decided not only by the signs of links but also by the density of links. Since MEA$_s$-SN takes into account both the signs and density of links, it deals very well with positive networks. Therefore, MEA$_s$-SN is insensitive to the noise level of P_+. In fact, Yang *et al.* [31] stated that FEC also showed a similar performance, which is sensitive to P_-, but insensitive to P_+.

It is well known that EAs have a high computational cost than heuristic algorithms since they need to maintain a population during the evolutionary process. To show the computational cost of MEA$_s$-SN, we have recorded the actual computational time for analyzing the above networks. All the experiments were run on a computer with a 3.2 GHz CPU and 4GB memory. The operating system was Windows 7, and the simulation was implemented and tested using Microsoft Visual Studio 2009. The averaged actual computational time of MEA$_s$-SN is reported in Table I. As can be seen, although the time taken by MEA$_s$-SN increases quickly with the network size, it only used 9.9 s for networks with 1000 nodes. For networks with 10 000 nodes, it used 328.1 s, which is also acceptable.

2) *Comparison between MEA$_s$-SN and existing algorithms:* In this experiment, MEA$_s$-SN is compared with three existing algorithms, namely, FEC [31], Louvain method [47], [48], and CSA$_{\text{HC}}$-SN [49]. FEC is an algorithm recently proposed for SNs and obtained a good performance. Louvain method is a special version for SNs of the algorithm in [48], and is implemented in Pajek [47]. CSA$_{\text{HC}}$-SN is a memetic algorithm for SNs proposed in our previous work [49], which can optimize both Q_{signed} and the improved modularity density D. In the following experiment, FEC and CSA$_{\text{HC}}$-SN run under the same experimental environment with MEA$_s$-SN. Both Q_{signed} and D are optimized by CSA$_{\text{HC}}$-SN, labeled as CSA$_{\text{HC}}$-SN(Q) and CSA$_{\text{HC}}$-SN(D), respectively. Louvain method runs under Pajek.

All these four algorithms are tested on the synthetic SNs with μ=0.1$\sim$0.5 and $P_+ = 0\sim1$ systematically. Since previous results showed that MEA$_s$-SN and FEC are sensitive to P_-, and the performance drops when P_- is larger, we only compare the performances of different algorithms when P_- is in the range of [0, 0.4]. Since the computational cost of CSA$_{\text{HC}}$-SN is too high for large networks, the network size for CSA$_{\text{HC}}$-SN is set to 1000 and 10 000 for three other algorithms. For each network, ten independent runs are conducted for each algorithm and the results are shown in Fig. 11.

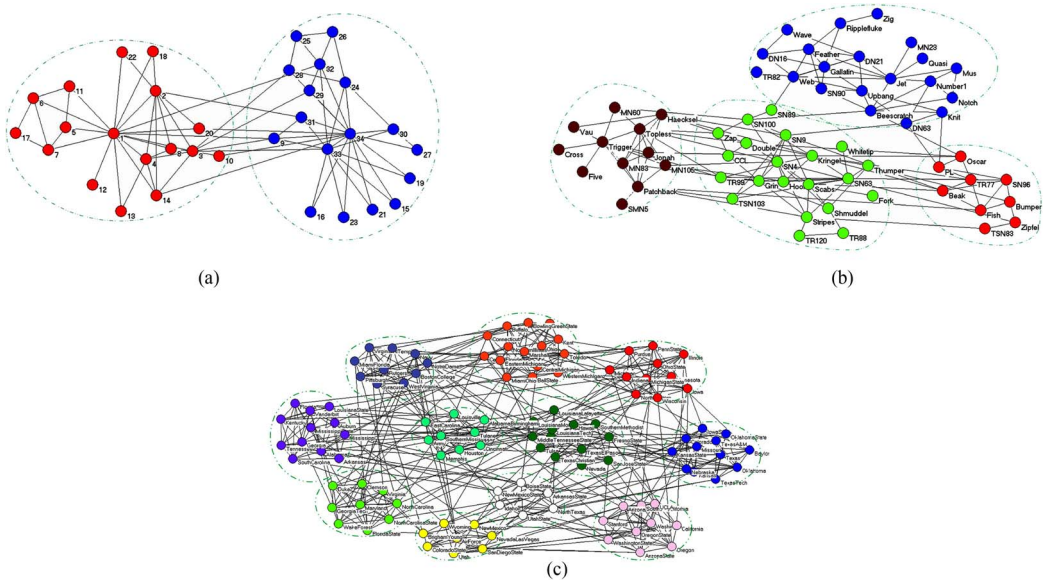


Fig. 8. Community structures found by MEA$_s$-SN for the three popular unsigned benchmark networks. (a) Zachary karate club network, $Q = 0.372$. (b) Bottlenose dolphin network, $Q = 0.521$. (c) American football network, $Q = 0.600$.

As can be seen, first, MEA$_s$-SN outperforms CSA$_{\text{HC}}$-SN(Q) and CSA$_{\text{HC}}$-SN(D) obviously almost in all parameter combinations. Some results for $P_+ = 1.0$ of CSA$_{\text{HC}}$-SN(Q) and CSA$_{\text{HC}}$-SN(D) are missing due to the high computational cost.

For the Louvain method, when $P_- = 0$, the *NMI* is always higher than 0.8 when μ increases from 0.1 to 0.5 and P_+ increases from 0 to 1. The *NMI* of MEA$_s$-SN is also always higher than 0.8 in these parameter combinations, and better than that of Louvain method in most cases. When P_- increases, the performance of Louvain method drops dramatically, which is also very sensitive to P_+; that is, when P_+ increases, the performance drops dramatically, too. Although the performance of MEA$_s$-SN also drops, it is clearly better than that of Louvain method, which illustrates that MEA$_s$-SN is more robust to noises.

For FEC, when $P_- = 0$, the *NMI* is always higher than 0.8 except when μ=0.5, and MEA$_s$-SN performs similar or better than FEC for these parameter combinations. When $P_- \geq 0.2$, the performance of FEC also drops, but is not so sensitive to the increase of P_+ like Louvain method. To compare with our method, FEC is outperformed by MEA$_s$-SN for all parameter combinations when $P_- = 0.2$. When $P_- = 0.4$, MEA$_s$-SN always outperforms FEC when $\mu \geq 0.3$.

3) *Advantage of the Multiobjective Framework:* Our algorithm is based on a multiobjective framework, and generally speaking, each nondominated solution in the final generation is valuable for MOPs. Therefore, in this experiment, we use a hierarchical network to show the different properties of the obtained solutions, which demonstrates the advantage of the multiobjective framework. The hierarchical network used is a revised version of the H13-4 network [53], [54], which is also used to demonstrate the advantage of multiobject techniques in CD in [22]. The original H13-4 has 256 nodes, and all nodes can be divided into four communities (each with 64 nodes). Further, each community can be divided into four small communities (each with 16 nodes).

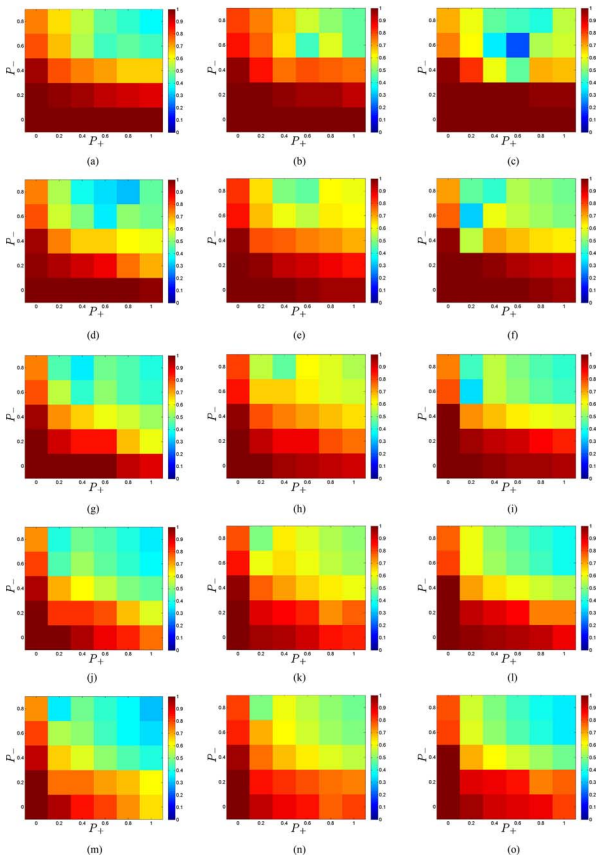


Fig. 9. Obtained *NMI* of MEA$_s$-SN for synthetic SNs. (a) μ=0.1, 1000 nodes. (b) μ=0.1, 5000 nodes. (c) μ=0.1, 10 000 nodes. (d) μ=0.2, 1000 nodes. (e) μ=0.2, 5000 nodes. (f) μ=0.2, 10 000 nodes. (g) μ=0.3, 1000 nodes. (h) μ=0.3, 5000 nodes. (i) μ=0.3, 10 000 nodes. (j) μ=0.4, 1000 nodes. (k) μ=0.4, 5000 nodes. (l) μ=0.4, 10 000 nodes. (m) μ=0.5, 1000 nodes. (n) μ=0.5, 5000 nodes. (o) μ=0.5, 10 000 nodes.

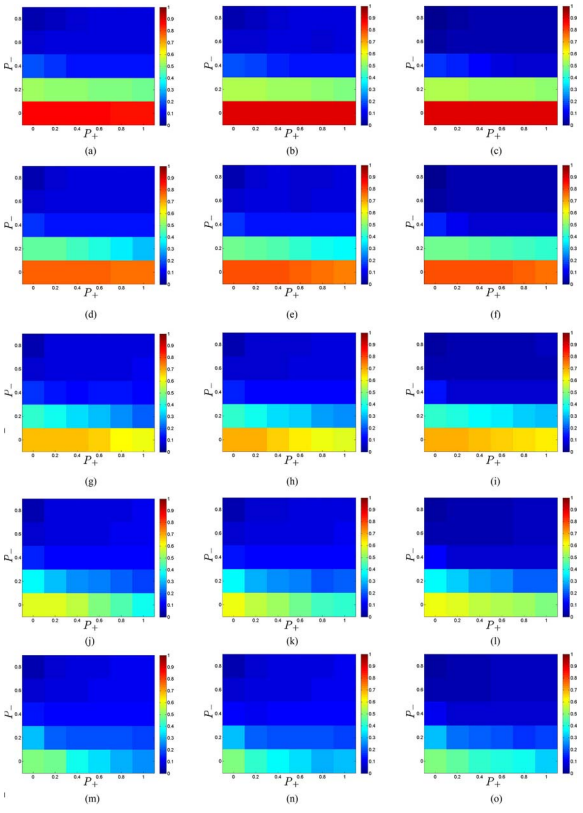


Fig. 10. Obtained Q_{signed} of MEA$_s$-SN for synthetic SNs. (a) μ=0.1, 1000 nodes. (b) μ=0.1, 5000 nodes. (c) μ=0.1, 10 000 nodes. (d) μ=0.2, 1000 nodes. (e) μ=0.2, 5000 nodes. (f) μ=0.2, 10 000 nodes. (g) μ=0.3, 1000 nodes. (h) μ=0.3, 5000 nodes. (i) μ=0.3, 10 000 nodes. (j) μ=0.4, 1000 nodes. (k) μ=0.4, 5000 nodes. (l) μ=0.4, 10 000 nodes. (m) μ=0.5, 1000 nodes. (n) μ=0.5, 5000 nodes. (o) μ=0.5, 10 000 nodes.

We change H13-4 to an SN by flipping all links between large communities to negative and the links between small communities to negative with probability 0.4. All final solutions obtained by MEA$_s$-SN are illustrated in Fig. 12(a), and the community structures corresponding to solutions P_a and P_b are shown in Fig. 12(b) and (c), respectively. As can be seen, P_a divides the network into four large communities while P_b divides the network into 16 small communities. It is clear that both partitions are meaningful. Thus, MEA$_s$-SN can provide different partitions in one run, which gives more choices for decision makers.

D. Experiments on Overlapping Community Detection From Synthetic SNs

In this experiment, synthetic SNs are used to test the performance of MEA$_s$-SN in detecting overlapping communities. Parameters *on* and *om* are set to 100 and 2, respectively, and the number of nodes is set to 1000. Other parameters of the generator are the same with those in Section IV-C. We also systematically test the effect of μ, P_+, and P_-. μ increases from 0.1 to 0.5, P_+ increases from 0 to 1, and P_- increases from 0 to 0.8. For each combination of these three parameters, 30 independent runs of MEA$_s$-SN are conducted, and the averaged *NMI* and Q_{signed} are reported in Fig. 13.

Compared with detecting separated communities, detecting overlapping communities is more difficult, which is demonstrated by the obtained Q_{os}. We can see that Q_{os} decreases

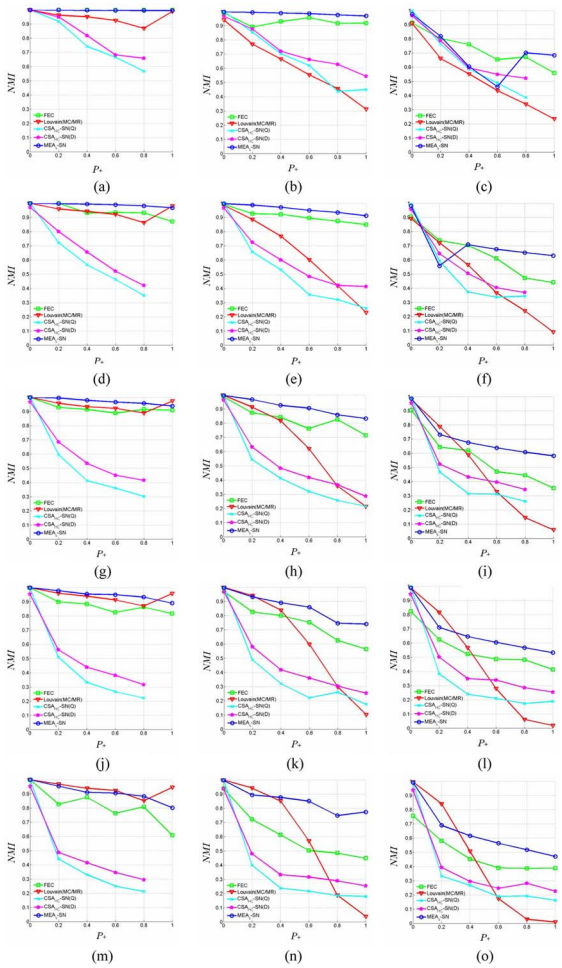


Fig. 11. Comparison between MEA$_s$-SN and existing algorithms. (a) μ=0.1, P_-=0.0. (b) μ=0.1, P_-=0.2. (c) μ=0.1, P_-=0.4. (d) μ=0.2, P_-=0.0. (e) μ=0.2, P_-=0.2. (f) μ=0.2, P_-=0.4. (g) μ=0.3, P_-=0.0. (h) μ=0.3, P_-=0.2. (i) μ=0.3, P_-=0.4. (j) μ=0.4, P_-=0.0. (k) μ=0.4, P_-=0.2. (l) μ=0.4, P_-=0.4. (m) μ=0.5, P_-=0.0. (n) μ=0.5, P_-=0.2. (o) μ=0.5, P_-=0.4.

dramatically when μ, P_+, and P_- increase; that is to say, the community structure is getting more and more ambiguous quickly. However, Fig. 13 shows that MEA$_s$-SN still obtains a good performance when μ, P_-, and P_+ are not too large. For example, when $\mu \leq 0.3$, $P_+ \leq 0.4$, and $P_- \leq 0.2$, the *NMI* is always larger than 0.8. In fact, even when $0.4 \leq P_+ \leq 0.8$, the *NMI* is still larger than 0.5. For $\mu \geq 0.4$, the community structure gets very ambiguous, and the *NMI* decreases quickly accordingly. However, when the noise levels are not too large, that is, when both P_+ and P_- are smaller than 0.2, the obtained *NMI* is still larger than 0.5.

V. Related Work

To understand and utilize the information in social networks, various methods to capture the network structure characteristics have been proposed from different perspectives. Next, we first introduce the related work on social network analysis, and then introduce the related work on community detection methods.

A. Related Work on Social Network Analysis

Due to the complexity of social networks, analyses from various aspects have been conducted, which can be roughly

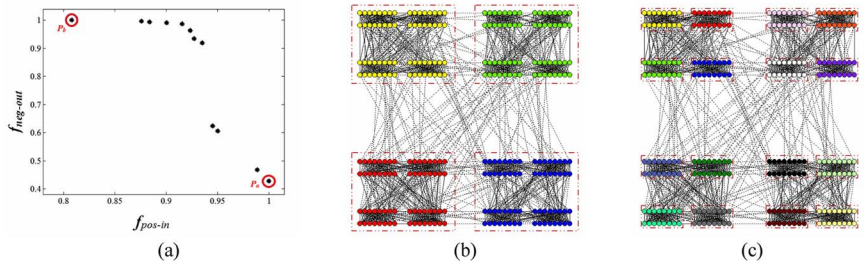


Fig. 12. Community structures obtained by MEA$_s$-SN for a hierarchical network. (a) Final population. (b) Community structure of the solution P_a in (a). (c) Community structure of the solution P_b in (a).

divided into two main categories, namely the micro level and the macro level. From the micro level, a great deal of work focused on the properties of individual nodes or links, such as triadic closure prediction [55], homophily analysis [55], [56], social ties inference [39], link prediction [30], trust evaluation, and propagation [29], [57], which are related to SNs to some extent.

Lou *et al*. [55] investigated how a reciprocal link was developed from parasocial relationship and how the relationship further developed into triadic closure, one of the fundamental processes of link formation. They also studied how homophily, the principle suggests that users with similar characteristics tend to associate with each other, is satisfied over networks with parasocial and reciprocal relationships.

Tang *et al*. [39] developed a framework for classifying the type of social relationships (both positive and negative relationships) by learning across multiple heterogeneous networks, while a bulk of similar research has focused on inferring particular types of relationships in a specific social network. The framework incorporates social theories into a factor graph model, which effectively improves the accuracy of inferring the types of social relationships in a target network by borrowing knowledge from a different source network.

Kunegis *et al*. [30] investigated the negative link feature of social networks that allows users to tag other users as foes or distrusted in addition to the usually friend and trusted links. To answer the question whether negative links have an added value for an online social network, they investigated the machine learning problem of predicting negative links of such a network using only positive links as a basis, with the idea that if this problem can be solved with high accuracy, when the negative links feature is redundant. The experimental results showed that the negative link feature has a small but measurable added value of these social networks, while illustrated the importance of negative links.

The trust evaluation and propagation in a distributed environment is an important topic related to social-network-based interrelationship analysis. Kamvar *et al*. [57] studied a peer-to-peer file-sharing network, in which a peer assigns a trust value to those peers who have provided it with authentic files. They proposed an EigenTrust algorithm that considers the entire history of uploading with individual peers by aggregating the normalized local trust values of all users. Based on a general framework of trust propagation scheme, Guha *et al*. [29] addressed the problem of predicting the trust between any two people in a social network connected by ratings or trust/distrust scores. The scheme is formulated based on a weighted chaining of four well-defined atomic propagations to a set of belief that users hold about each other. Based on this chaining, a final matrix that contains the trust or distrust of any two people can be derived after a number of propagations.

From the macro level, community detection, which focuses on the relationship between groups of nodes, has attracted increasing attention. CD from SNs, which is the focus of this paper, is also known as correlation clustering [16], [58], [59] in the field of social network analysis. Positive and negative links correspond to agreements and disagreements between nodes, respectively.

Bansal and Chawla [58] introduced correlation clustering motivated by document clustering and agnostic learning and showed that it is an NP-hard problem to make a partition to a complete signed graph. They provided a constant factor approximation for minimizing the number of disagreements and a polynomial-time approximation scheme for maximizing the total number of agreements that can achieve any constant error gap in polynomial time. Their algorithm can be extended to the case of minimizing weighted disagreements or maximizing weighted agreements.

Demaine and Immorlica [59] studied the problem of correlation clustering with partial information, and gave an approximation algorithm based on a linear-programming rounding and the region-growing technique and showed that the problem is APX-hard, i.e., any approximation would require improving the best approximation algorithms known for minimum multicut.

Gómez *et al*. [16] presented a reformulation of modularity that allows the analysis of the community structure in networks of correlated data. The new modularity preserves the probabilistic semantics of the original definition even when the network is directed, weighted, signed, and has self-loops. This is the most general condition one can find in the study of any network, in particular those defined from correlated data.

B. Related Work on Community Detection Methods

During the past decade, the research on analyzing the community structure in complex networks has drawn a great deal of attention, and various kinds of algorithms have been proposed. We first reviewed some classic and recent proposed algorithms for unsigned networks, and then those for SNs.

Girvan and Newman [6] proposed the Girvan–Newman (GN) algorithm which is one of the most known algorithms proposed so far. Newman [10] proposed the well known measure modularity Q to evaluate the quality of obtained communities, which is widely used. There have been other studies on community identification from complex networks

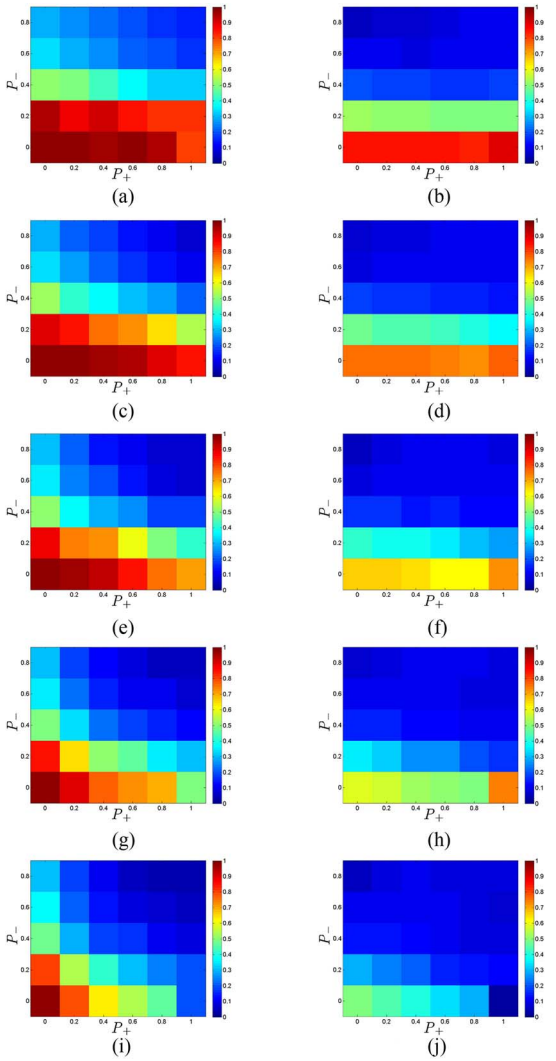


Fig. 13. Results of MEA_s-SN for detecting overlapping communities. (a) μ=0.1, *GNMI*. (b) μ=0.1, Q_{os}. (c) μ=0.2, *GNMI*. (d) μ=0.2, Q_{os}. (e) μ=0.3, *GNMI*. (f) μ=0.3, Q_{os}. (g) μ=0.4, *GNMI*. (h) μ=0.4, Q_{os}. (i) μ=0.5, *GNMI*. (j) μ=0.5, Q_{os}.

that utilize physics-based method. For example, Palla *et al*. [23] introduced the concept of clique percolation to the problem of identifying overlapping communities, and one node can belong to more than one community.

Rosvall and Bergstrom [24] introduced an information theoretic approach that reveals community structure in weighted and directed networks. The probability flow of random walks on a network as a proxy for information flows in the real system and decompose the network into modules by compressing a description of the probability flow. The result is a map that both simplifies and highlights the regularities in the structure and their relationship.

Lancichinetti *et al*. [25] presented the order statistics local optimization method, the first method capable to detect clusters in network accounting for edge directions, edge weights, overlapping communities, hierarchies, and community dynamics. It is based on the local optimization of a fitness function expressing the statistical significance of clusters with respect to random fluctuations, which is estimated with tools of Extreme and Order Statistics. OSLOM can be used alone or as a refinement procedure of partitions/covers delivered by other techniques.

Meo *et al*. [26] proposed a strategy to enhance existing CD algorithms by adding a preprocessing step in which edges are weighted according to their centrality, with respect to the network topology. In this approach, the centrality of an edge reflects its contribute to make arbitrary graph transversals, i.e., spreading messages over the network, as short as possible. The strategy is able to effectively complement information about network topology and it can be used as an additional tool to enhance community detection. The computation of edge centralities is carried out by performing multiple random walks of bounded length on the network.

Given the increasing popularity of algorithms for overlapping communities, quantitative measures are needed to measure the accuracy of a method. McDaid *et al*. [27] and Lancichinetti *et al*. [15] extended the popular measure normalized mutual information [18] to evaluate overlapping communities. Lázár *et al*. [28] also introduce a nonfuzzy measure which has been designed to rank the partitions for a network's nodes into overlapping communities. Shen *et al*. [17] extended the popular measure Q [10] to the overlapping situation.

Multiobjective evolutionary algorithms have also been used to solve CD. Pizzuti [11] proposed a multiobjective genetic algorithm to uncover community structure, which optimized two objective functions respectively for maximizing connections within the same community and minimizing connections between different communities. Shi *et al*. [22] formulated a multiobjective framework for CD and proposed a MOEA for finding efficient solutions under the framework. Shi *et al*. [38] also analyzed the correlations of 11 objective functions that have been used or can potentially be used for CD, and the results showed that MOEAs optimizing over a pair of negatively correlated objectives usually performed better than the single-objective algorithm optimizing over either of the original objectives. Liu *et al*. [13] proposed a MOEA to detect separated and overlapping communities simultaneously.

Although the above algorithms and measures for CD obtained a good performance in the type of networks they were target to, they all designed for unsigned networks. The studies for SNs are much less. Yang *et al*. [31] proposed the algorithm FEC for mining SNs, and both positive within-group relations and negative between-group relations are dense. FEC adopts an agent-based heuristic, and can be used to detect communities from both signed and unsigned networks.

Based on the structural balance theories [60], [61] and the blockmodel, Doreian and Mrvar [52] presented an approach to partition an SN by using the local search method, which divided a graph into several parts so as to minimize a predefined error function. One limitation of this algorithm is that it needs to know the number of groups beforehand, and also sensitive to the initial partition. In addition, this algorithm takes into account only the signs of links for partitioning signed networks, neglecting the density of links, which, in many cases, is a salient feature in partitioning. In [31], the experimental results showed that the performance of FEC is much better than that of this algorithm.

Doreian [62] stated that the generalized blockmodeling faces a pair of vulnerabilities. One is sensitivity to poor quality of the relational data and the other is a risk of over fitting blockmodels to the details of specific networks. Thus,

Doreian [62] presented a method of tackling these problems by viewing (when possible) observed social relations as multiple indicators of an underlying affect dimension. Quadratic assignment methods using matching coefficients, product moment correlations and Goodman and Kruskal's gamma are used to assess the appropriateness of using the sum of observed relations as input for applying generalized blockmodeling.

Further, Doreian and Mrvar [63] showed that even those methods based on blockmodel have been useful, the blockmodel structure discerned may not be appropriate for all SNs. The provided some illustrative examples and then broaden the types of blockmodel that can be specified and identified for SNs within the generalized blockmodeling framework.

Wu *et al*. [64] investigated the impacts of negative links and examined the patterns in the spectral space of the graph's adjacency matrix. Their results showed that communities in a *k*-balanced SN are greatly different in the spectral space of its signed adjacency matrix despite connections among communities are dense.

Traag and Brugggeman [65] adapted the concept of modularity to detect communities in networks, in which both positive and negative links are present, and also evaluated the social network of international disputes and alliances.

All the above algorithms for CD from SNs are based on heuristic methods, but not on EAs. Thus, in our previous work [49], two EAs and two memetic algorithms were proposed to detect communities from SNs, and a comprehensive comparison was made to compare the performance of these four algorithms. The results show that the memetic algorithms outperform EAs in solving this problem. But all these four EAs were based on a single-objective framework, but not the multiobjective framework.

VI. Conclusion

In this paper, we propose a novel multiobjective algorithm, called MEA$_s$-SN, based on a new similarity to detect both separated and overlapping communities from signed social networks. In MEA$_s$-SN, positive similarities within communities and negative similarities between communities are modeled as two contradictory objective functions. Especially, a direct and indirect combined representation is designed so that MEA$_s$-SN can switch between different representations during the evolutionary process and benefit from both representations.

MEA$_s$-SN requires no prior knowledge on the community structure, such as the number of communities and a good initial partition. Thus, it is easy to be applied to different networks. Rigorous experiments on both benchmark networks and large-scale synthetic networks with 1000, 5000, and 10 000 nodes show the effectiveness and efficacy of MEA$_s$-SN, even in noisy situations. The thorough comparison between MEA$_s$-SN and three existing algorithms also demonstrate the better performance of MEA$_s$-SN over other algorithms.

MEA$_s$-SN has three parameters, namely the population size, the number generation allowed, and the neighborhood size. The former two are normal parameters in EAs, while the last one is a parameter in MOEA/D. The values of these parameters can be easily set, and no other parameters need to be tuned in MEA$_s$-SN. MEA$_s$-SN also requires no prior knowledge on hidden community structures. However, MEA$_s$-SN currently can only handle undirected networks, and in our future work, we will consider handling direct networks.

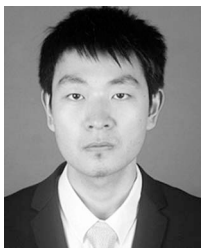
Acknowledgment

The authors would like to thank the reviewers for their helpful comments and valuable suggestions.

References

[1] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, pp. 268–276, Mar. 2001.
[2] R. Albert and A. L. Barabási, "Statistical mechanics of complex networks," *Rev. Mod. Phys*., vol. 74, pp. 47–97, Jan. 2002.
[3] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, no. 6638, pp. 440–442, 1998.
[4] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.
[5] L. A. Adamic and B. A. Huberman, "Power-law distribution of the world wide web," *Science*, vol. 287, no. 5461, p. 2115, 2000.
[6] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," in *Proc. Nat. Acad. Sci.*, vol. 99. 2002, pp. 8271–8276.
[7] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, no. 2, p. 026113, 2004.
[8] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Phys. Rev. E*, vol. 70, no. 6, p. 066111, 2004.
[9] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech.*, vol. 2008, no. 10, p. P10008, Oct. 2008.
[10] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E*., vol. 69, no. 6, p. 066133, 2004.
[11] C. Pizzuti, "A multiobjective genetic algorithm to find communities in complex networks," *IEEE Trans. Evol. Comput.*, vol. 16, no. 3, pp. 418–430, Jun. 2012.
[12] J. Liu, H. A. Abbass, W. Zhong, and D. G. Green, "Local-global interaction and the emergence of scale-free networks with community structures," *Artif. Life (MIT)*, vol. 17, no. 4, pp. 263–279, 2011.
[13] J. Liu, W. Zhong, H. A. Abbass, and D. G. Green, "Separated and overlapping community detection in complex networks using multiobjective evolutionary algorithms," in *Proc. CEC*, Jul. 2010, pp. 1–7.
[14] J. Huang, H. Sun, Y. Liu, Q. Song, and T. Weninger, "Towards online multiresolution community detection in large-scale networks," *PLOS ONE*, vol. 6. no. 8, p. e23829, 2011.
[15] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure of complex networks," *New J. Phys.*, vol. 11, p. 033015, Mar. 2009.
[16] S. Gómez, P. Jensen, and A. Arenas, "Analysis of community structure in networks of correlated data," *Phys. Rev. E*., vol. 80, no. 1, p. 016114, 2009.
[17] H. Shen, X. Cheng, K. Cai, and M. B. Hu, "Detect overlapping and hierarchical community structure in networks," *Physica A*, vol. 388, no. 8, pp. 1706–1712, 2009.
[18] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *J. Stat. Mech.*, vol. 2005, p. P09008, Sep. 2005.
[19] A. Lancichinetti and S. Fortunato, "Community detection algorithms: A comparative analysis," *Phys. Rev. E*, vol. 80, no. 5, p. 056117, 2008.
[20] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys. Rev. E*., vol. 78, no. 4, p. 046110, 2008.
[21] M. Tasgin, A. Herdagdelen, and H. Bingol, "Community detection in complex networks using genetic algorithms," in *Proc. Eur. Conf. Complex Syst.*, Apr. 2006.
[22] C. Shi, Z. Yan, Y. Cai, and B. Wu, "Multi-objective community detection in complex networks," *Appl. Soft Comput.*, vol. 12, no. 2, pp. 850–859, 2012.
[23] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.
[24] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proc. Nat. Acad. Sci.*, vol. 105, no. 4, pp. 1118–1123, 2008.
[25] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, "Finding statistically significant communities in networks," *PloS One*, vol. 6, no. 4, p. e18961, 2011.

[26] P. D. Meo, E. Ferrara, G. Fiumara, and A. Provetti, "Enhancing community detection using a network weighting strategy," *Inf. Sci.*, vol. 222, pp. 648–668, Feb. 2013.
[27] A. F. McDaid, D. Greene, and N. Hurley, "Normalized mutual information to evaluate overlapping community finding algorithms," *arXiv preprint arXiv:1110.2515*, 2011.
[28] A. Lázár, D. Ábel, and T. Vicsek, "Modularity measure of networks with overlapping communities," *Europhysics Letters*, vol. 90, no. 1, p. 18001, 2010.
[29] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, "Propagation of trust and distrust," in *Proc. WWW*, 2004, pp. 403–412.
[30] J. Kunegis, J. Preusse, and F. Schwagereit, "What is the added value of negative links in online social networks?" in *Proc. WWW*, 2013, pp. 727–736.
[31] B. Yang, W. K. Cheung, and J. Liu, "Community mining from signed social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 10, pp. 1333–1348, Oct. 2007.
[32] W. Zhong, J. Liu, X. Xue, and L. Jiao, "A multiagent genetic algorithm for global numerical optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 2, pp. 1128–1141, Apr. 2004.
[33] J. Liu, W. Zhong, and L. Jiao, "A multiagent evolutionary algorithm for constraint satisfaction problems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 1, pp. 54–73, Feb. 2006.
[34] J. Liu, W. Zhong, and L. Jiao, "Moving block sequence and organizational evolutionary algorithm for general floorplanning with arbitrarily shaped rectilinear blocks," *IEEE Trans. Evol. Comput.*, vol. 12, no. 5, pp. 630–646, Oct. 2008.
[35] L. Jiao, J. Liu, and W. Zhong, "An organizational coevolutionary algorithm for classification," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 67–80, Feb. 2006.
[36] J. Liu, W. Zhong, and L. Jiao, "A multiagent evolutionary algorithm for combinatorial optimization problems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 1, pp. 229–240, Feb. 2010.
[37] J. Liu, H. A. Abbass, D. G. Green, and W. Zhong, "Motif difficulty (MD): A predictive measure of problem difficulty for evolutionary algorithms using network motifs," *Evol. Comput. J. (MIT)*, vol. 20, no. 3, pp. 321–347, 2012.
[38] C. Shi, P. S. Yu, Y. Cai, Z. Yan, and B. Wu, "On selection of objective functions in multi-objective community detection," in *Proc. CIKM*, 2011, pp. 2301–2304.
[39] J. Tang, T. C. Lou, and J. Kleinberg. "Inferring social ties across heterogenous networks" in *Proc. WSDM*, 2012, pp 743–752.
[40] D. Easley and J. Kleinberg, *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
[41] A. C. Coello, "Evolutionary multi-objective optimization: A historical view of the field," *IEEE Comput. Intell. Mag.*, vol. 1, no. 1, pp. 28–36, Feb. 2006.
[42] D. E. Goldberg and R. Lingle, "Alleles, loci, and the traveling salesman problem," in *Proc. 1st Int. Conf. Genetic Algorithms*, 1985, pp. 154–159.
[43] J. Knowles and D. Corne, "The Pareto archived evolution strategy: A new baseline algorithm for multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput.*, Jul. 1999, pp. 98–105.
[44] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Proc. EIROGEN*, 2002, pp. 95–100.
[45] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[46] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
[47] A. Mrvar and V. Batagelj, *Pajek and Pajek-XXL, Programs for Analysis and Visualization of Very Large Networks*, Ref. Manual, 2013.
[48] R. Rotta and A. Noack, "Multilevel local search algorithms for modularity clustering," *ACM J. Experimental Algorithmics*, vol. 16, no. 2, pp. 2.3.1–2.3.27, 2011.
[49] Y. Li, J. Liu, and C. Liu, "A comparative analysis of evolutionary and memetic algorithms for community detection from signed social networks," *Soft Comput.*, vol. 18, no. 2, pp. 329–348, 2014.
[50] S. Kropivnik and A. Mrvar, "An analysis of the Slovene parliamentary parties network," in *Proc. Develop. Data Anal.*, 1996, pp. 209–216.
[51] K. E. Read, "Cultures of the central highlands, New Guinea," *Southwestern J. Anthropol.*, vol. 10, no. 1, pp. 1–43, 1954.
[52] P. Doreian and A. Mrvar, "A partitioning approach to structural balance," *Soc. Networks*, vol. 18, no. 2, pp. 149–168, 1996.
[53] A. Arenas, A. Diaz-Guiler, and C. J. Perez-Vicente, "Synchronization reveals topological scales in complex networks," *Phys. Rev. Lett.*, vol. 96, no. 11, p. 114102, Mar. 2006.
[54] A. Arenas, A. Fernández, and S. Gómez, "Analysis of the structure of complex networks at different resolution levels," *New J. Phys.*, vol. 10, no. 5, p. 053039, 2008.
[55] T. Lou, J. Tang, J. Hopcroft, Z. Fang, and X. Ding, "Learning to predict reciprocity and triadic closure in social networks," *ACM Trans. Knowl. Discov. Data*, vol. 7, no. 2, pp. 5:1–25, 2013.
[56] P. F. Lazarsfeld and R. K. Merton, "Friendship as a social process: A substantive and methodological analysis," in *Freedom and Control in Modern Society,* M. Berger, T. Abel, and C. H. Page, Eds. New York, NY, USA: Van Nostrand, 1954, pp. 8–66.
[57] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The EigenTrust algorithm for reputation management in P2P networks," in *Proc. WWW*, 2003, pp. 640–651.
[58] N. Bansal, A. Blum, and S. Chawla, "Correlation clustering," *Mach. Learn.*, vol. 56, nos. 1–3, pp. 89–113, 2004.
[59] E. D. Demaine and N. Immorlica, "Correlation clustering with partial information," in *Proc. Int. Workshop Approx. Algorithms Combinatorial Optimiz. Problems*, 2003, pp. 1–13.
[60] D. Cartwright and F. Harary, "Structural balance: A generalization of Heider's theory," *Psychol. Rev.*, vol. 63, pp. 277–292, Sep. 1956.
[61] F. Heider, "Attitudes and cognitive organization," *J. Psychol.*, vol. 21, no. 1, pp. 107–112, 1946.
[62] P. Doreian, "A multiple indicator approach to block modeling signed networks," *Soc. Netw.*, vol. 30, no. 3, pp. 247–258, 2008.
[63] P. Doreian and A. Mrvar, "Partitioning signed social networks," *Soc. Netw.*, vol. 31, no. 1, pp. 1–11, 2009.
[64] L. Wu, X. Ying, X. Wu, A. Lu, and Z. Zhou, "Spectral analysis of k-balanced signed graphs," in *Proc. 15th Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2011, pp. 1–12.
[65] V. Traag and J. Brugggeman, "Community detection in networks with positive and negative links," *Phys. Rev. E*, vol. 80, p. 036115, Sep. 2009.

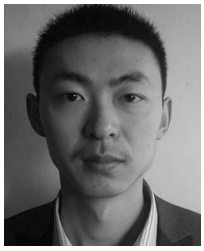


Chenlong Liu received the B.S. degree in materials physics from the Hefei University of Technology, Hefei, China, in 2011, and is currently pursuing the M.S. degree from the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an, China.

His current research interests include evolutionary algorithms, data mining, and complex networks analyses.

Jing Liu (M'05) received the B.S. degree in computer science and technology and the Ph.D. degree in circuits and systems from Xidian University, Xi'an, China, in 2000 and 2004, respectively.

In 2005, she joined Xidian University as a Lecturer, and was promoted to a Full Professor in 2009. From 2007 to 2008, she was a Post-Doctoral Research Fellow with the University of Queensland, Queensland, Australia, and from 2009 to 2011, she was a Research Associate with the University of New South Wales - Canberra, Canberra, Australia. She is currently a Full Professor with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University. Her current research interests include evolutionary computation, complex networks, multiagent systems, and data mining.



Zhongzhou Jiang received the B.S. degree in intelligence science and technology from Xidian University, Xi'an, China, in 2013, where he is currently pursuing the Ph.D. degree in circuits and systems from the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education.

His research interests include complex networks, machine learning, and evolutionary computation.

Mr. Jiang was a recipient of the National Scholarship of China in 2010 and the second prize of the Advanced Mathematics Competition of Shaanxi Province in 2011.