AMSE JOURNALS-AMSE IIETA publication-2017-Series: Advances A; Vol. 54; N°3; pp 384-406

Submitted Sept. 14, 2017; Revised Dec. 26, 2017; Accepted Jan. 04, 2018

https://doi.org/10.18280/ama_a.540301

A New Approach to Community Detection in Complex Networks by Using Memetic Algorithms

*Ali Ghorbanian, **Mehdi Neyestani

* Department of Industrial Engineering, Esfarayen University of technology, Iran, North Khorasan 96619-98195 (ali.ghorbanian83@gmail.com)

** Department of Electrical Engineering, Esfarayen University of technology, North Khorasan 96619-98195 (m.Neyestani@esfarayen.ac.ir)

Abstract

Nowadays, networks are widely used to show the structure and type of relationship in various fields of science including social sciences, engineering, and social network. Investigating the structure of these networks has always been of interest to many people. To this end, the present study tries to identify these networks as interconnected communities, while each community might have unique properties. In this regard various methods and algorithms with different evaluation criteria are used, with the evolutionary algorithm having the highest share. The present research attempts to introduce a relatively efficient method for community detection through introduction of a new evaluation criterion based on the modularity density and giving weight to the relationships between nodes in the unweighted networks. For this purpose, at first we propose a new modularity based on difference of weight of nodes in a same community with other community also for this we propose a method for calculation weight between nodes in a same community named α_ij and between nodes in different community named β_ij. And employing this new modularity as the evaluation criteria by using Memetic Algorithm(MA). The proposed algorithms are evaluated in very complex artificial and real networks and the results were analyzed and were compared to other algorithms. For accuracy of the algorithm the number of communities is identified also Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) criteria are used to evaluate the algorithm. The obtained results indicated that

MA based on new density is effective and efficient at detecting the community structure in complex real and artificial networks.

Keywords

Complex networks, Modularity density, Memetic algorithm, Unweighted networks

1. Introduction

In recent years, several evidences have shown that many complex systems can be modeled as complex networks. So it is possible to use network theory to study these models [1-3]. Many of these studies have shown that most of these systems have a group of nodes in the form of a community, and these nodes have several inner edges compared to the edges between groups [1-6]. In the network, communities have common characteristics. For example, in social networks, communities often have the same culture, interests, language or jobs [1-3]. Figure 1 shows the meaning of a community in a network.

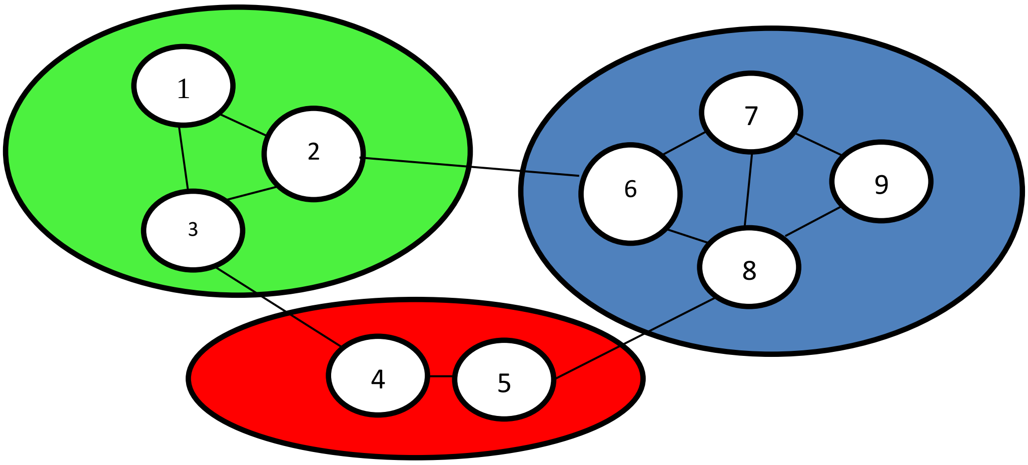


Figure 1. Showing community in a network

Networks can show the link between people, organizations, etc. [7]. Social networks in recent years have played important role in many issues; [8] such as social relations, marketing, and many challenging issues, security [9] [10], visualization [11], [12] or analyzing social networks studies [11].

In recent years, many methods have been proposed for community detection in complex networks. But it can be claimed that all of them have partly found the communities or in the case of large networks they mostly face difficulties.

Community finding issues in the networks are NP-hard and the traditional optimization methods cannot solve them [13]. Therefore, to solve these problems heuristic and meta-heuristic algorithms should be used.

There are various methods such as Newman _ Girvan, joint community detection method, DBSCAN algorithm, BGLL method, the use of evolutionary algorithm, fundamental categorization, the use of high optimization and the minimum spanning tree, and random walks. One of the methods that recently have received considerable attention is local neighborhood.

A considerable number of algorithms have used local neighborhood for community detection [14-16]. The DBSCAN algorithm is employed by using the concept of distance between nodes to identify adjacent and central nodes and community detection [17]. DBSCAN algorithm has been widely used in recent years for identifying HUB in a complex network [18].

A brief look at literature shows that evolutionary algorithms have the highest contribution in solving community detection problems. Several articles have employed evolutionary algorithms in order to detect communities in different networks [19-21]. The genetic algorithm is the most important and widely used among them. The grouping genetic algorithm is also sometimes used in recent years. Given that the proposed algorithm in this research is also a kind of evolutionary algorithms, a brief review of the use of these algorithms in the community detection in networks is presented.

Evolutionary algorithms represent one of the most widely used methods in the optimization of NP-hard problems, which are used in a variety of issues.

Bui et al. [22] for the first time used a genetic algorithm for community detection. Tasgin et al. for the first time used the concept of modularity in genetic algorithms for community detection by using genetic algorithms [23].

GA-Net algorithm is presented by Pizzuti [24]. It basically reduces the number of searches for finding the optimal solution in a network. Shang et al [25] used improved genetic algorithm (IMGA) based on modularity for community detection but in this algorithm, the number of community structures is used as a prior information and this is a weakness for proposed algorithm. Zhao et al [26] proposed the irregular cellular learning automata (ICLA) to reveal optimal community in complex networks. Rezapoor Mirsaleh et al [27] introduced and used a Michigan memetic algorithm; called MLAMA-Net to solve community detection problem. Density plays crucial role in most algorithms in some way or another. Perhaps the failure of many of these algorithms in complex networks for community detection is caused by this issue.

This research attempts to provide a new definition of density for a social network. Then, it is employed as an evaluation function in Memetic Algorithms (MA) for community detection in complex networks. Therefore, the study provides the already existing definition of density and then provides a new definition for density.

2. Modularity density

Given a network $\mathrm{G} = (\mathrm{V}, \mathrm{E})$, that V is the node set, E is the edge set, and A is the adjacency matrix of G. If V_1 and V_2 are two disjoint subsets of V and $\mathrm{L}(\mathrm{V}_1, \mathrm{V}_2) = \sum_{\mathrm{i}\in\mathrm{V}_1,\mathrm{j}\in\mathrm{V}_2} \mathrm{A}_{\mathrm{ij}}$, $\mathrm{L}(\mathrm{V}_1, \mathrm{V}_1) = \sum_{\mathrm{i}\in\mathrm{V}_1,\mathrm{j}\in\mathrm{V}_1} \mathrm{A}_{\mathrm{ij}}$ and $\mathrm{L}(\mathrm{V}_1, \overline{\mathrm{V}}_1) = \sum_{\mathrm{i}\in\mathrm{V}_1,\mathrm{j}\in\overline{\mathrm{V}}_1} \mathrm{A}_{\mathrm{ij}}$, where $\overline{\mathrm{V}}_1 = \mathrm{V} - \mathrm{V}_1$. Given the partition of a network G, $\mathrm{G}(\mathrm{V}_1, \mathrm{E}_1), \ldots., \mathrm{G}(\mathrm{V}_\mathrm{m}, \mathrm{E}_\mathrm{m})$, where V_i and E $_\mathrm{i}$ are, respectively, the node set and the edge set of G_i for i $= 1, \ldots, \mathrm{m}$, the modularity Q is defined as equation 1:

$$Q = \sum_{i=1}^{m} \left[\frac{L(V_i, V_i)}{L(V, V)} - \left(\frac{L(V_i, V)}{L(V, V)} \right)^2 \right] \tag{1}$$

Modularity maximization algorithms seem to be effective methods for community detection in networks. However, Fortunato and Barthélemy [28], showed the serious resolution limits of this method and claimed that the size of a detected community depends on the size of the whole network. This is mainly because the modularity measure does not contain information on the number of nodes in a community and the choice of partition is highly sensitive to the total number of edges in the network. Thus li et al. [29] introduce a measure D, which is related to the density of sub graphs to overcome this problem. The modularity density can be calculated as follows:

$$D = \sum_{i=1}^{m} [d_{in}(G_i) - d_{out}(G_i)] = \sum_{i=1}^{m} \left[\frac{L(V_i, V_i) - L(V_i, \bar{V})}{|V_i|} \right] \tag{2}$$

Where $\mathrm{d}_{\mathrm{in}}(\mathrm{G}_\mathrm{i})$ is the average inner degree of the subgraph G_i, which is equal to twice the number of edges in subgraph G_i, divided by the number of nodes in V_i. $\mathrm{d}_{\mathrm{out}}(\mathrm{G}_\mathrm{i})$, is the average outer degree of subgraph G_i, which is equal to the number of edges with one node in Vi and the other node outside Vi divided by the number of nodes in V_i. The larger the value of D, the more accurate a partition is. So the community-detection problem can be viewed as a problem of finding a partition of a network such that its modularity density D is maximized [29].

Li et al. proved that maximizing modularity density can resolve the resolution limit of modularity. They also showed that Modularity density does not divide a clique into two parts.

The search for optimal modularity density D is a NP-hard problem due to the fact that the space of possible partitions grows faster than any power of system size. In this paper we will propose a Memetic Algorithm (MA) to maximize the modularity density.

2.1 New Density

As mentioned in the above definition, to obtain density in a complex network, first of all the density for each of the communities is calculated and then the values are added together to get the total amount of density in a network. Moreover, according to the definition to obtain the amount of density in a community it is just needed to subtract the number of relationships (edge) within a community from the number of external relationships and divide them by the number of nodes within the community. But the biggest problem with this definition of the density is that the weight of all relationships within and outside the community is considered to be equal. In fact, to calculate the density the quantity is not the only important things. However, the new approach attempts to consider a weight for each of the nodes within the community and outside the community. By using the obtained weights, the new density is calculated and through this definition one of the problems of the previous definition is resolved.

To this end, two definitions for weighting each of the relationships within a network are employed. Of course, the definition of the weight varies depending on its position inside or outside the community.

Definition 1:

Considering v_i node, the set of nodes that are connected to this node are shown by$C(v_i)$. It should be noted that node i too is placed in this set. Also $|C(v_i)|$ shows the number of members inside the set. Similarly, $C(v_i)mn$ displays a set of nodes that are connected to the ith node and are a member of one of the two n and m communities. $|C(v_i)mn|$ shows the number of members inside the set.

Definition 2:

If the two groups of v_i and v_j are both within a community the proximity between them or, in fact, their weight are displayed as α_{ij} and defined as follows.

$$\alpha_{ij} = \frac{|C(v_i) \cap C(v_j)|}{\sqrt{(|C(v_i)|, |C(v_j)|)}} \tag{3}$$

Definition 3

If Com represents mth community and con shows nth community, and each of the communities has a number of nodes, then β_{ij} which is i ∈ com and j ∈ con are defined as follows:

$$\beta_{ij} = \frac{|C(v_i)mn \cap C(v_j)mn|}{\sqrt{(|C(v_i)mn|, |C(v_j)mn|)}} \tag{4}$$

Considering the above definitions, in a given network that is divided into a number of communities for each relationship between nodes, we can define the amount ofα_{ij}. Also β_{ij} for each node that connects two communities is defined, although there is no need to define the amount of α_{ij} for these nodes. For example, consider network A with 11 nodes that are hypothetically categorized into three communities. It is displayed in Figure 2. The numbers on the figure show the calculated values of α and β with respect to the above definition.

Now, according to the above definitions, as well as the definition of density function, a new definition for density function is provided.

If the hypothetical network A is composed of K communities, then the new density that is displayed as ND is show by the relationship number.

$$ND = \sum_{t=1}^{k} \frac{[nd_{in}(co_t) - nd_{out}(co_t)]}{k} \tag{5}$$

$$nd_{in}(co_t) = \sum_{i,j \in t} \alpha_{ij} \tag{6}$$

$$nd_{out}(co_t) = \sum_{i \in t, j \in \bar{t}} \beta_{ij} \tag{7}$$

According to the relationship defined in the formula, the more the amount of weight within the communities and the less the relations between communities are, the more the density value is.

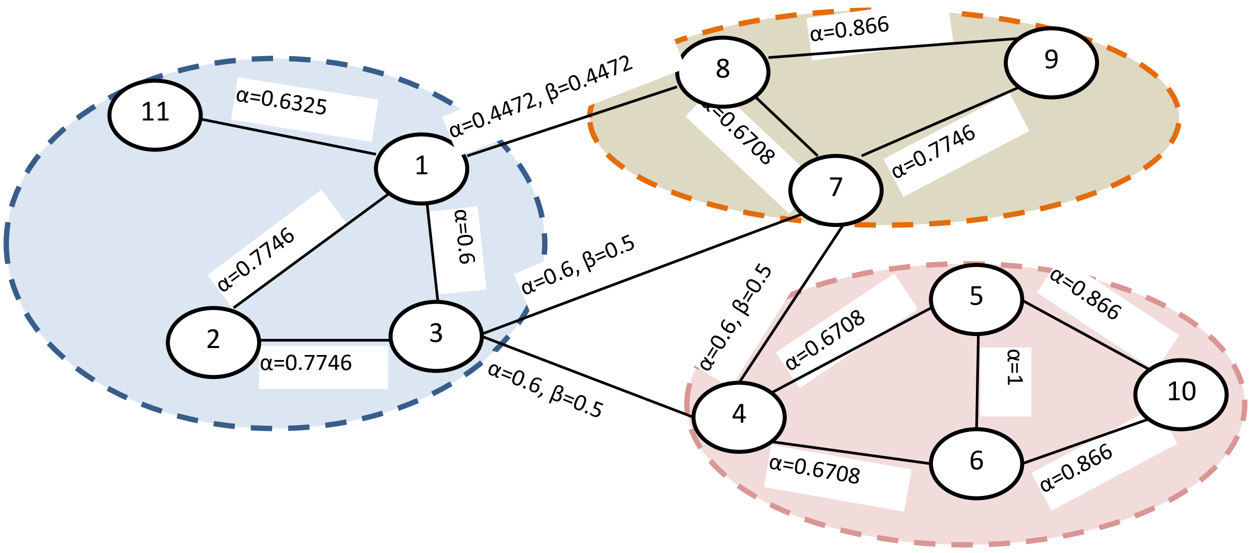


Figure 2. Calculating the value of α and β in a hypothetical network

3. Genetic and Memetic algorithm

As an efficient optimization tool, GA has considerable advantages compared to conventional methods. GA effectively examines a diverse search space of responses, including continuous variables, discrete variables and nonlinear constraints. Thus, GA is able to find appropriate answers to the problem in an acceptable computational time. The optimized response is searched using a stochastic process with the help of a population of answers. In this method, through improvement of population of responses from one generation to another generation the algorithm moves toward the optimal solution in the search space. In GA by applying three genetic operators of selection, cross-over, and mutations on the current population, the next generation is produced. These three operators act randomly.

GA can be performed as binary GA (BGA) and real GA (RGA). The first step in solving an optimization problem using the BGA is coding the variables which are shown by an array of 0 and 1. A set of these fields is called population. Then, the genetic operators of selection, cross-over and mutation are applied to the coded variables. Moreover, RGA uses the real values of variables and the genetic operators directly are applied on variables [25, 30].

Memetic Algorithms (MA)

MA is a meta-heuristic search method based on population inspired by the concept of meme by Dawkins. This method is a development of evolutionary optimization methods that apply certain tools to improve the members [26-28], [31-32]. In other words, the combination of evolutionary algorithms with local search is called MA algorithm [29-33]. These models are inspired by some models in nature's systems that combine the evolutionary adaptation of population with individuals' personal learning during life.

Therefore, it can be claimed that MA can be a combination of genetic algorithm (GA) and a local search that uses genetic algorithm to explore in the search space, and uses the local search for the exploitation of information in the search area. Memetic algorithm was first introduced by Moscato which is a combination of basic genetic algorithm and simulation algorithm of metals melting. Then, the algorithm was used by the Normans and Moscato. [33].

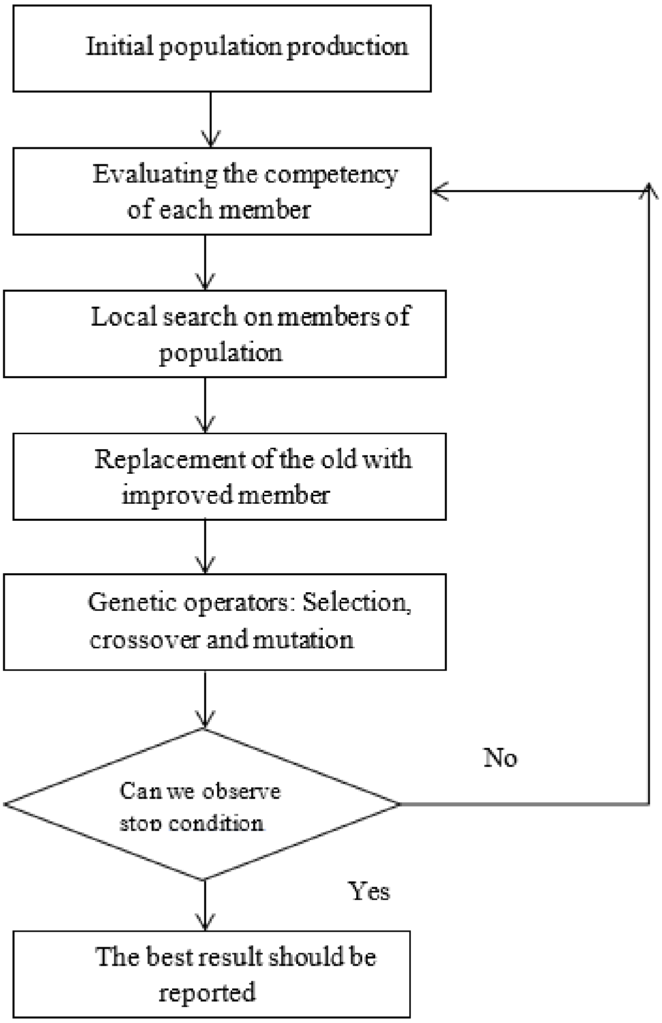

Initial population production
Evaluating the competency of each member
Local search on members of population
Replacement of the old with improved member
Genetic operators: Selection, crossover and mutation
Can we observe stop condition
No
Yes
The best result should be reported

Figure 3. The principle of MA Flowchart

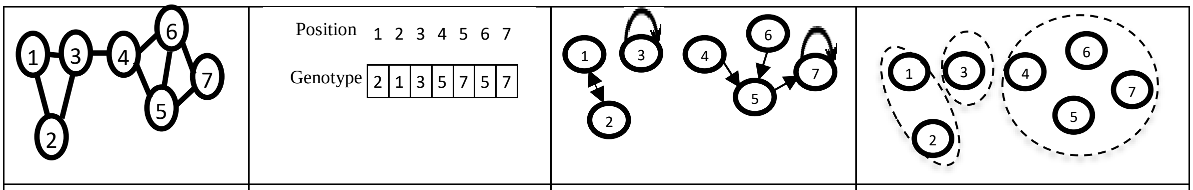
Memetic algorithm is similar to the genetic algorithm, but its components are called meme instead of the gene. A unique aspect of this algorithm is that before participating in the evolution process, all the chromosomes and children are allowed to gain a set of experiences by local search. As a result, MA is used to describe the genetic algorithm that clearly uses local search (Figure 3). Different types of MA have been reported in the literature [34-35] and may be classified into two different groups. The first group uses the effect of hill climbers in the area. Hill-climbing is a method that starts searching from one point in space (by an agent) and if a better answer regarding the current answer is found, it will replace the current answer. In this method, which is a greedy search, new search points are selected around the current point based on a random approach which depends on the issue. In the Memetic algorithm of the first group (combination of genetic algorithms and local hill-climbing search,), each member performs local hill climbing search to improve their situation. As a result of this operation, we have a population that its members are locally optimal. Then, the cross-over and mutation operators are applied to produce children. Children will be searched locally too. As a result, there is always local optimum.

The Memetic algorithm of second group performs the local search based on crossover. Crossover operator receives two chromosomes (from parents) and by combining the information produces two children. Therefore, it can be used as a local search strategy. In other words, the crossover operation in the XLS produces distributed children around parents with regard to the local promotion.

This article uses Memetic algorithm of the first group to implement the program. As mentioned before, in this group just like the genetic algorithm, after improvement of optimized answer there are crossover and mutation operators for producing children. In general, there are three operators for the genetic algorithm that will be briefly examined beside the problem.

3.1 Representation and Initialization

This method uses Locus [36] approach. To explain the implementation of the genetic algorithm for community detection problem, for example for a social network, (Figure 4) the stages of this method are clarified. This example, as shown in the figure, has 7 nodes.



(a) Graph topology (b) Gene representation (c) Corresponding graph (d) Community structure

Figure 4. Illustration of the Locus-based adjacency genetic representation. (a) The topology of the graph representing a complex network. (b) One possible genotype. (c) How (b) is translated into the graph structure, for example node 0 links to node 3, since gene g0 is 3. (d) The community structure[13]

Figure 4 shows the creation of genes based on Lucas method. A) Shows the topology of a complete network. B) A possible genotype of the network. C) Shows transformation of the B gene into a graph, for example node A is linked to node 2 so in the position of node 1 the value of genotype is 2. D) Shows the community structure [37].

In Locus display, each genotype g includes n genes $g_1, g_2, \ldots, g_n$ and each g_i gene can take the value of one of the nodes associated with the node i (which contains its own value as well). Therefore, the written amount of j at node i position represents the fact that there is a connection between nodes i and j. So these two nodes are in a community. Exposure of the display requires identification of all connections inside the network. All nodes that are connected together are regarded as a community. Using a simple return, the decoding stage can be performed in a linear time [37].

3.2 Crossover Operator

Combination or reproduction is an operation that is performed on two chromosomes in the existing population simultaneously and produces two new chromosomes, that exchanges characteristics of parents randomly from the point of intersection. The intersection point is selected randomly and can be single-point, two-point or multi-point. Increasing the combination speed leads to the increase of reproduction of population; however, the loss of chromosomes with high fitness is possible. In the proposed method, the single-point combination operator is used as shown is figure 5. In a single point combination, a combination point is randomly selected and two children are produced, in a way that the part before the selected point is from one parent and the rest is taken from the other parent. Therefore, combination operator does not produce an unacceptable response. The combination ratio of p_c ($p_c \in [0, 1]$) represents the probability of a combination operator on the parents.

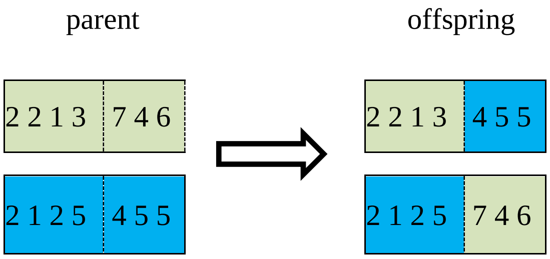


Figure 5. The combination stage of a) parents before the act of composition b) children of the new generation

3.3 Mutation

Mutation is an operator that acts on one of the chromosomes of the population and changes its characteristics only at the mutation point. From the perspective of the binary genetic algorithm this means changing the genes in mutation point on the chromosomes that are selected for mutation practice. But in this article, the mutation designates selection of another connection other than the existing connection with respect to other communications of the gene. The possibility of mutation of ($\mathrm{p_m}$) is determined by the operator that usually between 0.01 and 0.03 (Fig. 6).

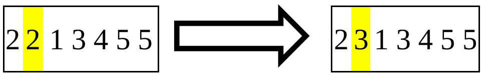


Figure 6. Mutation stage a) offspring before the mutation b) mutant offspring

3.4 Selection Operator

Selection designates an action in which the next generation of chromosomes is determined by the current population based on the law of survival selection. This is an important step in genetic algorithms and plays an important role in this process. There are various methods for this operation, such as Roulette Wheel and Tournament method. Tis study uses Roulette Wheel Selection method. A general method for selection of chromosome j is determining the probability of selection of Pj based on objective function value. To this end, N random numbers are generated and compared with the cumulative probability. The cumulative probability of the population is defined as follows:

$$C_i = \sum_{j=1}^{i} P_j \tag{8}$$

The chromosome i is selected for the next generation if $\mathrm{C_{i-1}} < \mathrm{U(0,1)} \leq \mathrm{C_i}$ where U is N random numbers between 0 and 1.

For each population the selection probability is obtained through the following equation.

$$P_i = \frac{F_i}{\sum_{j=1}^{Popsize} F_j} \tag{9}$$

Fi is the number of chromosomes i in optimizing function. Since there is not a specific convergence condition, the criterion for the end of optimization operation is based on the iteration of production of new chromosomes. Usually after a certain number of productions the optimization problem ends.

4. Adjusting Parameter’s Algorithms

In this study, Taguchi method is used to adjust the parameters of the proposed algorithms. In this way, at first, different levels of parameters are determined. Related parameters and values of Memetic algorithms are given in table 1.

Table 1. Parameters levels of algorithms

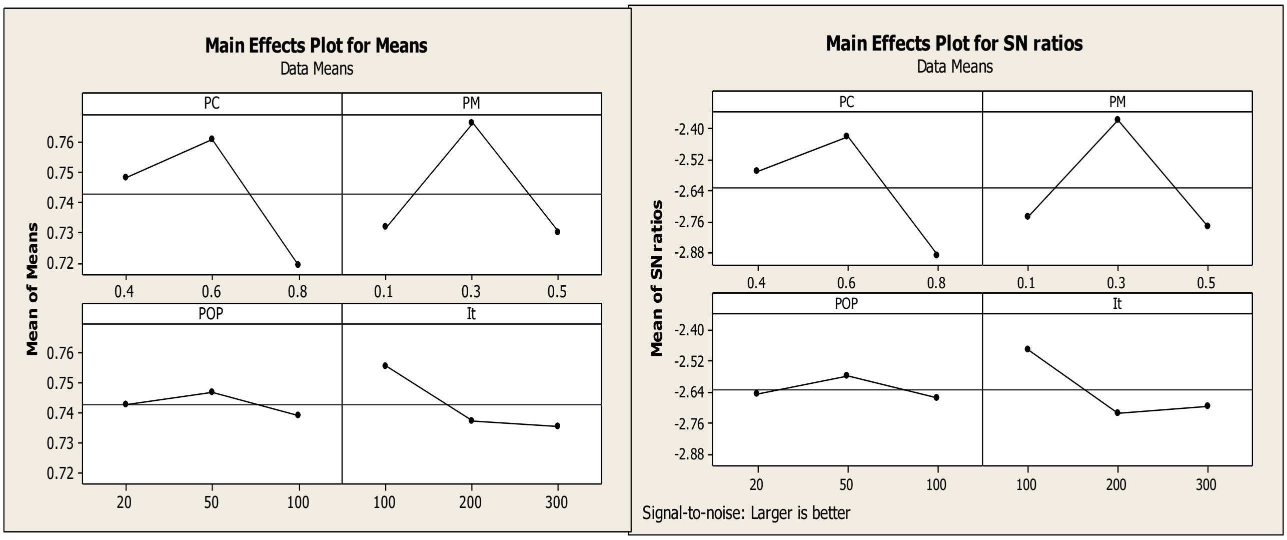
algorithms	parameters	Symbols	levels
Memetic	Crossover rate	p_c	0.4 – 0.6 – 0.8
	Mutation rate	p_m	0.1 - 0.3 – 0.5
	Pop size	pop	20 - 50 - 100
	iteration	It	100 - 200 - 300

Memetic run for 10 times and averaged values can be seen in table 2.

Value of parameters is available after running using the signal to noise method. This graph are available for Memetic algorithms that is showed in figures 7.

Table 2. Taguchi results to adjust parameter’s Memetic algorithm

Number of experiment	p_c	p_m	pop	It	ARI
1	0.4	0.1	20	100	0.750
2	0.4	0.3	50	200	0.771
3	0.4	0.5	100	300	0.724
4	0.6	0.1	50	300	0.747
5	0.6	0.3	100	100	0.793
6	0.6	0.5	20	200	0.742
7	0.8	0.1	100	200	0.699
8	0.8	0.3	20	300	0.736
9	0.8	0.5	50	100	0.723



Main Effects **Singal to nosie rate**

Figure 7. signal to noise for Memetic algorithm

Appropriate values for the parameters of Memetic algorithm have been reported in Table 3.

Table 3. best value for parameters of GA and NSGA algorithms

algorithm	parameters	Symbol	Value of parameter (practical discount)
Memetic	Crossover rate	p_c	0.6
	Mutation rate	p_m	0.3
	Pop size	pop	50
	iteration	It	100

5. Examining the Results

In order to express the accuracy of the algorithm, this paper uses two sets of networks. We can mention artificial and real-world networks; such as Zachary Club, Citation Journal network, football and boxing which is about America's policies. In these networks the real community structure in known in advance (in nodes). ARI value, which is between 0 and 1, is as a measure of performance measures considering the random partition expectation [16].

Table 4. parameters values of GA

Parameter	Value	Description
$popsize$	50	The population size
p_m	0.3	Mutation rate
p_c	0.6	Crossover rate
$gens$	100	Maximum number of iterations

The GA parameter values in Table 4 are regarded as the number of population members, percentage of mutation, percentage of combination, and the number of iterations of the program. Given the different tests for different modes, the obtained values are the most desirable ones.

5.1 Evaluation Criteria

To evaluate the performance of the algorithm the study used the following evaluation criteria:

ARI

Adjusted Rand Index (ARI) is provided by [38] and includes values between one and zero. If all nodes are located in the true community the value is 1, and if none of the nodes are in the true community the value is 0.

In the evaluation criteria each node is shown by nij that actually belongs to the true community of Lti as well as the obtained community of Lpj. Then, ni. and n.i display total nodes of the true community of i, as well as the total nodes related to the imagined community of j, respectively. The following relationship shows the amount of ARI.

$$ARI = \frac{\sum_{i,j}\binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i.}}{2} \sum_j \binom{n_{.j}}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i.}}{2} + \sum_j \binom{n_{.j}}{2}\right] - \left[\sum_i \binom{n_{i.}}{2} \sum_j \binom{n_{.j}}{2}\right] / \binom{n}{2}} \tag{10}$$

NMI

Normalized mutual information (NMI) is introduced as the performance measure in clustering quality and can be used for community detection quality. NMI shows the similarity between the class structure and the clusters structure but in this problem reflects the similarity between the true community and the detected community. Given two parts A and B of a network, C is the confusion matrix. In C, Cij is the number of nodes of community i of part A that are also in community j of part B . NMI I(A, B) is defined as follows:

$$I(A,B) = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} C_{ij} log\left(C_{ij}N/C_i.C_j\right)}{\sum_{j=1}^{C_A} C_i log(C_i/N) + \sum_{j=1}^{C_B} C_j log\left(C_j/N\right)} \tag{11}$$

If A = B, I(A, B) = 1; if A and B are totally different, I(A, B) = 0. The higher the value of NMI reveals the more approximate to the true communities are the detected communities.

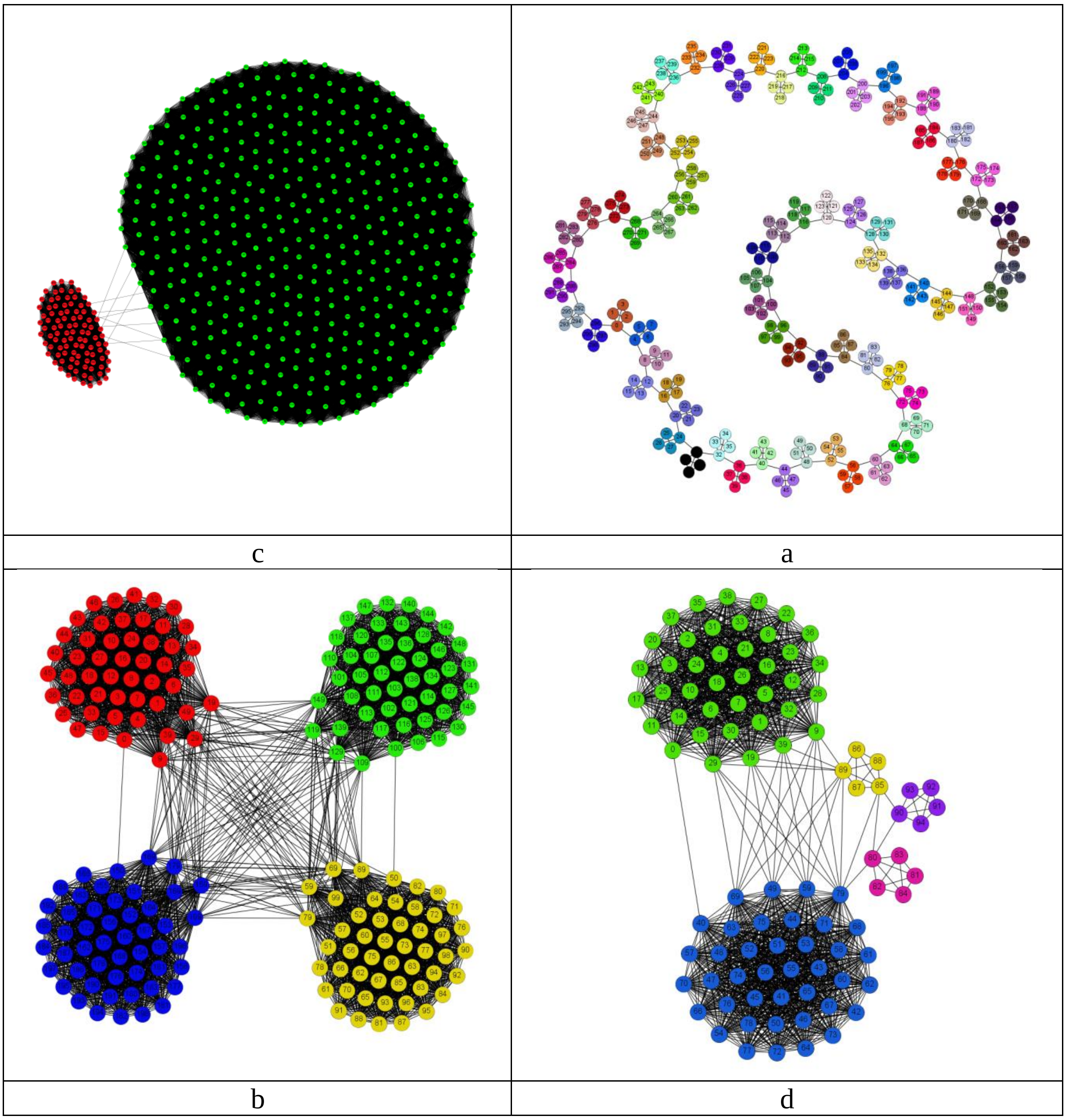
5.2 Test on Artificial Networks

To verify the accuracy of the algorithms in identifying complex communities, the algorithm is examined on networks with different characteristics. These networks are demonstrated in Figure 8. One network has 75 communities that each category contains 4 nodes, which are connected by an edge. Of course, in each category all of the nodes are connected to each other, i.e. each node in each category has at least 3 edges. The community is displayed in Figure a.

One network has 4 categories, and each category contains 50 nodes. The nodes in each category are fully linked to each other. On the other hand, nodes with multiples of ten anywhere in the network are connected to each other; in other words, each category is linked to other category by 50 edges.

A network is shown with two categories, one of the categories has 400 nodes and the other has 100 nodes. In each category, each node is connected to other nodes, and there are10 connection between these two categories.

A network is shown with 5 categories, with two similar categories, and another three similar categories. Each of the two larger categories are consisting of 40 nodes and the other three categories each containing 5 nodes. The proposed algorithm is used for the detection of the network communities. The results are shown in Table 5.



c

a

b

d

Figure 8. Artificial complex networks

In order to evaluate the effect of the proposed method, a number of networks in real-world are employed. The results of the network's community detection are given below, as well as the criteria for evaluating the results.

Table 5. The results of the artificial network

NMI	ND	nd_{out}	nd_{in}	Predicted Communities	True Communities	edge	node	
1	2.3247	30	204.355	75	75	525	300	a
1	248.8925	0.079	497.864	2	2	84761	500	b
1	16.9272	2.998	87.634	5	5	1620	95	c
1	47.8167	0.156	191.423	4	4	5054	200	d

Table 5 shows the results of the artificial network. As can be seen, the algorithm has identified the communities in each of the networks with 100 percent accuracy.

Now, given that the proposed algorithm in artificial networks has good performance, the performance of the presented algorithm on real networks that are usually referred to in the literature is investigated. There are four networks called Zachary's Club, America's policy book, risk map, and football clubs that are introduced. Then using the algorithm it has been tried to identify their communities.

Zachary Club [18]

Zachary Club is one of the most commonly used networks in terms of detection. The 34 members of the club make up the 34 nodes of the network. Relationships between members include 78 edges. It has two categories of membership. The network is shown in Figure 9.

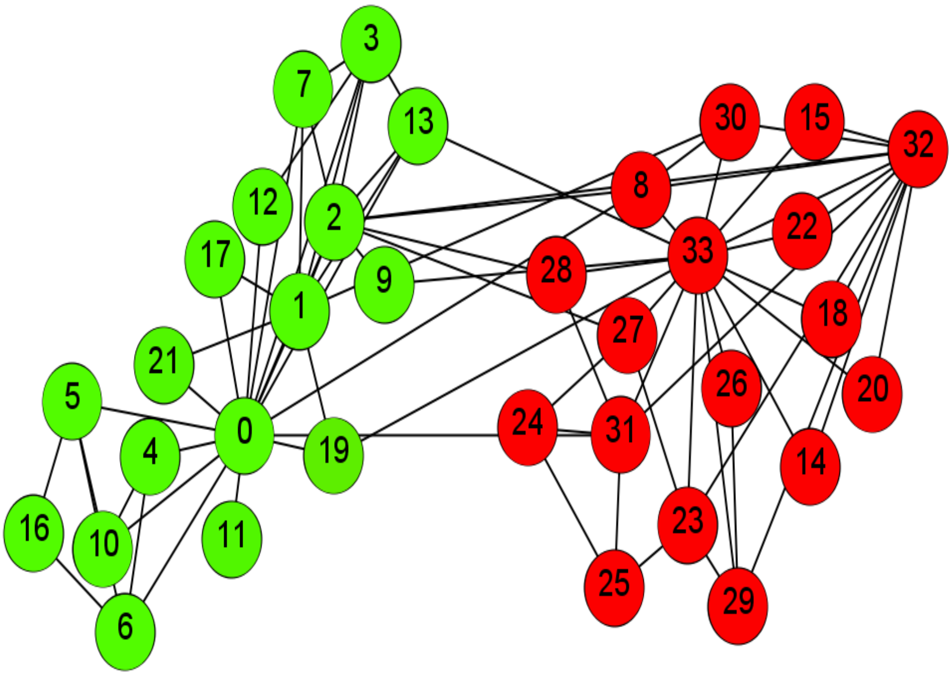


Figure 9. Community partition result of Zachary's karate club network using a) Modularity density maximization b) Modularity maximization.

America’s Policy Book [18]

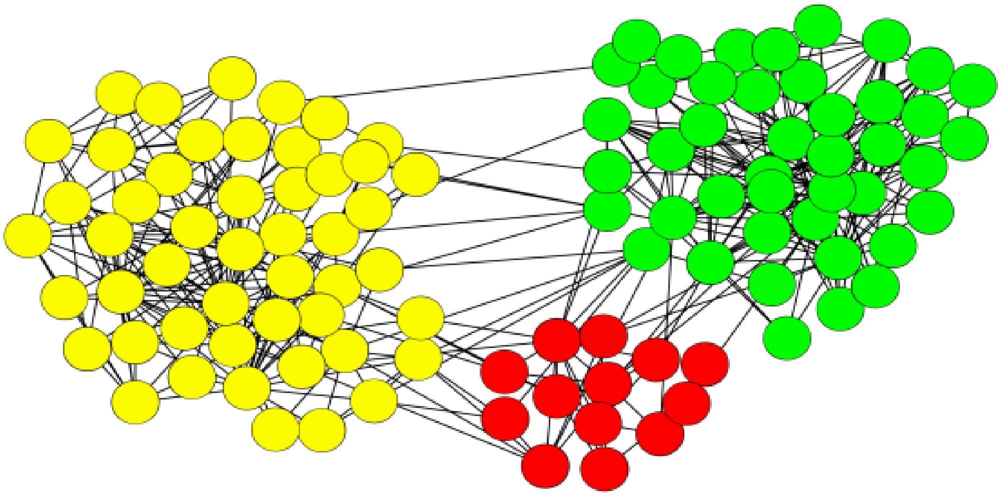


Figure 10. America’s Policy Book

In this network, the nodes represent 105 books about politics in America, purchased from the online bookstore Amazon.com, and the edges are between the pairs of books that are purchased by a buyer simultaneously. Books are divided according to their position or political alignment with them (liberal or conservative). Except for a number of books that are affiliated with no party the others could be clearly identified. The network is shown in Figure 10

Risk Map [16]

This network is one of the popular risk game maps, which are divided by six members. It is shown in Figure11.

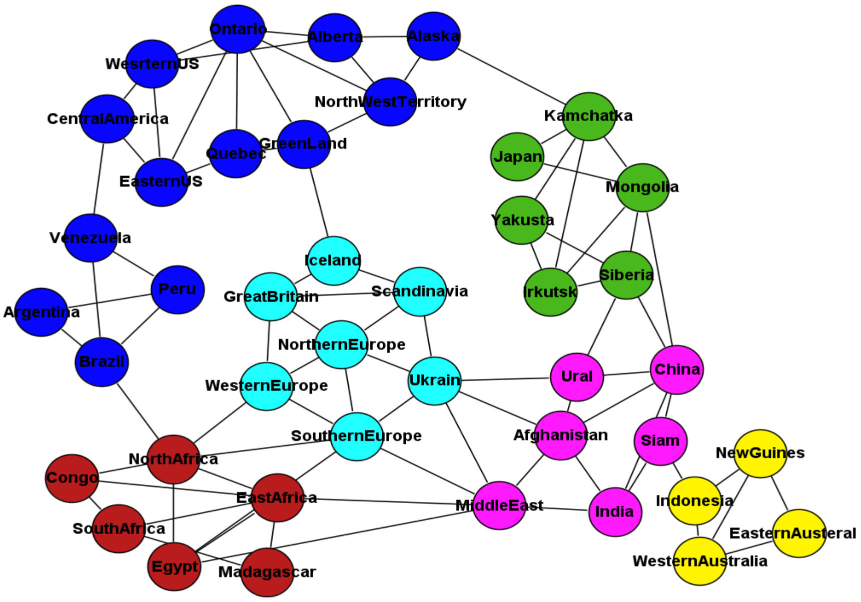


Figure 11. Risk map network; node shading indicates the six distinct continents in the game.

Football Club

The network shows America’s Football League first collected by Girvan and Newman [21]. Given that the network is somewhat complex, it is examined in the articles less than other networks. The network has 115 nodes and 615 edges. Each team belongs to a conference with 8-12 teams, which in fact display communities in the network. The network is shown in Figure 12.

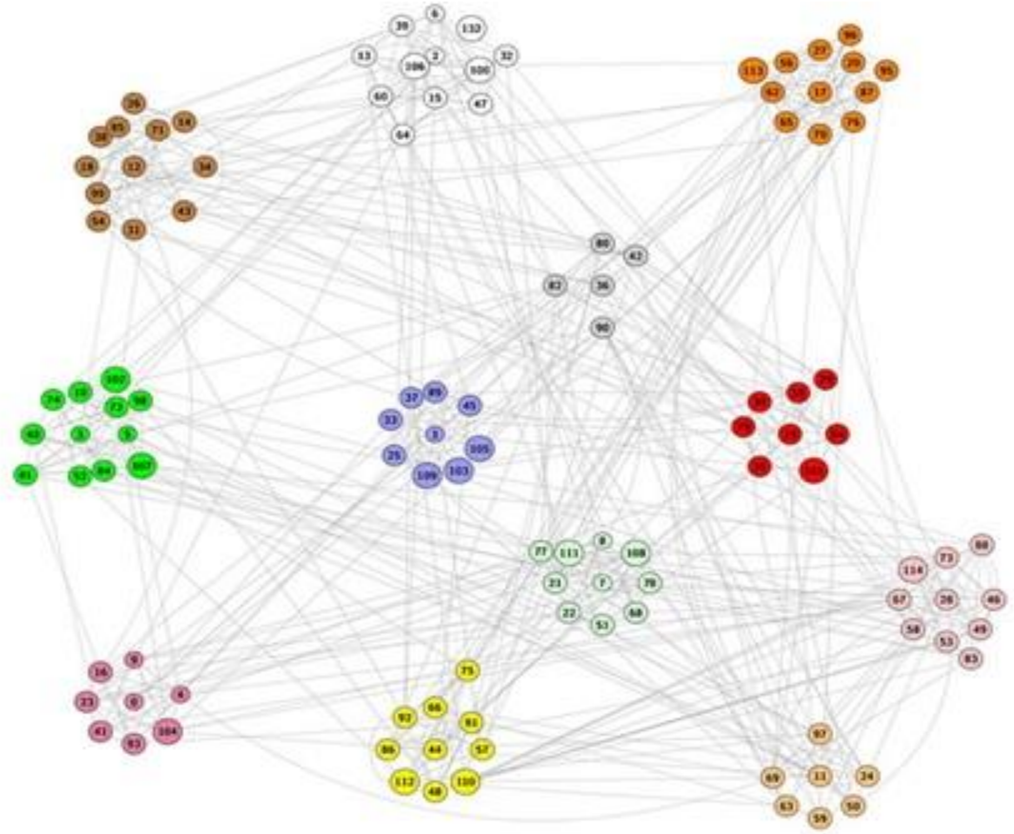


Figure 12. Football Club

6. Results

To demonstrate the function, the proposed approach was used in the networks in the real world, as well. The results of the tests are given in Table 6 and Table 7. To evaluate the results, the ARI criterion is employed in this table. And the results are evaluated with the real results of the problem that approves the degree of accuracy of the obtained results. The closer the amount of ARI to 1, [20] the better the accuracy of the obtained responses is.

As presented in table 7, two indices were used to investigate accuracy of the proposed algorithm; one shows the number of detected communities and the other one shows accuracy of nodes locations within the communities. As seen in the table, good performance was achieved for

Karate and football in terms of communities' number and accuracy; however, for risk map network, the number of detected communities is more than the real number. However, regarding high accuracy of NMI, the nodes are suitably located in the communities. Regarding Polbook network, although accurate number of communities is detected, its accuracy is far lower than other networks. In order to investigate algorithm quality, the results are compared with those obtained by 12 algorithms.

Table 6. The results of the real network

ND	nd_{out}	nd_{in}	**Network name**
1.67449309347604	0.983160589575919	4.33214677652800	Karate
1.29340390471647	37.9913854889965	53.5122323455941	football
3.11825356297995	0.51639777949322	9.87115846843417	PolBooks
1.4160123050500571	4.62994610333661	13.1260199333708	RiskMap

Table 7. Best Modularity density values and ARI obtained for four real world networks.

Network name	**nodes**	**edges**	**True Communities**	**Predicted Communities**	ARI	**NMI**	**Q**
Karate	34	78	2	2	1	1	0.4487
RiskMap	41	83	6	7	0.8644	0.90	0.9773
PolBooks	105	441	3	3	0.7367	0.6484	0.5370
Football	115	613	12	12	0.8873	0.9220	0.6829

Table 8. The comparison of Q between propose algorithm and 9 other algorithm

	MLAMA -Net	CLA -net	CNM	MIGA	Meme -net	GA -net	MOCD	MOGA -net	MOEA/D- net	**MA**
karate	0.4198	0.4188	0.3807	0.4188	0.4020	0.4059	0.4188	0.4198	0.4198	**0.4487**
Football	0.6050	0.6046	0.5733	0.5911	0.5888	0.5940	0.5958	0.5280	0.6044	**0.6829**
PolBooks	0.5255	0.5254	0.5178	0.5269	0.5178	0.5230	0.5208	0.5027	0.5236	**0.5370**

Table 8 compared our results from Memeic Algorithm with 9 other algorithms MLAMA-Net [27], CNM algorithm [39], Meme-Net algorithm [40], GA-Net algorithm [41],MOCD algorithm [42], MOGA algorithm [43], MOEAD-Net algorithm [44], MIGA [25] and CLA-Net algorithm [26] based on modularity Q for 3 real world networks

As is presented in table 1 value for modularity Q is better than all of other algorithms, for example the best value of Q was 0.4198 and gained by MLAMA-Net algorithm while this value is 0.4487 in proposed MA algorithm.

Table 9 compared our results from Memeic Algorithm with 3 other algorithms MIGA[25],MA[25] and GA[25] based on Normalized mutual information(NMI) for 3 real world networks

Table 9. The comparison of NMI between proposed algorithm and 3 other algorithm

The values of NMI at 3 different algorithms				
	MIGA	MA	GA	**Proposed MA**
karate	1	0.699	0.687	**1**
Football	0.916	0.910	0.532	**0.92**
PolBooks	0.585	0.455	0.521	**0.64**

As is observed in given table value of NMI is better than other algorithm for all real networks Karate, Football and Polbooks.

7. Conclusion

The present article used the Memetic algorithm method with real data. Due to the use of matrices with real data, it is easy to work with the network and its matrix relations, and to implement combination and mutation operations. Moreover, the study developed new criteria for density based on the relationship between nodes. Then, each network was evaluated. As the results indicate, the proposed method provides appropriate responses by taking into account the intended criteria for network classification. This method was evaluated for a number of artificial networks and existing community networks, which approves the efficiency of the proposed method. Future studies can apply this method on large and directed networks, as well.

References

1. S. Fortunato, Community detection in graphs, 2010, Physics Reports, vol. 486, no. 3–5, pp. 75–174.
2. M. Girvan, M. Newman, Community structure in social and biological networks, 2002, Proceedings of the National Academy of Sciences of the United States of America, vol. 99, no. 12, pp. 7821–7826.
3. S. H. Strogatz, Exploring complex networks, 2001, Nature, vol. 410, no. 6825, pp. 268–76.
4. Ben Sheldon, Community Detection Algorithms: A comparative evaluation on artificial and real-world networks, 2010.
5. M. Newman, Detecting community structure in networks, The European Physical Journal B – Condensed Matter, vol. 83, no. 2, pp. 823–883.
6. M. Newman, Finding community structure in networks using the eigenvectors of matrices, Physical Review E, no. 2, pp. 3–5.
7. Q. Yan, L. Wu, L. Zheng, Social Network Based Microblog User Behavior Analysis, 2013, Physica A: Statistical Mechanics and Its Applications, vol. 392, no. 7, pp. 1712-1723.

8. X. Zhang, J. Zhu, Skeleton of weighted social network, 2013, *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 6, pp.1547-1556
9. A. Beach, M. Gartrell, R. Han, August. Solutions to security and privacy issues in mobile social networking, 2009, Computational Science and Engineering, CSE'09, International Conference on, Vol. 4, pp. 1036-1042.
10. M. Salama, M. Panda, Y. Elbarawy, A. Hassanien and A. Abraham, Computational Social Networks: Security and Privacy, 2012, Springer Verlag, vol. 2, pp. 3-21.
11. A. Abraham, A. E. Hassanien, (Eds.). Computational Social Networks: Tools, 2012, Perspectives and Applications. Springer Science & Business Media.
12. M. C. Caschera, F. Ferri, P. Grifoni, SIM: A dynamic multidimensional visualization method for social networks, 2008, PsychNology Journal, vol. 6, no. 3, pp. 291-320.
13. U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, D. Wagner, On modularity clustering, 2008, IEEE Trans. Knowl. Data Eng. vol. 20, no. 2, pp. 172–188.
14. J. Eustace, X. Wang, Y. Cui, Community detection using local neighborhood in complex networks, 2015, Physica A: Statistical Mechanics and its Applications, vol. 436, pp. 665-677.
15. J. Xiang, Hu, T., Y. Zhang, K. Hu, J. M. Li, X. K. Xu, Chen, S. Local modularity for community detection in complex networks, 2016, Physica A: Statistical Mechanics and its Applications, vol. 443, 451-459.
16. B. Saoud, A. Moussaoui, Community detection in networks based on minimum spanning tree and modularity, 2016, Physica A: Statistical Mechanics and its Applications, vol. 460, 230-234.
17. Y. M. ElBarawy, R. F. Mohamed, N. I. Ghali, Improving social network community detection using DBSCAN algorithm, 2014, Computer Applications & Research (WSCAR), 2014 World Symposium on, pp. 1-6.
18. N. Aston, W. Hu, Community detection in dynamic social networks, 2014, Communications and Network.
19. Y. Li, G. Liu, S. Y. Lao, A genetic algorithm for community detection in complex networks, 2013, Journal of Central South University, vol. 20, no. 5, pp. 1269-1276.
20. Z. Li, J. Liu, A multi-agent genetic algorithm for community detection in complex networks, 2016, Physica A: Statistical Mechanics and its Applications, vol. 449, 336-347.
21. L.M. Naeni, R. Berretta, P. Moscato, MA-Net: A reliable memetic algorithm for community detection by modularity optimization, 2015, Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, vol. 1, Springer, pp. 311–323

22. T.N. Bui, B.R. Moon, Genetic algorithm and graph partitioning, 1996, IEEE Trans. Comput. vol. 45, no. 7, pp. 841–855.
23. M. Tasgin, A. Herdagdelen, H. Bingol, Community detection in complex networks using genetic algorithms. ArXiv preprint arXiv:0711.0491
24. C. Pizzuti, Ga-net: A genetic algorithm for community detection in social networks, 2008, Parallel Problem Solving from Nature–PPSN X, Springer, pp. 1081–1090.
25. R. Shang, J. Bai, L. Jiao, C. Jin, Community detection based on modularity and an improved genetic algorithm, 2013, Physica A: Statistical Mechanics and its Applications, vol. 392, no. 5, pp. 1215-1231.
26. Y. Zhao, W. Jiang, S. Li, Y. Ma, G. Su, X. Lin, A cellular learning automata based algorithm for detecting community structure in complex networks, 2015, Neurocomputing, vol. 151, pp. 1216-1226.
27. M. R. Mirsaleh, M. R. Meybodi, A Michigan memetic algorithm for solving the community detection problem in complex network. Neurocomputing, 2016, vol. 214, 535-545.
28. S. Fortunato, M. Barthelemy, Resolution limit in community detection, 2007, Proceedings of the National, pp. 1–8.
29. Z. Li, S. Zhang, R.-S. Wang, X.-S. Zhang, L. Chen, Quantitative function for community detection, 2008, Physical Review E, vol. 77, no. 3, p. 036109.
30. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Reading, 1989, MA: Addison-Wesley.
31. R. Dawkins, The Selfish Gene, 1976, Oxford, U.K.: Oxford Univ. Press.
32. P. Merz, B. Freisleben, Fitness landscape analysis and memetic algorithms for the quadratic assignment problem, 2000, IEEE Trans. Evol. Comput., vol. 4, no. 4, pp. 337–352.
33. P. Moscato, On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms, Memetic Algorithms' Home Page, http://www.densis.fee.unicamp.br/~moscato/memetic_home.html.
34. F. Herrera, N. Krasnogor and Daniel Molina, Real-Coded Memetic Algorithms with Crossover Hill-Climbing, 2004, Evolutionary Computation, vol. 12, no. 3, pp. 273-302.
35. A. Caponio, G. L. Cascella, F. Neri, N. Salvatore, M. Sumner, A fast adaptive memetic algorithm for off-line and on-line control design of PMSM drives, IEEE Trans. on Systems, Feb. 2007, Man and Cybernetics – Part B, Special Issue on Memetic Algorithms, vol. 37, no. 1, pp. 28-41.
36. Y.-J. Park and M.-S. Song, A genetic algorithm for clustering problem, 1998, ACGP98, pp. 568–575.

37. C. Shi, Z. Yan, Y. Cai, B. Wu, Multi-objective community detection in complex networks, 2012, Applied Soft Computing Journal, vol. 12, no. 2, pp. 850–859.
38. L. Hubert, P. Arabie, Comparing partitions. Journal of classification, 1985, vol. 2, no. 1, pp. 193-218.
39. A. Clauset, M. E. Newman, C. Moore, Finding community structure in very large networks, 2004, Physical review E, vol. 70, no. 6, pp. 066111.
40. M. Gong, B. Fu, L. Jiao, H. Du, Memetic algorithm for community detection in networks, 2011, Physical Review E, vol. 84, no. 5, pp. 056101.
41. Pizzuti, C. GA-Net: A Genetic Algorithm for Community Detection in Social Networks, 2008, PPSN, pp. 1081-1090.
42. C. Shi, Z. Yan, Y. Cai, B. Wu, Multi-objective community detection in complex networks, 2012, Applied Soft Computing, vol. 12, no. 2, pp. 850-859.
43. C. Pizzuti, A multiobjective genetic algorithm to find communities in complex networks, 2012, IEEE Transactions on Evolutionary Computation, vol. 16, no. 3, pp. 418-430.
44. M. Gong, L. Ma, Q. Zhang, L. Jiao, Community detection in networks by using multiobjective evolutionary algorithm with decomposition, 2012, Physica A: Statistical Mechanics and its Applications, vol. 391, no. 15, pp. 4050-4060.