

# A Review on Community Detection Methods and Algorithms in Social Networks: Open Trends and Challenges



Ranjana Sikarwar, Shashank Sheshar Singh, and Harish Kumar Shakya

**Abstract** Community detection in social networks is the most widely studied topic of research direction in complex networks. Among other challenging issues of social networks like link prediction, influence maximization and information propagation through diffusion models, community detection has shown substantial growth in popularity gained in the field of technical research. Investigating social structures through clustering and identification of communities has a plethora of applications in the scientific panorama. The identified community structures help in analyzing the common interest, behavior, and psychology of people connected through social ties irrespective of cultural barriers. This paper conducts a detailed review of several evolutionary and swarm intelligence-based algorithms used more recently and widely for finding the formulated community structures in social networks.

**Keywords** Social area networks · Community detection · Genetic algorithms · Metaheuristics · Complex Networks

## 1 Introduction

A complex network is a web containing a collection of nodes connected through edges, for instance, the world wide web, technological networks, biological networks, brain networks, collaboration networks, online social networks, etc. Community detection (CD) problem deals with finding groups of nodes that have strong intracommunity connections and weak intercommunity connectivity. Investigating important nodes in such networks through community detection can provide better insights to analyze the quality of interconnections between different nodes. Community detection problem is considered to be NP-Hard due to the high complexity of the

---

R. Sikarwar · S. S. Singh · H. K. Shakya (✉)

Department of Computer Science and Engineering, Amity University Gwalior, Gwalior, Madhya Pradesh 474005, India

e-mail: [hkshakya@gwa.amity.edu](mailto:hkshakya@gwa.amity.edu)

S. S. Singh

e-mail: [shashank.sheshar@thapar.edu](mailto:shashank.sheshar@thapar.edu)

network structure [1]. It has numerous applications in social networks, healthcare, modeling of epidemic spreading on networks, business, fraud detection, communication networks, biological networks, etc. [1]. Previous studies done on community detection problems found in the literature have worked upon graph partitioning methods, hierarchical clustering approaches, genetic algorithms, and many other evolutionary algorithms and swarm intelligence-based techniques. As shown in Fig. 1 community detection problems can be studied for disjoint communities (no nodes common in 2 or more different communities) and overlapping community detection (nodes common in two or more communities). Many approaches used for detecting community structure in the literature have focused on static communities satisfying the Modularity fitness function for assessing the quality of partitions. Later on, algorithms for detecting dynamic communities concerning the temporal smoothness along with community partitions have also been examined [2, 3]. CD is considered as an optimization problem, so approaches used for community detection have used either single-objective function for optimization or multiobjective functions. Some of the metaheuristic approaches which employs random searching algorithm have been implemented with great efficiency resulting in global optimal solutions which are discussed in this paper as those using genetic algorithms and PSO. These metaheuristic approaches with heuristic operators are used by many researchers for detecting communities [4, 5].

### ***1.1 Classification of Various Types of Algorithms Used in Literature for Community Detection Problem***

Community Detection is an important direction of research in multidisciplinary areas. So many algorithms are classified and proposed in the literature by the scientists and researchers according to the dimension of the work chosen as enlisted in Fig. 1. The algorithms implemented for community detection can be broadly categorized into Graph partitioning, Clustering, and Genetic algorithms for disjoint communities and clique-based algorithms for overlapping community detection.

### ***1.2 Classification of Various Methods of Community Detection Based on Social Network***

As social networks are so vast and widespread according to the applications, the detection of community structure in different types of social networks needs to exploit different algorithms for the analysis. The community detection problem is intended to identify the highly interrelated nodes or vertices in a network within a group which is strongly communicating with each other.

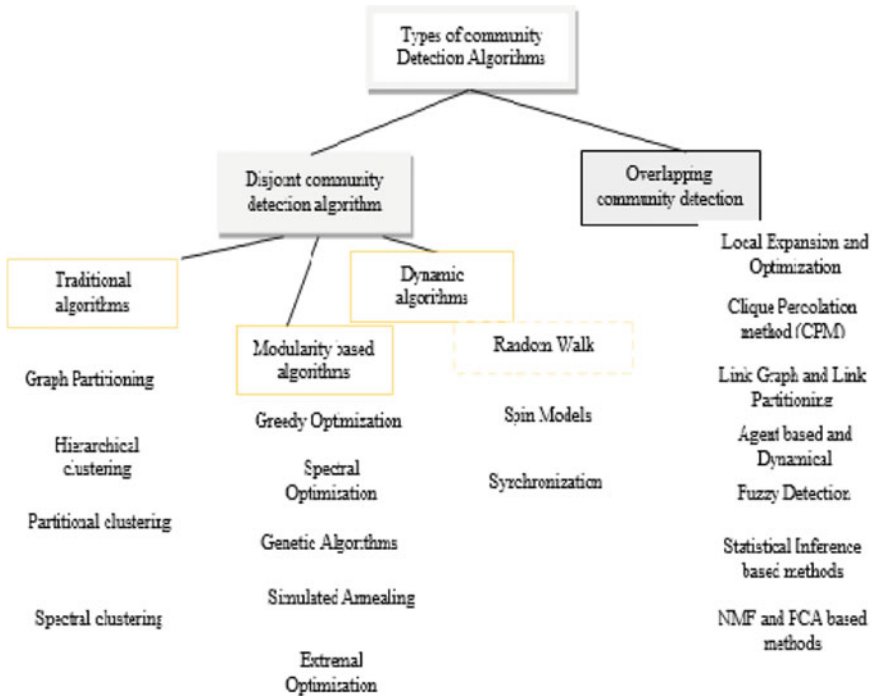


Fig. 1 Classification of Community Detection algorithms [7]

Initially, most of the networks were static but later on with the widespread use of social networking, the networks became more dynamic in nature. The different types of social networks to be investigated in research work are static, signed, positive, dynamic, directed, and heterogeneous [6]. So, the aforementioned task is to:

- Detect communities in static networks
- Community finding in signed networks
- Community detection in dynamic networks
- Detection of community in positive networks
- Detecting communities in heterogeneous networks
- Community detection in a directed network.

The sequence flow of the paper is as follows “**Literature Survey**” discusses the background or the related work, “**Contribution**” describes the novel work done by previous researchers in the latest approaches used, and the “**Conclusion**” the final viewpoint concluding the review done in the paper.

## 2 Literature Survey

Traditional methods used for community detection inspired by clustering methods are those using hierarchical and partitioning methods [8, 9]. But these methods require high computational time and are inefficient to generate optimal solutions in a reasonable time. Also not found efficient for implementation in large-scale networks. The aforementioned issues are addressed very well by evolutionary algorithms using heuristic search. The primary methods used for discovering communities are those by optimizing single-objective function (SO) and using modularity(Q) as an evaluation function which computes intracommunity edges [10]. But using only one objective function may direct the evolving population to form a particular type of community structure or result in a resolution limit problem [11]. Thus, this issue can be addressed by using a multiobjective function (MO) for optimization. MO methods find an optimal solution by establishing a trade-off between different objectives [12]. The concept of Pareto optimality is employed by many evolutionary methods utilizing MO functions. Here in the case of community detection problem MO can find communities with dense intracommunity links and sparse intercommunity connections by optimizing two objective functions simultaneously.

A short glimpse of the previous work done on community detection reveals its societal impact using social networks. The valuable knowledge which can be drawn from studying community structures has led many researchers to investigate the literature behind it. Many comprehensive surveys focus on detecting community structures in multilayer networks [13].

The work presented by (Che et al., 2021) focused on community detection in two modes (bipartite graphs). Their work proposal includes an algorithm which is known as IABC-BN (artificial bee colony algorithm) for detecting communities in bipartite graphs. The experimental results have proved the ABC method to be an excellent algorithm for the discovery of clusters in two-mode graphs. The main contribution of this new algorithm seen is cluster partition for bipartite graphs [14].

Yin et al., (2020) approached the real problem occurring in dynamic networks. The proposed method used DYN-MODPSO for dynamic community detection is an improved evolutionary clustering framework. The multiobjective method is devised for large-scale dynamic networks using PSO. The basic idea is to detect the evolving community structures based on temporal intervals [15].

Reference [16, 17] proposed the use of genetic algorithms with multiobjective criteria to detect communities in complex networks using the algorithm MOGA-Net [18]. His work contributed to the first proposal of using multiobjective GA to discover communities. This algorithm used two objective functions which were optimized to identify partitions in the network structure. The first one uses a community score to evaluate meaningful partitions in the network called communities. A high value of community score corresponds to dense clustering. Another objective function called community fitness is used to analyze the fitness of the nodes confined to a certain group. Further, they extended their work for the application

in dynamic networks using DYNMOGA optimizing modularity and Normalized Mutual Information(NMI) as fitness functions [19].

A particle swarm intelligence-based algorithm called MOPSO-Net was proposed by the authors [20]. Kernel k-means (KKM) and Ratio Cut (RC) are the objective functions to be minimized here. In each iteration, the swarm moved in the direction to achieve the global best solution using the NMI criterion. A Locus-based encoding scheme is used for representation and effective exploration of the solution space. In [21], the authors proposed a many objective(MaOPs) approach for community detection to address the challenges faced by multiobjective methods(using only 2 or 3 quality metrics) in community detection in multi-structural networks. Each quality measure has its specific property for detection thus ignoring other important features to be detected. For example, using only modularity as a quality metric, small communities are left unseen. This issue is addressed by using at least four or larger number of objective functions for identifying community structures.

## ***2.1 Datasets Description***

The datasets often used by many researchers for conducting experimental studies in research work for community detection can be categorized as real datasets (Zachary's karate club, Political blogs, Less Miserables, American college football, Books about US politics, Internet, Coauthorships in Network Science) as well as artificial datasets. They are also known as the benchmark datasets (Lancichinetti et al., Girvan and Newman). These network datasets are in GML format which can be interpreted by many network analysis packages like NetworkX, Cytoscape, etc. (link to download <http://www-personal.umich.edu/~mejn/netdata/>).

## ***2.2 Network Analysis Packages and Tools Used for Identification of Communities***

Some of the popular social network analysis frameworks and tools used for analyzing social network data and graphs are Igraph, Cytoscape, SocNetV, Stanford Network analysis platform (SNAP), Network workbench, NetMiner, NetworkX, Gephi, Graphviz, Neo4j, etc. These social network analysis tools accept network data as GraphML, CSV, GML, and Graphviz file formats and can analyze any type of network data and files. Also, they analyze social networks and outputs important network statistics such as link strength, node density, node strength, visual representation of data, etc. The output file of analyzed network data or graph can be saved or exported in the form of GraphML, GML, BMP, PNG, etc.

### 2.3 Community Detection (CD) as an Optimization Task

In most of the research papers, CD is formulated as an optimization task solved using either a single-objective function or multiobjective function. For instance, reference [12] used two objective functions Negative Ratio Association (NRA) and Ratio cut (RC) (sum of the density of intercommunity links) to be minimized. NRA corresponds to negative RA (sum of internal edge densities of the communities identified). Some of the papers have used modularity as single-objective function [22–25] and many of them used more than one objective function like modularity and NMI (when ground-truth communities are known in advance) [19, 26, 27]. Reference [28] used two objective functions Kernel k-means and Ratio cut with PSO algorithms. Kernel k-means finds solutions with maximum intracommunity edges density and Ratio Cut tries to approach solutions with minimized intercommunity links. The authors [29] have used different variants of objective functions like (Kernel k-means, Ratio Cut, Modularity) as the first variant and (community score, community fitness and modularity) as the second variant with a non-dominated sorting genetic algorithm (NSGA-III). Reference [21] used many objective quality functions such as modularity, NMI, Community Score, Normalized Cut, Conductance, Purity and Rand Index for evaluating the structural properties and quality of the detected communities.

### 2.4 Representation of the Solution

The success of any algorithm depends on the encoding scheme used for representing a solution in the computational search space. Some of the most widely used solution representation schemes used for addressing the community detection problems are discussed below [30].

**Label-based encoding**—Label-based encoding scheme represents the population in the computational space as an integer vector of size (position)  $n$ . Here  $n$  stands for the number (genotype) of nodes. Each location in this vector  $1 \leq i \leq n$ . Suppose if  $k$  is the number of communities in  $\{1, 2, \dots, k\}$ , the  $i$ th position (gene) corresponds to the  $i$ th node. Provided that a genotype has  $k$  number of communities then each gene has a value in the set  $\{1, \dots, k\}$  which is actually the label identifying the community to which the node  $i$  belongs to, thus known as label-based representation. The network in Fig. 2 is partitioned into 3 individual communities as  $\{1, 2, 3\}$ ,  $\{4, 5, 6\}$ ,  $\{7, 8, 9\}$ . Figure 3 below shows the label-representation scheme for Fig. 2.

**Locus-based representation**—This type of solution representation scheme employs an individual  $g$  consisting of  $n$  number of genes  $g_1, g_2, g_3, \dots, g_n$  and each gene  $g_i$  can be mapped to take any adjacent connected node of any node  $i$  as shown in Fig. 4. Thus, in this graph-based representation, a value  $j$  which is assigned to the  $i$ th gene can further be used as a link between node  $i$  and  $j$  in the resultant division of the nodes as communities or partitions of the network. It can be

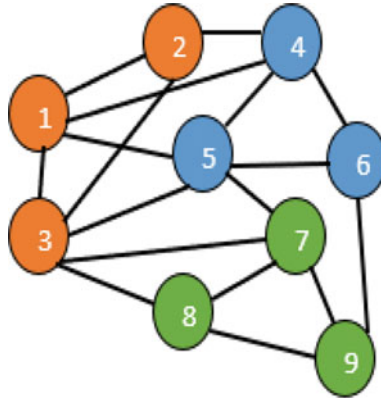


Fig. 2 A network of 9 nodes, 17 edges and 3 communities

<b>position</b>	1	2	3	4	5	6	7	8	9
<b>Label</b>	1	1	1	2	2	2	3	3	3

Fig. 3 Label-based representation of network of Fig. 2

<b>Position</b>	1	2	3	4	5	6	7	8	9
<b>neighbor</b>	2	3	1	5	6	4	8	9	7

Fig. 4 An example of Locus-based representation

concluded that nodes  $i$  and  $j$  belong to the same community. When this representation is decoded all the connected components of the network are identified. The nodes present in one connected component are assigned to one community. The decoding step here helps in finding connected components of the graph. The nodes which form these connected components are assigned to the desired community. This type of representation exhibits redundancy. Label-based representation scheme reduces the complexity of the search space from  $n^n$  (in case of) to  $\prod_{i=1}^n k_i$ ,  $k_i =$  degree of node  $i$ .

**Medoid- based representation** – It’s a prototype-based representation. Here, an  $n$ -dimensional array is used with input elements as the number of communities. For example, from Fig. 2 the partitioned communities are  $\{1,2,3\}$ ,  $\{4,5,6\}$ ,  $\{7,8,9\}$ . Here 1 is the element of the array indicating the prototype of community likewise. This is the medoid-based representation for Fig. 2. These community prototypes coincide with elements of the array. This type of representation scheme shows efficiency for space complexity. However, it has many drawbacks like it is redundant in

nature because medoid can be any element of a particular community and also prior knowledge of  $k$  is required.

Label-based and Locus-based solution representation schemes are the most widely used ones in the literature. The above-described representation schemes refrain a node from becoming a member of more than one community. To overcome this drawback a new representation scheme for overlapping communities was introduced by [31].

## 2.5 Crossover Operator

Although one-point or two-point crossover fits well with label-based representation still it has two main drawbacks. The first drawback is that a community may contain disconnected subgroups of the node means nodes having no connections are placed in the same community. To allay this problem, the idea of one-way crossover was proposed by [32]. But it produces only one child from two parents. Another drawback observed is that the children doesn't receive the genetic characteristics of the parent nodes fully. This issue was encountered by [33]. However, according to the author's observation and view point this crossover enhances the global search fitness of their method but they didn't throw any light on the increase in computational time. While medoid-based representation works with one-point crossover and standard uniform crossover is exploited by locus-based representation. Standard uniform crossover is used by the locus-based representation scheme in which the off springs fully inherit the genetic properties of their parents [16].

## 3 Contribution

### Evolutionary Algorithms (EA)

The category of EA algorithms particularly the genetic algorithms (GA) work on the concept of random population generation. These individuals in the population refer to chromosomes in the case of GA. The structure of chromosomes is organized according to the type of problem GA addresses. An objective function quantifies the quality of chromosomes in the population. This objective function evaluates the fitness value of the chromosomes and a percentage of high-fitness valued chromosomes are selected for the next iteration. Crossover and mutation operators on the chromosomes generate an improved population of individuals until the termination condition is achieved. An optimal solution is produced at the last step of the algorithm. These are the widely embraced techniques to solve NP-complete problems related to optimization due to their robustness in contrast to other traditional methods. GAs that can use different representation schemes are good for solving dynamic problems [34].



### Particle swarm optimization (PSO)

PSO is used as a population-based stochastic searching algorithm for the community detection problem. It solves the optimization problems simulating the bird flocking behavior which are randomly searching for food in an area. The exact location of the food particle is not known to them. So, they apply the strategy of following those birds which are in close proximity to the food particle. To address any problem, a population or swarm of particles(solutions) is randomly generated initially. These particles search for the optimal solutions in the state space of possible solutions by updating generations. Each particle is associated with a position vector ( $X_i$ ) and a velocity vector ( $V_i$ ). At each iteration, every particle is attracted towards its personal best position ( $Pbest_{id}$ ) and best position of all particles ( $Gbest_{id}$ ) while moving randomly at the same time. [35, 36].

$$v_i^{t+1} = v_i^t + c_1z_1(Gbest_{id} - x_i^t) + c_2z_2(Pbest_{id} - x_i^t) \quad (1)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (2)$$

where,  $c_1, c_2$  stands for acceleration parameters known as cognitive and social components  $r_1, r_2$  are random numbers between  $\{0,1\}$ .

### Bat Algorithm (BA)

Bat Algorithm is also a metaheuristic algorithm that mimics the prey hunting behavior of bats using an echolocation strategy to sense distance and velocity with static variations and loudness frequency. Bat algorithm address the CD as an optimization task where each 'bat' represents an individual in the population. It adapts the features of both particle swarm optimization (PSO) and simulated annealing. These combined features make Bat algorithm an outstanding one to achieve global search capability and strong convergence capability. BA simulates the emission rates, loudness and frequency variations of bats when they go for prey hunting. Bats transform their wavelength according to pulse frequency variations to locate the target. The updation rules for position and velocity for BA are similar to those of PSO algorithms. Continuous process of frequency and loudness adjustment maintains a balance between the intensification and diversification operations of the algorithm. BA overcome the drawback of PSO by generating a random solution using random flight behavior to avoid sinking into local optimum [37, 38, 27, 37]. The main equation for updating the bat location based on frequency and velocity is shown below:

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \quad (3)$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x^*)f_i \quad (4)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (5)$$

where,  $f_{\min}$  is minimum frequency,  $f_{\max}$  is a maximum frequency,  $\beta$  is a random number which takes a value between 0 and 1.  $x_i^t$  is the current location of the  $i$ th bat,  $x_i^{t-1}$  is the previous location of the  $i$ th bat,  $v_i^t$  is the new velocity,  $v_i^{t-1}$  is the previous velocity of the  $i$ th bat.

### Differential Evolution (DE)

DE is a new population-based stochastic search evolutionary algorithm. As compared to the traditional GA algorithm, DE algorithm exhibits some merits: fast convergence, identifies optimized solutions regardless of initial parameters, requiring only a few control parameters. DE initiates the search procedure with a population of NP individuals randomly sampled where each individual signifies the target vector is selected from the population used to generate the mutant vector using the mutation operator. DE's performance depends on the setting of control parameters like the size of the population, crossover, scale factor and the mutation scheme. These parameters should be set properly for the efficient solution of the problem. The mutation scheme of the DE algorithm exploits the genetic information of several individuals to utilize the distributed population characteristics and improve the search ability [39, 40]. Some mutation strategies often used with DE are: DE/rand/1 (known as classical mutation scheme in DE), DE/best/1, DE/best/2, and DE/rand-to-best/1. DE/rand/1 is the most popular mutation strategy used with DE in community detection is as follows:

$$v_{i,m} = x_{r1,m} + F * (x_{r2,m} - x_{r3,m}) \quad (6)$$

where  $i = \{1, 2, \dots, NP\}$ ,  $r1, r2$  and  $r3$  are randomly selected integer values from  $1, 2, \dots, NP$ , satisfying  $r1 \neq r2 \neq r3 \neq i$ , scaling factor  $F$  is a real number between  $\{0,1\}$ .

### Memetic Algorithms (MAs)

Memetic Algorithms (MA) are considered as the hybridization of previous GA based evolutionary algorithms. It is also a population-based approach with separate individual learning or optimization intersperse with the recombination of high-quality solutions. They consider evolution as a baseline principle of working. It relies on the local search improvement procedures for problem search thus reducing the premature convergence. The word "memetic" is stirred by the Dawkin's notion of the word 'meme', an element of social development resulting in local refinement [41]. The meme used in MAs represents a distinct learning procedure which can exhibit local refinements. MA uses the combination of GA and local search procedure to solve the optimization problem. MA outperforms existing genetic algorithms for specific applications of community detection [42–44].

### Ant Colony Optimization

It is a metaheuristic optimization algorithm, basically a simulation of the ants foraging behavior independently communicating with each other through pheromone. It is also considered as a distributed multi-agent system where the search for food begins from different locations at the same time [45]. The population of ants construct solutions iteratively by finding the shortest path using pheromone and leaving the heuristic information behind them by crossing the paths. ACO algorithms are used in finding the community structure in the network. The positive feedback mechanism is used to find optimal solutions. The quality of the solution achieved by each artificial ant is assessed by its modularity. The probability of selecting a route by the ants from vertex  $x$  to  $y$  is given by the following formula below:

$$p_{xy} = \frac{\mu_{xy}^\alpha h_{xy}^\beta}{\sum_{x,y=1}^n \mu_{xy}^\alpha h_{xy}^\beta} \tag{7}$$

where,

- $\mu_{xy}$  is the pheromone concentration of the path between  $x, y$
- $h_{xy}$  is a heuristic function with a likelihood to select an edge from point  $x$  to  $y$ .
- $\alpha, \beta$  determines relative influence of trail information and visibility.

### Firefly Algorithm

It's a population-grounded algorithm where each firefly represents a feasible solution. This algorithm imitates the flashing patterns and activities of the fireflies [46]. The main principle for the sparkle of fireflies is to attract other fireflies. This algorithm was proposed with a few assumptions like a firefly is attracted towards another firefly according to the brightness intensity. With the increase in the distance the brightness of the firefly decreases. The movement of fireflies towards the brightest firefly is to achieve a global optimal solution. FA algorithm depends on the parameters like random movement and attractiveness as performance measures. Community detection problems can be solved using the FA algorithm as an optimization algorithm by maximizing the modularity function. The main update formula [47] for any pair of two fireflies  $x_i$  and  $x_j$  is

$$x_i^{t+1} = x_i^t + \beta_0^{-\gamma r_{ij}^2} (x_j^t - x_i^t) + \alpha_t e_i^t \tag{8}$$

where,  $x_i^t$  represents the  $i^{\text{th}}$  solution (firefly) at iteration  $t$ .

$\beta_0$  is brightness at source.

A solution  $x_i$  will be attracted towards a brighter firefly  $x_j$ , means  $x_i$  moves towards  $x_j$ ,  $\alpha$  randomization parameter,  $e_i^t$  vector of random variables.

## 4 Conclusion

The aim of this comprehensive review is to encompass various evolutionary and swarm intelligent-based algorithms for community detection that have encouraged a flurry of research. The widespread use of the aforementioned algorithms has shown an outstanding performance in detecting communities in static, dynamic, complex or multi-structural networks. Classification of different types of methods and algorithms used in addressing the community detection problem on the basis of social networks is also discussed here. The discussion of evolutionary and nature-inspired (NIA) algorithms based on single-objective or multiobjective has also been covered along with the most commonly adopted evaluation metrics like Modularity and NMI. A detailed description of the most widely used EA and NIA algorithms is statistically broken down and summarized in the tabular form according to the common key components used. These statistics provides a direction to the readers and researchers to select the characteristics of the algorithms like population initialization methods, perturbation operators and types of objective functions. It is observed that most of the research papers have shown a research gap for community detection in overlapping communities, multilayer networks and large-scale networks, implementing the algorithms independent of the increasing network size and substantial improvement in speed and accuracy.

## References

1. Javed MA, Younis MS, Latif S, Qadir J, Baig A (2018) Community detection in networks: a multidisciplinary review. *J Netw Comput Appl* 108:87–111. <https://doi.org/10.1016/j.jnca.2018.02.011>
2. Zeng X, Member S, Wang W, Chen C, Yen GG (2019) A consensus community-based particle swarm optimization for dynamic community detection. *IEEE Trans Cybern* 1–12. <https://doi.org/10.1109/TCYB.2019.2938895>
3. Messaoudi I, Kamel N (2019) A multi-objective bat algorithm for community detection on dynamic social networks. *Appl Intell* 49(6):2119–2136. <https://doi.org/10.1007/s10489-018-1386-9>
4. Hussain K, Mohd Salleh MN, Cheng S, Shi Y (2019) Metaheuristic research: a comprehensive survey. *Artif Intell Rev* 52(4), 2191–2233. <https://doi.org/10.1007/s10462-017-9605-z>
5. Dokeroglu T, Sevinc E, Kucukyilmaz T, Cosar A (2019) A survey on new generation metaheuristic algorithms. *Comput Ind Eng* 137. <https://doi.org/10.1016/j.cie.2019.106040>
6. Pourkazemi M, Keyvanpour M (2013) A survey on community detection methods based on the nature of social networks. In: *International conference on computer and knowledge engineering ICCKE 2013*, no. Ickce, pp 114–120. <https://doi.org/10.1109/ICCKE.2013.6682855>
7. Fortunato S (2010) Community detection in graphs. *Phys Rep* 486(3–5):75–174. <https://doi.org/10.1016/j.physrep.2009.11.002>
8. Lu X, Kuzmin K, Chen M, Szymanski BK (2018) Adaptive modularity maximization via edge weighting scheme. *Inf Sci (Ny)* 424:55–68. <https://doi.org/10.1016/j.ins.2017.09.063>
9. Wu W, Kwong S, Zhou Y, Jia Y, Gao W (2018) Nonnegative matrix factorization with mixed hypergraph regularization for community detection. *Inf Sci (Ny)* 435:263–281. <https://doi.org/10.1016/j.ins.2018.01.008>

10. Newman MEJ, Girvan M (2004) Finding and evaluating community structure in networks. *Phys Rev E - Stat Nonlinear Soft Matter Phys* 69(22), 1–15. <https://doi.org/10.1103/PhysRevE.69.026113>
11. Fortunato S, Barthélemy M (2007) Resolution limit in community detection. *Proc Natl Acad Sci USA* 104(1):36–41. <https://doi.org/10.1073/pnas.0605965104>
12. Shang J, Li Y, Sun Y, Li F, Zhang Y, Liu J (2021) SS symmetry MOPIO : a multi-objective pigeon-inspired optimization, pp 1–16
13. Huang X, Chen D, Ren T, Wang D (2020) A survey of community detection methods in multilayer networks. Springer, US
14. Che S, Yang W, Wang W (2021) An improved artificial bee colony algorithm for community detection in bipartite networks. *IEEE Access* 9:10025–10040. <https://doi.org/10.1109/ACCESS.2021.3050752>
15. Yin Y et al (2020) Multi-objective evolutionary clustering for large-scale dynamic community detection. <https://doi.org/10.1016/j.ins.2020.11.025>
16. Pizzuti C (2008) GA-Net: a genetic algorithm for community detection in social networks. *Lecture notes in computer science (including Lecture notes in artificial intelligence and lecture notes in bioinformatics)*, vol 5199 LNCS, pp 1081–1090. [https://doi.org/10.1007/978-3-540-87700-4\\_107](https://doi.org/10.1007/978-3-540-87700-4_107)
17. Pizzuti C (2012) A multiobjective genetic algorithm to find communities in complex networks. *IEEE Trans Evol Comput* 16(3):418–430. <https://doi.org/10.1109/TEVC.2011.2161090>
18. Pizzuti C (2009) A multi-objective genetic algorithm for community detection in networks. In: *Proceedings of international conference on tools with artificial intelligence ICTAI*, no. October 2014, pp 379–386. <https://doi.org/10.1109/ICTAI.2009.58>
19. Folino F, Pizzuti C (2014) An evolutionary multiobjective approach for community discovery in dynamic networks. *IEEE Trans Knowl Data Eng* 26(8):1838–1852. <https://doi.org/10.1109/TKDE.2013.131>
20. Rahimi S, Abdollahpouri A, Moradi P (2017) “SC,” and evolutionary computation BASE DATA. <https://doi.org/10.1016/j.swevo.2017.10.009>
21. Tahmasebi S, Moradi P, Ghodsi S, Abdollahpouri A (2019) An ideal point based many-objective optimization for community detection of complex networks. *Inf Sci (Ny)* 502:125–145. <https://doi.org/10.1016/j.ins.2019.06.010>
22. Guerrero M, Montoya FG, Baños R, Alcalde A, Gil C (2017) Adaptive community detection in complex networks using genetic algorithms. *Neurocomputing* 266:101–113. <https://doi.org/10.1016/j.neucom.2017.05.029>
23. Moradi M, Parsa S (2019) An evolutionary method for community detection using a novel local search strategy. *Phys A*. <https://doi.org/10.1016/j.physa.2019.01.133>
24. Guo X, Su J, Zhou H, Liu C, Cao J, Li L (2019) Community detection based on genetic algorithm using local structural similarity. *IEEE Access* 7:134583–134600. <https://doi.org/10.1109/ACCESS.2019.2939864>
25. Li C, Wang R, Li J, and Fei L, *Face detection based on YOLOv3*, vol. 1031 AISC. 2020
26. Ghaffaripour Z, Abdollahpouri A, Moradi P (2016) A multi-objective genetic algorithm for community detection in weighted networks. In: *2016 8th international conference on information and knowledge technology IKT 2016*, pp 193–199. <https://doi.org/10.1109/IKT.2016.7777766>
27. Doush IA, Alrashdan WB, Al-Betar MA, Awadallah MA (2020) Community detection in complex networks using multi-objective bat algorithm. *Int J Math Model Numer Optim* 10(2):123–140. <https://doi.org/10.1504/IJMMNO.2020.106529>
28. Gong M, Cai Q, Chen X, Ma L (2014) Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition. *IEEE Trans Evol Comput* 18(1):82–97. <https://doi.org/10.1109/TEVC.2013.2260862>
29. Shaik T, Ravi V, Deb K (2021) Evolutionary multi - objective optimization algorithm for community detection in complex social networks. *SN Comput Sci*. <https://doi.org/10.1007/s42979-020-00382-x>

30. Hruschka ER, Campello RJGB, Freitas AA, de Carvalho ACPLF (2009) A survey of evolutionary algorithms for clustering. *IEEE Trans Syst Man Cybern Part C Appl Rev* 39(2):133–155. <https://doi.org/10.1109/TSMCC.2008.2007252>
31. Liu J, Zhong W, Abbass HA, Green DG (2010) Separated and overlapping community detection in complex networks using multiobjective Evolutionary Algorithms. In: 2010 IEEE world congress on computational intelligence WCCI 2010 - 2010 IEEE congress on evolutionary computation CEC 2010. <https://doi.org/10.1109/CEC.2010.5586522>
32. Tasgin M, Bingol H (2006) Community detection in complex networks using genetic algorithm, pp 1–6. <http://arxiv.org/abs/cond-mat/0604419>
33. He D, Wang Z, Yang B, Zhou C (2009) Genetic algorithm with ensemble learning for detecting community structure in complex networks. In: ICCIT 2009 - 4th international conference on computer sciences and Convergence Information Technology, pp 702–707. <https://doi.org/10.1109/ICCIT.2009.189>
34. Abduljabbar DA, Hashim SZM, Sallehuddin R (2020) Nature-inspired optimization algorithms for community detection in complex networks: a review and future trends. *Telecommun Syst* 74(2):225–252. <https://doi.org/10.1007/s11235-019-00636-x>
35. Cai Q, Gong M, Ma L, Ruan S, Yuan F, Jiao L (2015) Greedy discrete particle swarm optimization for large-scale social network clustering. *Inf Sci (Ny)* 316:503–516. <https://doi.org/10.1016/j.ins.2014.09.041>
36. Gao C, Chen Z, Li X, Tian Z, Li S (2018) Multiobjective discrete particle swarm optimization for community detection in dynamic networks. <https://doi.org/10.1209/0295-5075/122/28001>
37. Hassan EA, Hafez AI, Hassanien AE, Fahmy AA (2015) A discrete bat algorithm for the community detection problem. *Lecture notes in computer science (including Lecture notes in artificial intelligence and lecture notes in bioinformatics)*, vol 9121, pp 188–199. [https://doi.org/10.1007/978-3-319-19644-2\\_16](https://doi.org/10.1007/978-3-319-19644-2_16)
38. Chunyu W, Yun P (2015) Discrete bat algorithm and application in community detection. *Open Cybern Syst J* 9(1):967–972. <https://doi.org/10.2174/1874110X01509010967>
39. Jia G et al. (2012) Community detection in social and biological networks using differential evolution. *Lecture notes in computer science (including Lecture notes in artificial intelligence and lecture notes in bioinformatics)*, vol 7219. LNCS, pp 71–85. [https://doi.org/10.1007/978-3-642-34413-8\\_6](https://doi.org/10.1007/978-3-642-34413-8_6)
40. Sun H, Ma S, Wang Z (2018) A community detection algorithm using differential evolution. In: 2017 3rd IEEE international conference on computer and communications ICC 2017, pp 1515–1519. <https://doi.org/10.1109/CompComm.2017.8322793>
41. Mu CH, Xie J, Liu Y, Chen F, Liu Y, Jiao LC (2015) Memetic algorithm with simulated annealing strategy and tightness greedy optimization for community detection in networks. *Appl Soft Comput J* 34:485–501. <https://doi.org/10.1016/j.asoc.2015.05.034>
42. Žalik KR, Žalik B (2018) Memetic algorithm using node entropy and partition entropy for community detection in networks. *Inf Sci (Ny)* 445–446:38–49. <https://doi.org/10.1016/j.ins.2018.02.063>
43. Wang S, Gong M, Liu W, Wu Y (2020) Preventing epidemic spreading in networks by community detection and memetic algorithm. *Appl Soft Comput J* 89:106118. <https://doi.org/10.1016/j.asoc.2020.106118>
44. Haque MN, Mathieson L, Moscato P (2018) A memetic algorithm for community detection by maximising the connected cohesion. In: 2017 IEEE symposium series on computational intelligence SSCI 2017 - Proceedings, pp 1–8. <https://doi.org/10.1109/SSCI.2017.8285404>
45. Chen B, Chen L, Chen Y (2012) Detecting community structure in networks based on ant colony optimization. In: International conference on information and knowledge engineering, pp 247–253
46. Jaradat AS, Hamad SB (2018) Community structure detection using firefly algorithm. *Int J Appl Metaheuristic Comput* 9(4):52–70. <https://doi.org/10.4018/IJAMC.2018100103>
47. Del Ser J, Lobo JL, Villar-Rodríguez E, Bilbao MN, Perfecto C (2016) Community detection in graphs based on surprise maximization using firefly heuristics. In: 2016 IEEE congress on evolutionary computation CEC, pp 2233–2239. <https://doi.org/10.1109/CEC.2016.7744064>