Big Network Analytics Based on Nonconvex Optimization

Maoguo Gong[a,*], Qing Cai[a], Lijia Ma[a], Licheng Jiao[a]

[a]*Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, Xidian University, Xi'an, Shaanxi Province 710071, China*

Abstract

The scientific problems that Big Data faces may be network scientific problems. Network analytics contributes a great deal to networked Big Data processing. Many network issues can be modeled as nonconvex optimization problems and consequently they can be addressed by optimization techniques. In the pipeline of nonconvex optimization techniques, evolutionary computation gives an outlet to handle these problems efficiently. Because, network community discovery is a critical research agenda of network analytics, in this chapter we focus on the evolutionary computation based nonconvex optimization for network community discovery. The single and multiple objective optimization models for the community discovery problem are thoroughly investigated. Several experimental studies are shown to demonstrate the effectiveness of optimization based approach for big network community analytics.

Keywords: Big data, complex networks, nonconvex optimization, evolutionary computation, multiobjective optimization

1. Introduction

Recent years have witnessed the growing enthusiasm for the concept of "Big Data" [86]. Big Data has been an active topic and has attracted great attention from every walk of life [18, 64, 89]. It should be noted that the scientific problems that Big Data faces may be that of network scientific problems, and complex network analytics should be an important cornerstone of data science [125, 71, 1, 114]. Network analytics undoubtedly can contribute a great deal to networked Big Data processing.

Network analytics contains many issues, to name a few, community structure discovery, network structural balance, network robustness, link prediction, network resource allocation, anomaly detection, network security, network recommendation, network propagation, and network ranking, etc. Most if not all of these issues can be modeled as nonconvex optimization problems and consequently they can be computed by optimization techniques. Because, those optimization models for network issues are nonconvex from mathematical view, thus, canonical mathematical optimization methods can hardly solve these problems. In the

*Corresponding author

Email address: gong@ieee.org (Maoguo Gong)

pipeline of optimization techniques, evolutionary computation gives an outlet to handle these nonconvex optimization problems efficiently.

Because network community discovery may be the cornerstone to the analytics of many other network issues, consequently this chapter focuses on the optimization based community structure discovery from networks. The rest of this chapter is organized as follows. Section 2 briefly talks about the issues that network analytics concerns and several eminent properties of networks. Section 3 discusses the basic definitions of optimization and evolutionary computation. Section 4 presents the related work of network community structure analytics, including the definition of a network community and the research progress of community discovery. Section 5 surveys the optimization models for network community discovery. The network data sets commonly used for community discovery benchmarking are listed in Section 6. Section 7 exhibits some experiments on network community discovery, and the conclusions are finally drawn in Section 8.

2. Network issues, properties and notations

2.1. Issues concerning network analytics

Network analytics is an essential research agenda of network and networked big data mining. Fig. 1 shows the importance of network analytics to network and networked data mining. Network analysis not only may very likely result in the discovery of important patterns hidden beneath the networks, but also can potentially shed light on important properties that may control the growth of the networks. Network analytics involves many issues. To move forward, we show 12 critical issues that concern network analytics in Fig. 2.

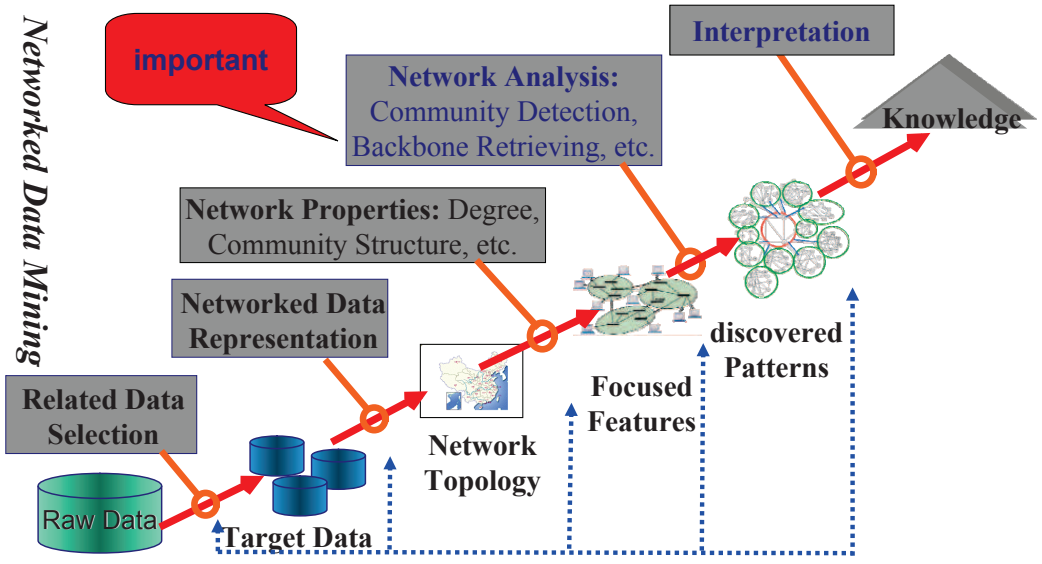

Figure 1: Network analytics plays an important role in network and networked data mining. Reprinted figure with permission from Ref. [74].

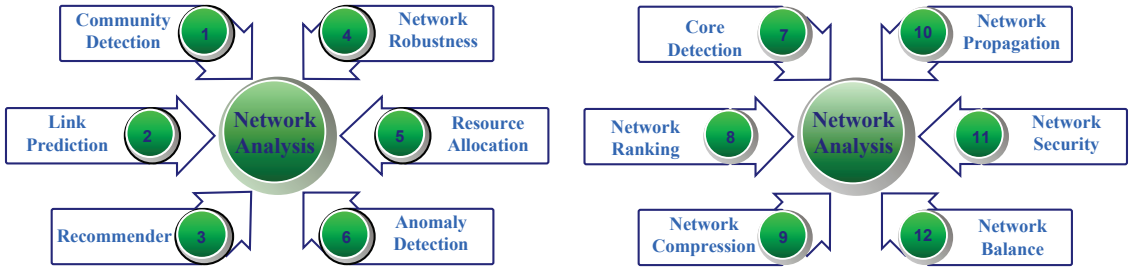

Figure 2: Twelve critical issues that concern network analytics.

Very often, to analyze a network issue one should consider the properties of the corresponding network. In the following, we are going to discuss several eminent properties of networks.

2.2. Eminent properties of network

Because structure always affects function, consequently, a substantial volume of work has been done to analyze the structural properties of complex networks [94, 16, 41, 96, 97]. Networks have many notable properties, such as the small-world property [126], the scale-free property [14], the community structure property [45], etc.

The analysis of network properties is dispensable to network analytics. It is an essential part of network science. Fig. 3 shows some representative properties of networks in the language of graph.

A scale-free network is a network whose degree distribution follows a power law, at least asymptotically. That is, the fraction $P(k)$ of nodes in the network having k connections to other nodes goes for large values of k as

$$P(k) \sim k^{-\gamma} \tag{1}$$

where γ is a parameter whose value is typically in the range $2 < \gamma < 3$, although occasionally it may lie outside these bounds.

A small-world network is a type of mathematical graph in which most nodes are not neighbors of one another, but most nodes can be reached from every other by a small number of hops or steps.

A network with community structure means that the network can be separated into clusters with different sizes, and the similarities between nodes coming from the same cluster are large while from different clusters they are small.

2.3. Graph based network notation

Data sets collected from many different realms can be represented in the form of interaction big networks in a very natural, concise and meaningful fashion. In order to better analyze a big network, one direct way is to represent a network with a graph denoted as $G = \{V, E\}$, where V representing the network objects is the aggregation of vertices, and E representing

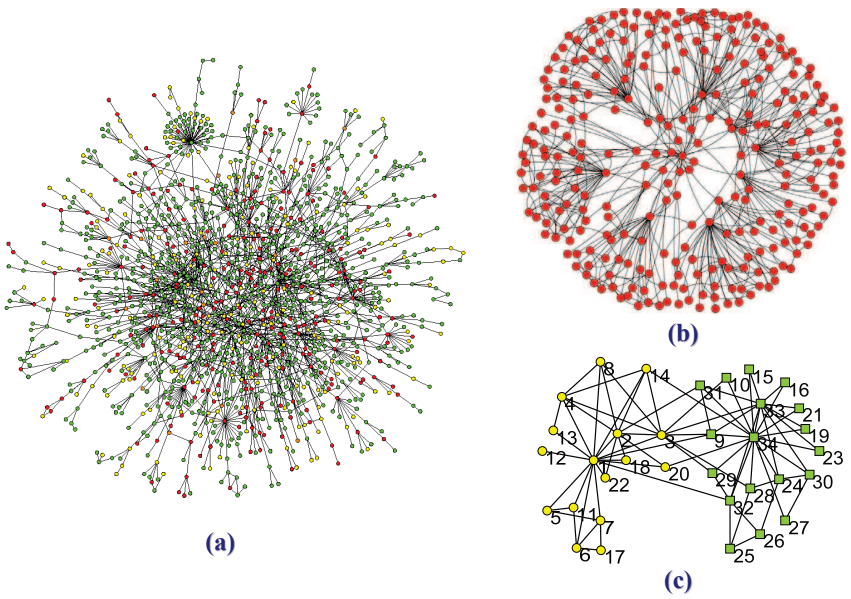

Figure 3: (a) An example of a scale-free network. (b) An example of a small-world network. (c) An example of a network with two communities. Reprinted figure with permission from Ref. [74].

the relations between the objects is the aggregation of edges. Graph G can be denoted by an adjacency matrix $A_{n\times n}$ whose element a_{ij} is defined as:

$$\begin{cases} a_{ij} = \omega_{ij} & if \quad \exists L < i,j > \\ a_{ij} = 0 & if \quad \nexists L < i,j > \end{cases} \tag{2}$$

where $L < i,j >$ represents the link between nodes i and j and ω_{ij} denotes the weight of $L < i,j >$.

In the field of social science, the networks that include both positive and negative edges are called signed social networks [37] or signed networks for short. In signed networks, the so called positive links (L^+) denote positive relationships such as friendship, common interests, and negative links (L^-) may denote negative relationships such as hostility, different interests, and so forth. A signed graph is normally denoted as $G = \{V, PE, NE\}$, where PE and NE represent the aggregations of positive and negative edges, respectively, and the element a_{ij} of the corresponding adjacency matrix $A_{n\times n}$ is defined as:

$$\begin{cases} a_{ij} = \omega_{ij} & if \quad \exists L^+ < i,j > \\ a_{ij} = -\omega_{ij} & if \quad \exists L^- < i,j > \\ a_{ij} = 0 & if \quad \nexists L < i,j > \end{cases} \tag{3}$$

Matrix A is symmetric with the diagonal elements 0, but, if the corresponding network is directed, like the e-mail network, A is asymmetric.

3. Introduction to nonconvex optimization and evolutionary computation

3.1. What is optimization

Optimization has long been an active research topic. Mathematically, a single objective optimization problem (assuming minimization) can be expressed as:

$$\begin{aligned} &\min\ f(x),\ x = [x_1, x_2, ..., x_d] \in \Phi \\ &s.t.\ g_i(x) \leq 0,\ i = 1, ..., m \end{aligned} \tag{4}$$

where x is called the decision vector, d is the number of parameters to be optimized, Φ is the feasible region in decision space, and $g_i(x)$ is the constraint function.

Given that Φ is a convex set, $f(x)$ is said to be convex if $\forall x_1, x_2 \in \Phi, \forall \alpha \in [0,1]$, and the following condition holds:

$$f(\alpha x_1 + (1-\alpha)x_2) \leq \alpha f(x_1) + (1-\alpha) f(x_2) \tag{5}$$

Particularly, $f(x)$ is strictly convex if $\forall x_1 \neq x_2 \in \Phi, \forall \alpha \in (0,1)$, and the following condition holds:

$$f(\alpha x_1 + (1-\alpha)x_2) < \alpha f(x_1) + (1-\alpha) f(x_2) \tag{6}$$

If $f(x)$ and $g_i(x)$ are all convex, then we call Eq. 4 as a convex optimization problem. For a strictly convex optimization problem, there is at most one minimal solution which is also the global one. In real applications, the functions $f(x)$ and $g_i(x)$ may be nonconvex and there may exist many local and/or global minimum. In this respect, we call Eq. 4 as a nonconvex optimization problem. As a matter of fact, many real-world optimization problems are nonconvex [56, 92].

In reality, many optimization problems involve multiple objectives, i.e., there are more than one $f(x)$ to be optimized. A multiobjective optimization problem can be mathematically formulated as:

$$\min\ F(x) = (f_1(x), f_2(x), ..., f_k(x))^T \tag{7}$$

The objectives in Eq. 7 often conflict with each other. Improvement of one objective may lead to deterioration of another. Thus, a single solution, which can optimize all objectives simultaneously, does not exist. For multi-objective optimization problems, the aim is to find good compromises (trade-offs) which are also called Pareto optimal solutions. The Pareto optimality concept was first proposed by Edgeworth and Pareto. To understand the concept, here are some related definitions.

- **Definition 1** (Pareto Optimality): A point $x^* \in \Phi$ is Pareto optimal if for every $x \in \Phi$ and $I = \{1, 2, ..., k\}$ either $\forall i \in I$, $f_i(x) = f_i(x^*)$ or, there is at least one $i \in I$ such that $f_i(x) > f_i(x^*)$.

- **Definition 2** (Pareto Dominance): Given two vectors $x, y \in \Phi$, where $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$, we say that x dominates y (denoted as $x \prec y$), if $x_i \leqq y_i$ for

$i = 1, 2, ..., n$, and $x \neq y$. x is nondominated with respect to Φ, if there does not exist another $x' \in \Phi$ such that $F(x') \prec F(x)$.

- **Definition 3** (Pareto Optimal Set): The set of all Pareto optimal solutions is called Pareto Optimal Set which is defined as:

$$PS = \{x \in \Phi | \neg \exists x^* \in \Phi, F(x^*) \prec F(x)\} \tag{8}$$

- **Definition 4** (Pareto Front): The image of the Pareto set (PS) in the objective space is called the Pareto front (PF) which is defined as:

$$PF = \{F(x) | x \in PS\} \tag{9}$$

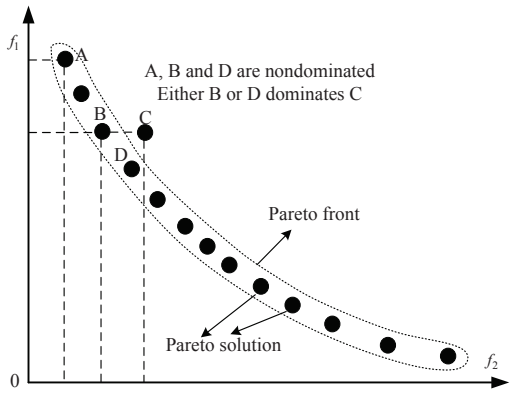

Figure 4: Graphical illustration of Pareto optimal solution and Pareto front.

Fig. 4 gives an example of the above mentioned definitions. Each dot except that labeled by C in the figure represents a nondominated solution to the optimization problem. The aim of a multiobjective optimization algorithm is to find the set of those nondominated solutions approximating the true PF.

3.2. How to tackle optimization problems

In the field of optimization, evolutionary computation, a class of intelligent optimization techniques, has been proved to be an efficient tool for solving nonconvex optimization problems. In the last several decades, many evolutionary algorithms (EAs) originated from the evolution principles and behavior of living things, have sprung out and have found nationwide applications in the optimization domain [34, 31]. Most if not all of the EAs share the following commom properties:

1. They are population based stochastic searching methods. A population consists of a set of individuals, each individual represents a solution to the optimization problem. An evolutionary algorithm optimizes the problem by having a population of initialized solutions and then apply stochastic components to generate new solutions in the decision space.

2. They are recursively iterative methods. These methods iteratively search for optimal solutions in the search space. The search process will not stop untill the maximum iteration number or a prescribed threshold is reached.
3. They have some inherent parameters, like the population size and the maximum iteration number, etc. These parameters are normally set empirically.

Algorithm 1 General framework of evolutionary algorithms.

Input: algorithm parameters, problem instance
Output: optimal solutions to the optimization problem

1. **Begin**
2. population initialization
3. store optimal solutions
4. **for** *i*=1 to max_iteration **do**
   (a) **for** each individual in the population, **do**
      i. generate a new individual through stochastic components
      ii. evaluate the fitness of the new individual
   (b) **end for**
   (c) update optimal solutions
5. **end for**
6. **End**

A general framework of EAs is shown in Algorithm 1. In the last few years, many efforts have been devoted to the application of EAs to the development of multiobjective optimization. A lot of multiobjective evolutionary algorithms (MOEAs) have been proposed, e.g., [66, 35, 135, 30, 133, 53, 13, 11, 131].

4. Community structure analytics

Community structure discovery is one of the cornerstones of network analytics. It can provide useful patterns and knowledge for further network analysis. This section is dedicated to summarizing the related works for community structure analytics.

4.1. Description of community discovery

Network community discovery plays an important role in the networked data mining field. Community discovery helps to discover latent patterns in networked data and it affects the ultimate knowledge presentation.

As illustrated above, a complex network can be expressed with a graph that is composed of nodes and edges. The task for network community discovery is to separate the whole network into small parts which are also called communities. There is no uniform definition for community in the literature, but in academic domain, a community, also called a cluster

or a module, is normally regarded as a groups of vertices which probably share common properties and/or play similar roles within the graph. Fig. 5 exhibits the community discovery problem under different network scenarios.

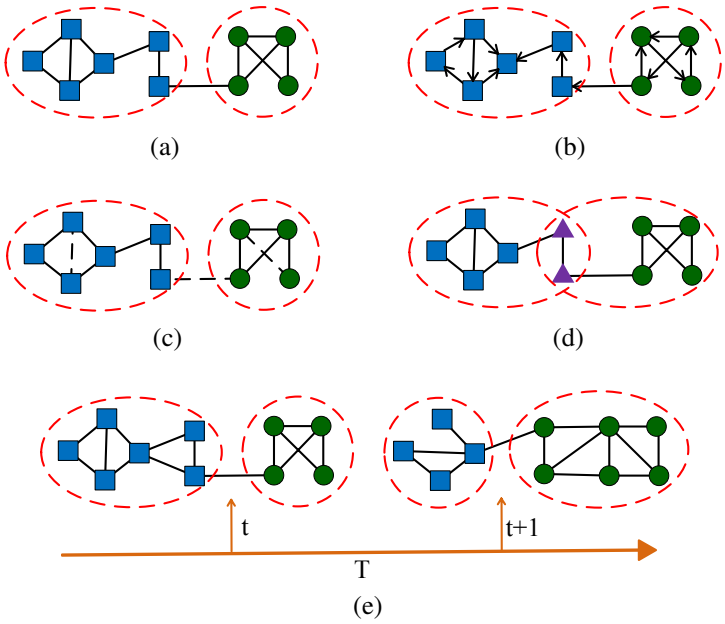

Figure 5: Graphical illustration of community discovery. (a) common model, (b) directed model, (c) signed model, (d) overlapping model and (d) dynamic model.

From Fig. 5 we can notice that community discovery under dynamic context is quite different from the others. In a dynamic network, the community structure is temporally changed. How to design algorithms to uncover time-varying communities is challenging.

4.2. Qualitative community definition

In order to formalize the qualitative community in unsigned network, Radicchi et al. in [107] gave a definition based on node degree. Given a network represented as $G = (V, E)$, where V is the set of nodes and E is the set of edges. Let k_i be the degree (the number of links that have connections with node i) of node i and A be the adjacency matrix of G. Given that $S \subset G$ is a subgraph, let $k_i^{in} = \sum_{i,j \in S} A_{ij}$ and $k_i^{out} = \sum_{i \in S, j \notin S} A_{ij}$ be the internal and external degree of node i, then S is a community in a strong sense if

$$\forall i \in S, \quad k_i^{in} > k_i^{out} \tag{10}$$

S is a community in a weak sense if

$$\sum_{i \in S} k_i^{in} > \sum_{i \in S} k_i^{out} \tag{11}$$

The above community definition only fits for unsigned networks. In [48] the authors give a definition under signed context. Given a signed network modeled as $G = (V, PE, NE)$, where PE and NE are the set of positive and negative links, respectively. Given that $S \subset G$ is a subgraph, let $(k_i^+)^{in} = \sum_{j \in S, L_{ij} \in PE} A_{ij}$ and $(k_i^-)^{in} = \sum_{j \in S, L_{ij} \in NE} |A_{ij}|$ be the positive and negative internal degree of node i, respectively. Then S is a community in a strong sense if

$$\forall i \in S, \quad (k_i^+)^{in} > (k_i^-)^{in} \tag{12}$$

Let $(k_i^-)^{out} = \sum_{j \notin S, L_{ij} \in NE} |A_{ij}|$ and $(k_i^+)^{out} = \sum_{j \notin S, L_{ij} \in PE} A_{ij}$ be the negative and positive external degree of node i, respectively. Then S is a community in a weak sense if

$$\begin{cases} \sum_{i \in S} (k_i^+)^{in} > \sum_{i \in S} (k_i^+)^{out} \\ \sum_{i \in S} (k_i^-)^{out} > \sum_{i \in S} (k_i^-)^{in} \end{cases} \tag{13}$$

The above definitions only give the conditions that a community should satisfy, but they have not told how good on earth a community is. Therefore, there should have quantitative indexes that can measure the quality of a community. These indexes will be illustrated in Section 5.

4.3. Existing approaches for community discovery

In the literature, a large amount of methods have been proposed to discover communities in big networks. Roughly, these methods can be divided into two categories: optimization based class and non-optimization based class.

Table 1: Representative non-optimization based methods for big network community discovery.

Method	Ref.	Key Technique	Network Scale
CNM	[29]	greedy optimization + sophisticated data structure	medium
LPA	[12]	mark each node with a label and then let them propagate	very large
Infomod	[111]	information compression, transmission, and decoding	large
FEC	[130]	random walk + cutoff function	very large
BGLL	[15]	fast hierarchical modularity optimization	medium
Infomap	[112]	clustering + information compression + random walks	large

For the non-optimization based avenues, in Table 1 we list several outstanding methods that can handle big networks. For more information about the existing community discovery methods developed before 2012, please refer to [41, 129].

As for the optimization based methods, most of them are nonconvex. The essence of them is to model the network community discovery task as different optimization problems and then design suitable nonconvex optimization methods such as EAs to deal with them. As what follows we will summarize the optimization models for community structure analytics.

5. Optimization models for community structure analytics

5.1. Single objective optimization model

5.1.1. Modularity based model

The most popular evaluation criterion for community detection is the modularity (normally denoted as Q) proposed by Newman and Girvan in [98]. The modularity index can be given in the following form:

$$Q = \frac{1}{2m}\sum_{i,j}^{n}\Big(A_{ij} - \frac{k_i \cdot k_j}{2m}\Big)\delta(i,j) \tag{14}$$

where n and m are the number of nodes and edges of a network, respectively. $\delta(i,j) = 1$, if node i and j are in the same group, otherwise, 0. By assumption, higher values of Q indicate better partitions.

Q is very popular, a lof of bio-inspired metaheuristics have been utilized to optimize Q to find the community structure with biggest Q value [46, 124, 85, 128, 82, 60, 123, 119, 63, 76, 62, 43, 61, 115, 75, 77, 78, 22, 88]. However, Q has several drawbacks. First, to maximize Q is proved to be NP-hard [19]. Second, large Q value does not always make sense. Random networks with no community structures can also possess high Q values [59, 110]. Third, which is also the most important, Q has the resolution limitation [42], i.e., maximizing Q cannot discover communities whose sizes are smaller than a scale which depends on the total size of the network and on the degree of inter connectedness of the modules, even in the case scenario where modules are unambiguously defined.

To overcome these demerits, many researchers have devoted themselves to designing efficient operators for the optimization algorithms to enhance the exploration and exploitation; some scholars make efforts to design new evaluation criteria, such as extended modularity [110, 10, 106], multi-resolution index [80], and so forth. Because Q is originally designed for unsigned, unweighted, undirected, nonoverlapped and static networks, thus, many creative jobs have been done to extend Q to handle other types of networks.

Gómez et al. in [47] presented a reformulation of Q that allows the analysis of weighted, signed, and netwoks that have self-loops. The presented Q is formulized as:

$$Q_{sw} = \frac{1}{2(w^+ + w^-)}\sum_{i,j}\Big[w_{ij} - \Big(\frac{w_i^+ w_j^+}{2w^+} - \frac{w_i^- w_j^-}{2w^-}\Big)\Big]\delta(i,j) \tag{15}$$

where w_{ij} is the weight of the signed adjacency matrix, $w_i^+(w_i^-)$ denotes the sum of all positive (negative) weights of node i. Based on the Q_{sw} metric, the authors in Cai et al. [23] suggested a discrete particle swarm optimization (DPSO) algorithm to detect communities from signed networks.

Q_{sw} can be easily changed to handle directed, weighted graphs [8, 72, 113], and the expression of directed and weighted Q reads:

$$Q_{dw} = \frac{1}{w}\sum_{i,j}\Big(A_{ij} - \frac{w_i^{out} \cdot w_j^{in}}{w}\Big)\delta(i,j) \tag{16}$$

where w_i^{out} (w_i^{in}) denotes the out-degrees (in-degrees) of node i. It can be noticed that the factor 2 is removed because the sum of the in-degrees (outdegrees), the number of non-vanishing elements of the asymmetric adjacency matrix, all equal w.

In the case when a node may belong to more than one community, Q has been modified to fit overlapping communities [134, 99, 117], and a general expression reads:

$$Q_{ov}(U_k) = \sum_{c=1}^{k} \left[\frac{A(\overline{V}_c, \overline{V}_c)}{A(V,V)} - \left(\frac{A(\overline{V}_c, V)}{A(V,V)} \right)^2 \right] \tag{17}$$

where $U_k = [u_1, u_2, \ldots, u_k]$ is a fuzzy parition of the nodes of the network into k clusters. $A(\overline{V}_c, \overline{V}_c) = \sum_{i \in \overline{V}_c} \sum_{j \in \overline{V}_c} ((u_{ic} + u_{jc})/2) w_{ij}$, where $\overline{V}_c$ is the set of vertices in community c, $A(\overline{V}_c, V) = A(\overline{V}_c, \overline{V}_c) + \sum_{i \in \overline{V}_c} \sum_{j \in V - \overline{V}_c} ((u_{ic} + (1 - u_{jc}))/2) w_{ij}$ and $A(V,V) = \sum_{i \in V} \sum_{j \in V} w_{ij}$. u_{ic} is the membership value that node i belongs to community c.

The existing overlapping community detection methods can be roughly divided into two categories, the node-based (directly cluster nodes) and the link-based (cluster links and then map link communities to node communities) ones, but the the mainstream for single solution based overlapping community detection is to first utilize soft clustering technique such as fuzzy K-means to find a fuzzy partition of the nodes of a network into k clusters, and then apply a criterion to choose the best overlapping network partition [134, 68, 70, 109]. The key technique lies in the evaluation of an overlapped community. As long as an evaluation criterion is decided, bio-inspired metaheuristics can be easily utilized to solve this problem [104, 84, 24, 81]. For more information about the fitness evaluation for overlapping communities, please refer to [28, 129].

Other extended criteria such as the local modularity can be found in [90, 93], the triangle modularity in [9] and the bipartite modularity in [58].

5.1.2. Multi-resolution model

To overcome the resolution limitation of modularity, many multi-resolution models have been developed. Pizzuti in [102] proposed a genetic algorithm for community detection. The highlight of the work is the suggested community score (CS) evaluation metric. Let $\mu_i = \frac{1}{|S|} k_i^{in}$ be the fraction of edges connecting node i to the other nodes in S and $M(S) = \frac{\sum_{i \in S} (\mu_i)^r}{|S|}$ be the power mean of S of order r. $|S|$ is the cardinality of S, i.e., the number of nodes in S. We further define $v_S = \frac{1}{2} \sum_i k_i^{in}$ be the volume of S, i.e., the number of edges connecting vertices inside S, then the score of S is defined as $score(S) = M(S) \times v_S$. Assume that G has a partition of k subgraphs, i.e., $\Omega = \{S_1, S_2, ..., S_k\}$, then CS can be written as:

$$CS = \sum_{i=1}^{k} score(S_i) \tag{18}$$

The CS metric takes one parameter r which is hard to tune. The author claims that higher values of the exponent r bias the CS towards matrices containing a low number of zeroes, i.e., higher values of r help in detecting communities.

Li et al. in [80] put forward the modularity density (D) index. D can break the resolution limitation brought by Q. For an unsigned network, let us define $L(S_a, S_b) = \sum_{i \in S_a, j \in S_b} A_{ij}$ and $L(S_a, \overline{S_a}) = \sum_{i \in S_a, j \in \overline{S_a}} A_{ij}$, where $\overline{S_a} = \Omega - S_a$. Then D is defined as:

$$D_\alpha = \sum_{i=1}^{k} \frac{2\alpha L(S_i, S_i) - 2(1-\alpha)L(S_i, \overline{S_i})}{|S_i|} \tag{19}$$

where $\alpha \in [0,1]$ is a resolution control parameter. D_α can be viewed as a combination of the ratio association and the ratio cut [36]. Generally, optimize the ratio association algorithm often divides a network into small communities, while optimize the ratio cut often divides a network into large communities. By tuning the α value, we can use this general function to uncover more detailed and hierarchical organization of a complex network. Based on modularity density, many algorithms have emerged [25, 27, 51, 49, 79, 21].

5.2. Multi-objective optimization model

Many real-world optimization problems involve multiple objectives. From the statement of the community detection problem discussed earlier we can notice that, community detection can also be modeled as multiobjective optimization problems. Many multiobjective optimization based community detection methods have been developed in this respect. Each run of these methods can yield a set of community partitions for the decision maker to choose. The most important point for these methods should own to their abilities for breaking through the resolution limit of modularity. As stated earlier, components used in single objective optimization models, such as the individual representation, recombination, etc., serve multiobjective optimization models as well. This section primarily deals with the multiobjective community detection models.

5.2.1. General model

As stated earlier, for an unsigned network, the links within a community should be dense while the links between communities should be sparse, as for a signed network, the inter and intra links should all be dense. On the basis of this property, many multiobjective community models are established.

Pizzuti in [105] and [103] proposed a multiobjective genetic algorithm-based method called MOGA-Net. In this method, the author modeled the community detection task as a multiobjective optimization problem and then applied the fast elitist non-dominated sorting genetic algorithm (NSGA-II) [35] framework to solve it. The two objectives introduced are the *CS* and the *CF*. Thus, the proposed optimization model is:

$$\max \left\{ \begin{array}{l} f_1 = CS \\ f_2 = -CF \end{array} \right\} \tag{20}$$

CF (community fitness) is a criterion put forward by Lancichinetti in [68]. *CF* is formulated as:

$$CF = \sum_{S \in \Omega} \sum_{i \in S} \frac{k_i^{in}}{k_i} \tag{21}$$

From the formulation of *CF* and *CS* we may notice that, *CF* to some extent measures the link density within communities, while *CS* can be regarded as an index to measure the averaged degrees within communities.

An improved version of MOGA-Net can be found in [20]. To optimize the above model, other metaheuristics, such as the multi–objective enhanced firefly algorithm [6], hybrid evolutionary algorithm based on HSA (harmony search algorithm [44]) and CLS (chaotic local search) [5, 4, 7], non-dominated neighbor immune algorithm [52], have all find their niche in community detection.

In [54] the authors presented a multiobjective evolutionary algorithm based on decomposition (MOEA/D) based method. MOEA/D is proposed by Zhang and Li in [133]. The highlight of this work is the newly cranked out multiobjective community optimization model which optimizes two objectives termed as *NRA* (Negative Ratio Association) and *RC* (Ratio Cut). The optimization model is:

$$\min \left\{ \begin{array}{l} NRA = -\sum_{i=1}^{k} \frac{L(S_i,S_i)}{|S_i|} \\ RC = \sum_{i=1}^{k} \frac{L(S_i,\overline{S_i})}{|S_i|} \end{array} \right\} \tag{22}$$

It can be noticed that Eq. 22 is the decomposition of Eq. 19. *RC* measures the link density between two communities and *RA* calculates the link density within a community. To minimize *NRA* and *RC* we can ensure that the connections within a community is dense and the links between communities are sparse. A similar optimization model can be found in [50].

Other optimization models such as maximizing the combinations of Q and *CS* can be found in [2], and maximizing the two parts of the Q index, i.e., Q is decomposed into two objectives, can be found in [120]. A three objectives model can be found in [116]. Small surveys on the selection of objective functions in multiobjective community detection can be found in [121, 122].

5.2.2. Signed model

Many social networks involve friendly and hostile relations between the objects that compose the networks. These networks are called signed networks. In [48] the authors put forward a novel discrete multiobjective PSO framework for community detection. To handle signed networks, the authors have suggested a signed optimization model which optimizes two objectives named as *SRA* (Signed Ratio Association) and *SRC* (Signed Ratio Cut). The optimization model reads:

$$\min \left\{ \begin{array}{l} SRA = -\sum_{i=1}^{k} \frac{L^+(S_i,S_i) - L^-(S_i,S_i)}{|S_i|} \\ SRC = \sum_{i=1}^{k} \frac{L^+(S_i,\overline{S_i}) - L^-(S_i,\overline{S_i})}{|S_i|} \end{array} \right\} \tag{23}$$

where $L^+(S_i,S_j) = \sum_{i\in S_i, j\in S_j} A_{ij}, (A_{ij} > 0)$ and $L^-(S_i,S_j) = \sum_{i\in S_i, j\in S_j} |A_{ij}|, (A_{ij} < 0)$. To minimize *SRA* and *SRC* we can make sure that the positive links within a community are dense while the negative links between communities are also dense, which is in accordance with the feature of signed community.

In [3] the authors put forward another signed optimization model which uses the NSGA-II framework to optimize it. The model reads:

$$\min \left\{ \begin{array}{l} f_1 = -Q_{sw} \\ f_2 = frustration \end{array} \right\} \tag{24}$$

where $frustration = \sum_{i,j}^{n}(A_{ij}^{+}(1-\delta(i,j)) - A_{ij}^{-}\delta(i,j))$. The first objective Q_{sw} measures how good a signed community is and to minimize $frustration$ we will ensure that the sum of the negative links within a community and the positive links between difference communities are minimum.

Recently, to detect communities from signed networks, the authors in [83] put forward a signed optimization model based on node similarity. The optimization model is as follows:

$$\max \left\{ \begin{array}{l} f_{pos-in}(\Omega) = \dfrac{1}{k}\displaystyle\sum_{i=1}^{k} \dfrac{P_{in}^{S_i}}{P_{in}^{S_i} + P_{out}^{S_i}} \\ f_{neg-out}(\Omega) = \dfrac{1}{k}\displaystyle\sum_{i=1}^{k} \dfrac{N_{out}^{S_i}}{N_{in}^{S_i} + N_{out}^{S_i}} \end{array} \right\} \tag{25}$$

where $P_{in}^{S_i}$ (or $P_{out}^{S_i}$) is the internal (or external) positive similarity of community S_i, and $N_{in}^{S_i}$ (or $N_{out}^{S_i}$) is the internal (or external) negative similarity of community S_i. See reference [83] for more information about the similarity of a community. To maximize f_{pos-in} we can ensure high positive similarities within communities, and to maximize $f_{neg-out}$ we can guarantee high negative similarities between different communities.

5.2.3. Overlapping model

In real world, a node of a network may belong to more than one community, just like the friendship network. From the perspective of finding overlapping communities, intuitively, the nodes that connect multiple communities with similar strength are more likely to be overlapping nodes. For instance, if node i has both l links with community a and b, then we can regard i as an overlapping node. From the viewpoint of finding nonoverlapping or separated communities, the less the number of overlapping nodes, the more the separated communities.

Based on the above principle, the authors in [84] put forward a three objectives optimization model reads:

$$\max \left\{ \begin{array}{l} f_1 = f_{\text{quality}}(\Omega) = \dfrac{CF}{k} \\ f_2 = f_{\text{separated}}(\Omega) = -\mid V_{overlap} \mid \\ f_3 = f_{\text{overlapping}}(\Omega) = \displaystyle\sum_{i \in V_{overlap}} \min_{s \in \Omega} \{\frac{k_i^s}{k_i}\} \end{array} \right\} \tag{26}$$

where k_i^s denotes the number of edges connect node i and community s, $V_{overlap}$ is the set of the overlapping nodes. To maximize f_2 and f_3 one can get a tradeoff between nonoverlapping and overlapping communities.

5.2.4. Dynamical model

In reality, networks may evolve with the time, the nodes and the links may disappear or new nodes may just come out, therefore, the community structures are also changing according to the time. However, traditional approaches mostly focuse on static networks for small groups. As the technologies move forward, in the presence of big data, how to design methods and tools for modeling and analyzing big dynamic networks is a challenging research topic in the years to come. To analyze the community structures of dynamical networks will help to predict the change tendency which may give support to the analysis of other network or networked scientific issues. Community detection in dynamic networks is challenging.

Dynamic community detection is normally based on a temporal smoothness framework which assumes that the variants of community division in a short time period are not desirable [39]. According to the temporal smoothness framework, the community detection in dynamic networks can be naturally modeled as a bi-objective optimization problem. The optimization of one objective is to reveal a community structure with high quality at this moment, and the optimization of the other objective is to uncover a community structure at the next moment which is highly similar with that at the previous time [38, 39, 55, 26, 40]. The commonly used dynamical optimization model can be written as:

$$\max \begin{cases} f_1 = CS \text{ or } Q \text{ or } D_\alpha \\ f_2 = NMI \end{cases} \tag{27}$$

NMI, *Normalized Mutual Information* [33], comes from the field of information theory. *NMI* can be regarded as a similarity index. For the community detection problem, given that A and B are two partitions of a network, respectively, C is a confusion matrix, C_{ij} equals to the number of nodes shared in common by community i in partition A and by community j in partition B. Then $NMI(A,B)$ is written as:

$$NMI = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} C_{ij} log(C_{ij} \cdot n / C_{i.} C_{.j})}{\sum_{i=1}^{C_A} C_{i.} log(C_{i.}/n) + \sum_{j=1}^{C_B} C_{.j} log(C_{.j}/n)} \tag{28}$$

where C_A (or C_B) is the number of clusters in partition A(or B), $C_{i.}$ (or $C_{.j}$) is the sum of elements of C in row i(or column j). $NMI(A,B) = 1$ means that A and B are identical and $NMI(A,B) = 0$ indicates that A and B are completely different.

The first objective in Eq. 27 is the snapshot cost which measures how well a community structure A is at time t and the second objective is the temporal cost which measures how similar the community structure B is at time $t+1$ with the previous community structure A.

Another dynamical model which maximizes the Min-max cut and global silhouette index can be found in [65].

6. Network data sets

This section will list the network data sets commonly used in the literature for testing purpose. The data sets contain two types, artificial benchmark networks and real-world networks. Benchmark networks have controlled topologies. They are used to mimic real-world

networks. Different real-world networks may have different properties. Hence, real-world networks are still needed for testing purpose.

6.1. Artificial generated benchmark networks

6.1.1. GN benchmark and its extended version

Girvan and Newan (GN) in [45] put forward a benchmark network generator which is normally recognized as the GN benchmark. For a GN benchmark network, it was constructed with 128 vertices divided into four communities of 32 vertices each. Edges were placed between vertex pairs independently at random, with probability P_{in} for vertices belonging to the same community and P_{out} for vertices in different communities, with $P_{out} < P_{in}$. The probabilities were chosen so as to keep the average degree z of a vertex equal to 16.

An extended version of the GN model was introduced in [32]. The extended benchmark network also consists of 128 nodes divided into four communities of 32 nodes each. Every node has an average degree of 16 and shares a fraction γ of links with the rest in its community, and $1-\gamma$ with the other nodes of the network. Here, γ is called the mixing parameter. When $\gamma < 0.5$, the neighbours of a vertex inside its community are more than the neighbors belonging to the rest groups.

6.1.2. LFR benchmark

Standard benchmarks, like the GN benchmark or its extended version, do not account for important features in graph representations of real systems, like the fat-tailed distributions of node degree and community size, since on those benchmark networks, all vertices have approximately the same degree, moreover, all communities have exactly the same size by construction.

To overcome these drawbacks, a new class of benchmark graphs have been proposed by Lancichinetti, Fortunato, and Radicchi (LFR) in [69], in which the distributions of node degree and community size are both power laws with tunable exponents. They assume that the distributions of degree and community size are power laws, with exponents τ_1 and τ_2, respectively. Each vertex shares a fraction $1-\mu$ of its edges with the other vertices of its community and a fraction μ with the vertices of the other communities; $0 \leq \mu \leq 1$ is the mixing parameter. The software to create the LFR benchmark graphs can be freely downloaded at `http://santo.fortunato.googlepages.com/inthepress2`. In our experiments, we generate 17 networks with the mixing parameter increasing from 0 to 0.8 with an interval of 0.05.

6.1.3. Signed LFR benchmark

The LFR network generator is a reliable model for benchmarking. However, this model is originally designed for unsigned networks. In order to mimic signed social networks, The LFR model can be extended into signed version. Here we give a feasible way to do so.

A signed LFR model can be depicted by $SLFR(n, k_{avg}, k_{max}, \gamma, \beta, s_{min}, s_{max}, \mu, p-, p+)$, where n is the number of nodes; k_{avg} and k_{max} are the averaged and maximum degree of a node, respectively; γ and β are the exponents for the power law distribution of node degree and community size, respectively; s_{min} and s_{max} are the minimum and maximum community

size, respectively. μ is a mixing parameter. Each node shares a fraction $1-\mu$ of its links with the other nodes of its community and a fraction μ with the other nodes of the network. $p-$ is the fraction of negative edges within communities, and $p+$ is the fraction of positive edges between different communities.

6.2. Real-world networks

Table 2: Eight commonly tested unsigned networks.

Network	#Node	#Edge	#Clusters	$\overline{k}$	Ref.
Karate	34	78	2	4.588	[132]
dolphin	62	159	2	5.129	[87]
football	115	613	12	10.661	[45]
SFI	118	200	unknown	3.390	[45]
e-mail	1133	5451	unknown	9.622	[57]
netscience	1589	2742	unknown	3.451	[95]
power grid	4941	6594	unknown	2.669	[126]
PGP	10680	24340	unknown	4.558	[17]

Table 3: Eight commonly tested signed networks.

Network	#Node	#Edge	m^+	m^-	$\overline{k}$	Ref.
SPP	10	45	18	27	9.000	[67]
GGS	16	58	29	29	7.250	[108]
EGFR	329	779	515	264	4.736	[101]
Macrophage	678	1,425	947	478	4.204	[100]
Yeast	690	1,080	860	220	3.130	[91]
Ecoli	1,461	3,215	1,879	1,336	4.401	[118]
WikiElec	7,114	100,321	78,792	21,529	28.204	[73]
Slashdot	77,357	466,666	352,890	113,776	12.065	[73]

Tables 2 and 3 list the parameters of 8 commonly tested unsigned and signed networks. In the Tables, m^+ and m^- denote the numbers of positive and negative edges, respectively. $\overline{k}$ is the averaged node degree.

6.3. Famous websites

Apart from the above mentioned network data sets, many other network data sets are available on the Internet. In this part we list several famous websites as follows:

- http://www-personal.umich.edu/∼mejn/ (Mark Newman Website)
- http://deim.urv.cat/∼aarenas/data/welcome.htm (Alex Arenas Website)

- http://snap.stanford.edu/index.html (Stanford Network Analysis Project. Diverse kinds of network data and graphical visualization softwares and tools and useful codes are available.)

- http://www.correlatesofwar.org/ (The Correlates of War Project. A large amount of signed networks mainly related to war are free to access.)

- http://www.gmw.rug.nl/∼huisman/sna/software.html (A collection of softwares for social network analysis.)

- http://tuvalu.santafe.edu/∼aaronc/hierarchy/ (Hierarchical Random Graphs)

7. Experimental exhibition

In [22] we have suggested a greedy discrete particle swarm optimization algorithm (GDPSO) for big network community discovery. The GDPSO algorithm optimizes the modularity index. As what follows we will show its performance over several real-world networks.

Table 4: Averaged modularity values obtained by five methods over 30 independent runs.

network	GDPSO	CNM	BGLL	Infomap	LPA
Karate	0.4198	0.3800	0.4180	0.4020	0.3264
dolphin	0.5280	0.4950	0.5188	0.5247	0.4964
football	0.6041	0.5770	0.6046	0.6005	0.5848
e-mail	0.4783	0.4985	0.5412	0.5355	0.0070
power grid	0.8368	0.9229	0.7756	0.8140	0.7476
PGP	0.8013	0.8481	0.9604	0.7777	0.7845

Table 4 lists the averaged modularity values obtained by five methods over 30 independent runs on six networks. The GDPSO algorithm is an optimization based method. GDPSO is competitive to the rest four methods in terms of the modularity index.

On one hand, it is natural to model network community discovery as a multiobjective optimization problem. On the other hand, based on the preliminary shown in subsection 3.1, we can get to know that a single run of a MOEA based community discovery method can output a set of solutions, as shown in Fig. 6.

As can be seen from Fig. 6 that each Pareto solution denotes a certain network community structure. However, each single run of the methods listed in Table 4 can only output one solution. There is no doubt that the MOEA based community discovery facilitates intelligent multi-criteria decision making. For more exhibitions about the MOEA based community discovery please refer to our recent work in [48].

It should be noted that based on the optimization models discussed in Section 5, one can design different single objective EAs or MOEAs to optimize those models. However, according to the NFL (No Free Lunch) theory [127], there is no one-for-all method that can deal with all kinds of networks. For one thing, for different network issues, we can solve

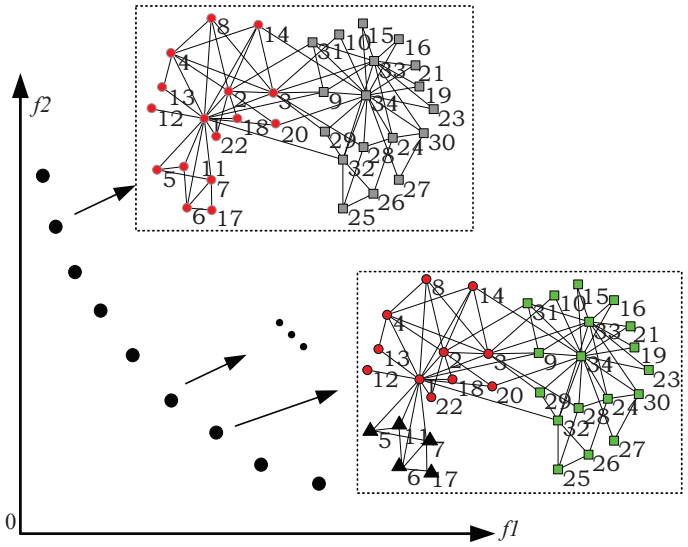

Figure 6: An illustration of the Pareto front obtained by an MOEA for community discovery from the Karate network.

them well as long as we can establish a good optimization model that can well depict the nature of those problems. For another thing, we should make efforts to enhance the search abilities of the optimization algorithms. Meanwhile, different networks have different space-time properties. Consequently, we should take into account the special characters of the networks when designing algorithms to solve network issues.

8. Concluding remarks

Network analysis is one of the theoretical underpinnings of big data. Network community discovery serves as the backbone of network analysis. The past decades have witnessed the prosperity of the research on community discovery. A large number of techniques have been cranked out to discover communities in the networks. Among the extant avenues for solving the network community discovery problem, many of them are nonconvex optimization based.

This chapter tries to investigate the network community discovery problem from the optimization view. Single objective and multiobjective optimization models for network community discovery problems are delineated. Experimental studies are also shown to demonstrate the promise of the optimization based idea for network analytics.

We expect that complex network analysis's scope will continue to expand and its applications to multiply. We are positive that methods and theories that work for community detection are helpful for other network issues. From both theoretical and technological perspectives, network community discovery technology will move beyond network analytics toward emphasizing network intelligence. We do hope that this chapter can benefit scholars who set foot in this field. Our future work will focus on more in-depth analysis of network issues. Such analysis is expected to shed light on how networks change the real world.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant nos. 61273317, 61422209, and 61473215), the National Top Youth Talents Program of China, and the Specialized Research Fund for the Doctoral Program of Higher Education (Grant no. 20130203110011).

References

[1] Aggarwal, C. C., 2011. An introduction to social network data analytics. Springer.

[2] Agrawal, R., 2011. Bi-objective community detection (bocd) in networks using genetic algorithm. In: Contemporary Computing. pp. 5–15.

[3] Amelio, A., Pizzuti, C., 2013. Community mining in signed networks: a multiobjective approach. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp. 95–99.

[4] Amiri, B., Hossain, L., Crawford, J., 2012. A hybrid evolutionary algorithm based on hsa and cls for multi-objective community detection in complex networks. In: Advances in Social Networks Analysis and Mining (ASONAM), 2012 IEEE/ACM International Conference on. pp. 243–247.

[5] Amiri, B., Hossain, L., Crawford, J. W., 2011. An efficient multiobjective evolutionary algorithm for community detection in social networks. In: Evolutionary Computation (CEC), 2011 IEEE Congress on. pp. 2193–2199.

[6] Amiri, B., Hossain, L., Crawford, J. W., Wigand, R. T., 2013. Community detection in complex networks: Multi–objective enhanced firefly algorithm. Knowledge-Based Systems 46, 1–11.

[7] Amiri, B., Hossain, L., Crowford, J., 2012. A multiobjective hybrid evolutionary algorithm for clustering in social networks. In: Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference companion. pp. 1445–1446.

[8] Arenas, A., Duch, J., Fernández, A., Gómez, S., 2007. Size reduction of complex networks preserving modularity. New Journal of Physics 9 (6), 176.

[9] Arenas, A., Fernández, A., Fortunato, S., Gómez, S., 2008. Motif-based communities in complex networks. Journal of Physics A: Mathematical and Theoretical 41 (22), 224001.

[10] Arenas, A., Fernández, A., Gómez, S., 2008. Analysis of the structure of complex networks at different resolution levels. New Journal of Physics 10 (5), 053039.

[11] Bader, J., Zitzler, E., 2011. Hype: An algorithm for fast hypervolume-based many-objective optimization. Evolutionary Computation 19 (1), 45–76.

[12] Bagrow, J. P., Bollt, E. M., Oct. 2005. A local method for detecting communities. Physical Review E 72 (4), 046108.

[13] Bandyopadhyay, S., Saha, S., Maulik, U., Deb, K., 2008. A simulated annealing-based multiobjective optimization algorithm: AMOSA. IEEE Transaction on Evolutionary Computation 12 (3), 269–283.

[14] Barabási, A.-L., Albert, R., 1999. Emergence of scaling in random networks. Science 286 (5439), 509–512.

[15] Blondel, V. D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 2008 (10), P10008.

[16] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.-U., 2006. Complex networks: Structure and dynamics. Physical Reports 424 (4), 175–308.

[17] Boguna, M., Pastor-Satorras, R., Díaz-Guilera, A., Arenas, A., 2004. Models of social networks based on social distance attachment. Physical Review E 70 (5), 056112.

[18] Bollier, D., Firestone, C. M., 2010. The promise and peril of big data. Aspen Institute, Communications and Society Program Washington, DC, USA.

[19] Brandes, U., Delling, D., Gaertler, M., Görke, R., Hoefer, M., Nikoloski, Z., Wagner, D., 2006. Maximizing modularity is hard. arXiv preprint physics/0608255.

[20] Butun, E., Kaya, M., 2013. A multi-objective genetic algorithm for community discovery. In: Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), 2013 IEEE 7th International Conference on. Vol. 1. pp. 287–292.

[21] Cai, Q., Gong, M., Ma, L., Jiao, L., 2014. A novel clonal selection algorithm for community detection in complex networks. Computational Intelligence 0 (0), 1–24.

[22] Cai, Q., Gong, M., Ma, L., Ruan, S., Yuan, F., Jiao, L., 2015. Greedy discrete particle swarm optimization for large-scale social network clustering. Information Sciences 316, 503–516.

[23] Cai, Q., Gong, M., Shen, B., Ma, L., Jiao, L., 2014. Discrete particle swarm optimization for identifying community structures in signed social networks. Neural Networks 58, 4–13.

[24] Cai, Y., Shi, C., Dong, Y., Ke, Q., Wu, B., 2011. A novel genetic algorithm for overlapping community detection. In: Advanced Data Mining and Applications. pp. 97–108.

[25] Chen, G., Guo, X., 2010. A genetic algorithm based on modularity density for detecting community structure in complex networks. In: Proceedings of the 2010 International Conference on Computational Intelligence and Security. pp. 151–154.

[26] Chen, G., Wang, Y., Wei, J., 2013. A new multiobjective evolutionary algorithm for community detection in dynamic complex networks. Mathematical Problems in Engineering 2013.

[27] Chen, G., Wang, Y., Yang, Y., 2011. Community detection in complex networks using immune clone selection algorithm. International Journal of Digital Content Technology and its Applications 5, 182–189.

[28] Chira, C., Gog, A., 2011. Fitness evaluation for overlapping community detection in complex networks. In: IEEE Congress on Evolutionary Computation. pp. 2200–2206.

[29] Clauset, A., Newman, M. E. J., Moore, C., 2004. Finding community structure in very large networks. Physical Review E 70 (6), 066111.

[30] Coello, C., Pulido, G., Lechuga, M., 2004. Handling multiple objectives with particle swarm optimization. IEEE Transaction on Evolutionary Computation 8 (3), 256–279.

[31] Črepinšek, M., Liu, S.-H., Mernik, M., 2013. Exploration and exploitation in evolutionary algorithms: a survey. ACM Computing Surveys (CSUR) 45 (3), 35.

[32] Danon, L., Díaz-Guilera, A., Arenas, A., 2006. The effect of size heterogeneity on community identification in complex networks. Journal of Statistical Mechanics: Theory and Experiment 2006 (11), P11010.

[33] Danon, L., Díaz-Guilera, A., Duch, J., Arenas, A., 2005. Comparing community structure identification. Journal of Statistical Mechanics: Theory and Experiment 2005 (09), P09008.

[34] Dasgupta, D., Michalewicz, Z., 2013. Evolutionary algorithms in engineering applications. Springer Science & Business Media.

[35] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transaction on Evolutionary Computation 6 (2), 182–197.

[36] Dhillon, I. S., Guan, Y., Kulis, B., 2004. Kernel k-means: spectral clustering and normalized cuts. In: Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 551–556.

[37] Doreian, P., Mrvar, A., 1996. A partitioning approach to structural balance. Social Networks 18 (2), 149–168.

[38] Folino, F., Pizzuti, C., 2010. A multiobjective and evolutionary clustering method for dynamic networks. In: Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on. pp. 256–263.

[39] Folino, F., Pizzuti, C., 2010. Multiobjective evolutionary community detection for dynamic networks. In: Proceedings of the 12th annual conference on Genetic and evolutionary computation. pp. 535–536.

[40] Folino, F., Pizzuti, C., 2014. An evolutionary multiobjective approach for community discovery in dynamic networks. IEEE Transactions on Knowledge and Data Engineering 26 (8), 1838–1852.

[41] Fortunato, S., 2010. Community detection in graphs. Physical Reports 486 (3), 75–174.

[42] Fortunato, S., Barthelemy, M., 2007. Resolution limit in community detection. Proceedings of the National Academy of Sciences 104 (1), 36–41.

[43] Gach, O., Hao, J.-K., 2012. A memetic algorithm for community detection in complex networks. In: Parallel Problem Solving from Nature-PPSN XII. pp. 327–336.

[44] Geem, Z. W., Kim, J. H., Loganathan, G., 2001. A new heuristic optimization algorithm: harmony search. Simulation 76 (2), 60–68.

[45] Girvan, M., Newman, M. E. J., 2002. Community structure in social and biological networks. Proceedings of the National academy of Sciences of the United States of America 99 (12), 7821–7826.

[46] Gog, A., Dumitrescu, D., Hirsbrunner, B., 2007. Community detection in complex networks using collaborative evolutionary algorithms. In: Advances in Artificial Life. pp. 886–894.

[47] Gómez, S., Jensen, P., Arenas, A., 2009. Analysis of community structure in networks of correlated data. Physical Review E 80 (1), 016114.

[48] Gong, M., Cai, Q., Chen, X., Ma, L., 2014. Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition. IEEE Transaction on Evolutionary Computation 18 (1), 82–97.

[49] Gong, M., Cai, Q., Li, Y., Ma, J., 2012. An improved memetic algorithm for community detection in complex networks. In: Proceedings of 2012 IEEE Congress on Evolutionary Computation. pp. 1–8.

[50] Gong, M., Chen, X., Ma, L., Zhang, Q., Jiao, L., 2013. Identification of multiresolution network structures with multi-objective immune algorithm. Applied Soft Computing 13 (4), 1705–1717.

[51] Gong, M., Fu, B., Jiao, L., Du, H., 2011. Memetic algorithm for community detection in networks. Physical Review E 84 (5), 056101.

[52] Gong, M., Hou, T., Fu, B., Jiao, L., 2011. A non-dominated neighbor immune algorithm for community detection in networks. In: Proceedings of the 13th annual conference on Genetic and evolutionary computation. pp. 1627–1634.

[53] Gong, M., Jiao, L., Du, H., Bo, L., 2008. Multiobjective immune algorithm with non-dominated neighbor-based selection. Evolutionary Computation 16 (2), 225–255.

[54] Gong, M., Ma, L., Zhang, Q., Jiao, L., 2012. Community detection in networks by using multiobjective evolutionary algorithm with decomposition. Physica A 391 (15), 4050–4060.

[55] Gong, M., Zhang, L., Ma, J., Jiao, L., 2012. Community detection in dynamic social networks based on multiobjective immune algorithm. Journal of Computer Science and Technology 27 (3), 455–467.

[56] Grossmann, I. E., 2013. Global Optimization in engineering design. Vol. 9. Springer Science & Business Media.

[57] Guimerà, R., Danon, L., Díaz-Guilera, A., 2003. Self-similar community structure in a network of human interactions. Physical Review E 68 (6), 065103.

[58] Guimerà, R., Sales-Pardo, M., Amaral, L. A. N., 2007. Module identification in bipartite and directed networks. Physical Review E 76 (3), 036102.

[59] Guimerò, R., Sales-Pardo, M., Anaral, L. A. N., 2004. Modularity from fluctuations in random graphs and complex networks. Physical Review E 70 (2), 025101.

[60] He, D., Wang, Z., Yang, B., Zhou, C., 2009. Genetic algorithm with ensemble learning for detecting community structure in complex networks. In: Proceedings of the Fourth International Conference on Computer Sciences and Convergence Information Technology. pp. 702–707.

[61] Huang, Q., White, T., Jia, G., Musolesi, M., Turan, N., Tang, K., He, S., Heath, J. K., Yao, X., 2012. Community detection using cooperative co-evolutionary differential evolution. In: Parallel Problem Solving from Nature-PPSN XII. pp. 235–244.

[62] Jia, G., Cai, Z., Musolesi, M., Wang, Y., Tennant, D. A., Weber, R. J., Heath, J. K., He, S., 2012. Community detection in social and biological networks using differential evolution. In: Learning and Intelligent Optimization. pp. 71–85.

[63] Jin, D., He, D., Liu, D., Baquero, C., 2010. Genetic algorithm with local search for community mining in complex networks. In: Tools with Artificial Intelligence (ICTAI), 2010 22nd IEEE International Conference on. Vol. 1. pp. 105–112.

[64] Kang, U., Tsourakakis, C. E., Appel, A. P., Faloutsos, C., Leskovec, J., 2010. Radius plots for mining tera-byte scale graphs: Algorithms, patterns, and observations. In: SIAM International Conference on Data Mining.

[65] Kim, K., McKay, R. I., Moon, B.-R., 2010. Multiobjective evolutionary algorithms for dynamic social network clustering. In: Proceedings of the 12th annual conference on Genetic and evolutionary computation. pp. 1179–1186.

[66] Knowles, J. D., Corne, D. W., 2000. Approxmating the nondominated front using the pareto archived evolution strategy. Evolutionary Computation 8 (2), 149–172.

[67] Kropivnik, S., Mrvar, A., 1996. An analysis of the slovene parliamentary parties network. Developments in Statistics and Methodology, A. Ferligoj, A. Kramberger, eds., 209–216.

[68] Lancichinetti, A., Fortunato, S., Kertész, J., 2009. Detecting the overlapping and hierarchical community structure in complex networks. New Journal of Physics 11 (3), 033015.

[69] Lancichinetti, A., Fortunato, S., Radicchi, F., 2008. Benchmark graphs for testing community detection algorithms. Physical Review E 78 (4), 046110.

[70] Lázár, A., Ábel, D., Vicsek, T., 2010. Modularity measure of networks with overlapping communities. Europhysics Letters 90 (1), 18001.

[71] Lazer, D., Pentland, A., Adamic, L., Aral, S., Barabasi, A. L., Brewer, D., Christakis, N., Contractor, N., Fowler, J., Gutmann, M., Jebara, T., King, G., Macy, M., Roy, D., Van Alstyne, M., 2009. Social science. computational social science. Science 323 (5915), 721–723.

[72] Leicht, E. A., Newman, M. E., 2008. Community structure in directed networks. Physical Review Letters 100 (11), 118703.

[73] Leskovec, J., Krevl, A., Jun. 2014. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`.

[74] Li, D., Xiao, L., Han, Y., Chen, G., Liu, K., 2007. Network thinking and network intelligence. In: Web Intelligence Meets Brain Informatics. Springer, pp. 36–58.

[75] Li, J., Song, Y., 2013. Community detection in complex networks using extended compact genetic algorithm. Soft Computing 17 (6), 925–937.

[76] Li, S., Chen, Y., Du, H., Feldman, M. W., 2010. A genetic algorithm with local search strategy for improved detection of community structure. Complexity 15 (4), 53–60.

[77] Li, X., Gao, C., 2013. A novel community detection algorithm based on clonal selection. Journal of Computational Information Systems 9 (5), 1899–1906.

[78] Li, Y., Liu, G., Lao, S.-y., 2013. Complex network community detection algorithm based on genetic algorithm. In: The 19th International Conference on Industrial Engineering and Engineering Management. pp. 257–267.

[79] Li, Y., Liu, J., Liu, C., 2014. A comparative analysis of evolutionary and memetic algorithms for community detection from signed social networks. Soft Computing 18 (2), 329–348.

[80] Li, Z., Zhang, S., Wang, R.-S., Zhang, X.-S., Chen, L. N., 2008. Quantitative function for community detection. Physical Review E 77 (3), 036109.

[81] Lin, C.-C., Liu, W.-Y., Deng, D.-J., 2013. A genetic algorithm approach for detecting hierarchical and overlapping community structure in dynamic social networks. In: Wireless Communications and Networking Conference (WCNC), 2013 IEEE. pp. 4469–4474.

[82] Lipczak, M., Milios, E., 2009. Agglomerative genetic algorithm for clustering in social networks. In: Proceedings of the 11th Annual conference on Genetic and evolutionary computation. pp. 1243–1250.

[83] Liu, C., Liu, J., Jiang, Z., 2014. A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks. IEEE Transactions on Cybernetics 44 (12), 2274–2287.

[84] Liu, J., Zhong, W., Abbass, H. A., Green, D. G., 2010. Separated and overlapping community detection in complex networks using multiobjective evolutionary algorithms. In: IEEE Congress on Evolutionary Computation. pp. 1–7.

[85] Liu, X., Li, D., Wang, S., Tao, Z., 2007. Effective algorithm for detecting community structure in complex networks based on ga and clustering. In: Computational Science–ICCS 2007. pp. 657–664.

[86] Lohr, S., 2012. The age of big data. New York Times 11.

[87] Lusseau, D., Schneider, K., Boisseau, O. J., Haase, P., Slooten, E., Dawson, S. M., 2003. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology 54 (4), 396–405.

[88] Ma, L., Gong, M., Liu, J., Cai, Q., Jiao, L., 2014. Multi-level learning based memetic algorithm for community detection. Applied Soft Computing 19, 121–133.

[89] Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Byers, A. H., Institute, M. G., 2011. Big data: The next frontier for innovation, competition, and productivity.

[90] Massen, C. P., Doye, J. P., 2005. Identifying communities within energy landscapes. Physical Review E 71 (4), 046101.

[91] Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U., 2002. Network motifs: simple building blocks of complex networks. Science 298 (5594), 824–827.

[92] Mistakidis, E. S., Stavroulakis, G. E., 2013. Nonconvex optimization in mechanics: algorithms, heuristics and engineering applications by the FEM. Vol. 21. Springer Science & Business Media.

[93] Muff, S., Rao, F., Caflisch, A., 2005. Local modularity measure for network clusterizations. Physical Review E 72 (5), 056107.

[94] Newman, M. E. J., 2003. The structure and function of complex networks. SIAM Review 45 (2), 167–256.

[95] Newman, M. E. J., 2006. Finding community structure in networks using the eigenvectors of matrices. Physical Review E 74, 036104.

[96] Newman, M. E. J., 2010. Networks: an introduction. Oxford University Press.

[97] Newman, M. E. J., 2011. Complex systems: A survey. arXiv preprint arXiv:1112.1440.

[98] Newman, M. E. J., Girvan, M., 2004. Finding and evaluating community structure in networks. Physical review E 69 (2), 026113.

[99] Nicosia, V., Mangioni, G., Carchiolo, V., Malgeri, M., 2009. Extending the definition of modularity to directed graphs with overlapping communities. Journal of Statistical Mechanics: Theory and Experiment 2009 (03), P03024.

[100] Oda, K., Kimura, T., Matsuoka, Y., Funahashi, A., Muramatsu, M., Kitano, H., 2004. Molecular interaction map of a macrophage. AfCS Research Reports 2 (14), 1–12.

[101] Oda, K., Matsuoka, Y., Funahashi, A., Kitano, H., 2005. A comprehensive pathway map of epidermal growth factor receptor signaling. Molecular Systems Biology 1 (1), 1–17.

[102] Pizzuti, C., 2008. GA-Net: A genetic algorithm for community detection in social networks. In: Parallel Problem Solving from Nature (PPSN). Vol. 5199. pp. 1081–1090.

[103] Pizzuti, C., 2009. A multi-objective genetic algorithm for community detection in networks. In: Tools with Artificial Intelligence, 2009. ICTAI'09. 21st International Conference on. pp. 379–386.

[104] Pizzuti, C., 2009. Overlapped community detection in complex networks. In: GECCO. Vol. 9. pp. 859–866.

[105] Pizzuti, C., 2012. A multiobjective genetic algorithm to find communities in complex networks. IEEE Transaction on Evolutionary Computation 16 (3), 418–430.

[106] Pons, P., Latapy, M., 2011. Post-processing hierarchical community structures: Quality improvements and multi-scale view. Theoretical Computer Science 412 (8-10), 892–900.

[107] Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D., 2004. Defining and identifying communities in networks. Proceedings of the National academy of Sciences of the United States of America 101 (9), 2658–2663.

[108] Read, K. E., 1954. Cultures of the central highlands, new guinea. Southwestern Journal of Anthropology 10 (1), 1–43.

[109] Rees, B. S., Gallagher, K. B., 2012. Overlapping community detection using a community optimized graph swarm. Social Network Analysis and Mining 2 (4), 405–417.

[110] Reichardt, J., Bornholdt, S., 2006. Statistical mechanics of community detection. Physical Review E 74 (1), 016110.

[111] Rosvall, M., Bergstrom, C. T., 2007. An information-theoretic framework for resolving community structure in complex networks. Proceedings of the National academy of Sciences of the United States of America 104 (18), 7327–7331.

[112] Rosvall, M., Bergstrom, C. T., 2008. Maps of random walks on complex networks reveal community structure. Proceedings of the National academy of Sciences of the United States of America 105 (4), 1118–1123.

[113] Rosvall, M., Bergstrom, C. T., 2008. Maps of random walks on complex networks reveal community structure. Proceedings of the National Academy of Sciences 105 (4), 1118–1123.

[114] Scott, J., 2012. Social network analysis. Sage.

[115] Shang, R., Bai, J., Jiao, L., Jin, C., 2013. Community detection based on modularity and an improved genetic algorithm. Physica A: Statistical Mechanics and its Applications 392 (5), 1215–1231.

[116] Shelokar, P., Quirin, A., Cordón, Ó., 2014. Three-objective subgraph mining using multiobjective evolutionary programming. Journal of Computer and System Sciences 80 (1), 16–26.

[117] Shen, H., Cheng, X., Cai, K., Hu, M.-B., 2009. Detect overlapping and hierarchical community structure in networks. Physica A 388 (8), 1706–1712.

[118] Shen-Orr, S. S., Milo, R., Mangan, S., Alon, U., 2002. Network motifs in the transcriptional regulation network of escherichia coli. Nature Genetics 31 (1), 64–68.

[119] Shi, C., Wang, Y., Wu, B., Zhong, C., 2009. A new genetic algorithm for community detection. In: Complex Sciences. pp. 1298–1309.

[120] Shi, C., Yan, Z., Cai, Y., Wu, B., 2012. Multi-objective community detection in complex networks. Applied Soft Computing 12 (2), 850–859.

[121] Shi, C., Yu, P. S., Cai, Y., Yan, Z., Wu, B., 2011. On selection of objective functions in multi-objective community detection. In: Proceedings of the 20th ACM international conference on Information and knowledge management. ACM, pp. 2301–2304.

[122] Shi, C., Yu, P. S., Yan, Z., Huang, Y., Wang, B., 2014. Comparison and selection of objective functions in multiobjective community detection. Computational Intelligence 30 (3), 562–582.

[123] Shi, Z., Liu, Y., Liang, J., 2009. PSO-based community detection in complex networks. In: Proceedings of the 2nd International Symposium on Knowledge Acquisition and Modeling. Vol. 3. pp. 114–119.

[124] Tasgin, M., Herdagdelen, A., Bingol, H., 2007. Community detection in complex networks using genetic algorithms. arXiv preprint arXiv:0711.0491.

[125] Wang, F. Y., Zeng, D., Carley, K. M., Mao, W. J., 2007. Social computing: From social informatics to social intelligence. IEEE Intelligent Systems 22 (2), 79–83.

[126] Watts, D. J., Strogatz, S. H., 1998. Collective dynamics of 'small-world' networks. Nature 393 (6684), 440–442.

[127] Wolpert, D. H., Macready, W. G., 1997. No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation 1 (1), 67–82.

[128] Xiaodong, D., Cunrui, W., Xiangdong, L., Yanping, L., 2008. Web community detection model using particle swarm optimization. In: IEEE Congress on Evolutionary Computation. pp. 1074–1079.

[129] Xie, J., Kelley, S., Szymanski, B. K., 2013. Overlapping community detection in networks: The state-of-the-art and comparative study. ACM Computing Surveys (CSUR) 45 (4), 43.

[130] Yang, B., Cheung, W. K., Liu, J. M., 2007. Community mining from signed social networks. IEEE Transactions on Knowledge and Data Engineering 19 (10), 1333–1348.

[131] Yang, X.-S., 2013. Multiobjective firefly algorithm for continuous optimization. Engineering with Computers 29 (2), 175–184.

[132] Zachary, W. W., 1977. An information flow model for confict and fission in small groups. Journal of Anthropology Research, 452–473.

[133] Zhang, Q., Li, H., 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Transaction on Evolutionary Computation 11 (6), 712–731.

[134] Zhang, S., Wang, R.-S., Zhang, X.-S., 2007. Identification of overlapping community structure in complex networks using fuzzy c-means clustering. Physica A 374 (1), 483–490.

[135] Zitzler, E., Laumanns, M., Thiele, L., 2002. SPEA2: Improving the strength pareto evolutionary algorithm. In: Proceedings of Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems. pp. 95–100.