Determining the Membership Percentage of Each Node in Overlapping Communities Using Multi-agent Collective Intelligence

Mohammad Akafan, Behrouz Minaei-Bidgoli, Alireza Bagheri

***Abstract*— The social network embodies real-life social graphs. Detecting communities or clusters from these graphs is an ill-posed difficult task. The communities are identified using the adjacent nodes that have shared edges and similar features. One of the principal concerns after community detection is to identify the active nodes in a network who attend several communities. So, finding communities that are overlapped in a social network is an important topic in social network analysis. This paper introduces an algorithm based on the multi-agent particle swarm optimization. The proposed algorithm detects overlapping as well as non-overlapping communities. Following the detection of overlapping communities, this algorithm can identify those nodes leading to overlapping, and ultimately it can determine the affiliation ratio of each node to the given community. The algorithm uses a special type of coding to identify the number of communities without any prior knowledge. In this method, the modularity measure is used as a fitness function to optimize particle swarm. Several experiments show that the proposed algorithm which is called Fuzzy Overlapping Community Detection based on Multi-Agent Particle Swarm Optimization (FOCDMAPSO), is superior compared with four other competitive algorithms. This algorithm is implemented over six well-known datasets and five LFR datasets in the literature. Our algorithm is capable of detecting nodes in overlapping communities with high accuracy. Furthermore, the proposed algorithm can detect the affiliation percentage of each node that leads to overlapping communities which is a novel feature in the area of the social network.**

***Index Terms*— Complex networks, Multi-agent, Overlapping community detection, Particle swarm optimization, Modularity.**

I. Introduction

MANY complex systems in society can be described in terms of networks or graphs. A social network is composed of a set of social actors and a set of bilateral relationships between these actors. The recognition of communities in networks is one of the major challenges in network science. One of the biggest concerns after community detection is to identify the main community of active nodes in the networks that cover several communities. Finding communities that overlap in social networks is an important topic in social networks analysis.

Mohammad Akafan is with Islamic Azad University, North Tehran Branch, Iran (e-mail: Akafan@ chmail.ir).

Behrouz Minaei-Bidgoli, is with Iran University of Science and Technology, School of Computer Engineering, Iran (e-mail: b_minaei@iust.ac.ir).

Alireza Bagheri is with Amirkabir University of Technology, Department of Computer Engineering, Iran (e-mail: ar_bagheri@aut.ac.ir).

A. Complex Network Clustering

A social network is a kind of social structure that comprises a group of social nodes like individuals and organizations and there is an interplay among these significant nodes. It is likely to analyze the entire structure of social networks from different perspectives. Local and global patterns could be emerged from this analysis and consequently, effective institutions and network dynamics could be identified. We can envisage a sort of dichotomy regarding social networks. On one hand, consistent patterns could be found in these networks, and on the other hand, social networks might be characterized as fluid, dynamic, and complex. There are numerous forms of complex networks in different aspects of life, such as communication networks of individuals in society, communication networks of students, communication networks of clerks of a university, transportation networks, communication networks of cells of the body for finding cancer and non-cancer cells, etc., these communities could be investigated and analyzed as complex networks. Using these investigations, we can divide a complex network into subnetworks with less outer connections and more interconnections. This technique can facilitate the analysis of complex networks. One of the biggest challenges in the clustering of a network into subnetworks is the existence of active internetwork nodes which lead to overlapping subnetworks [1-3].

B. Background

The network analysts developed community detection on the ground of different algorithms like Girvan and Newman algorithms [4], hierarchical clustering, spectral clustering, partitional clustering [5]. Newman offered an index regarding the modularity of community detection[6], which is widely welcomed by many researchers. This index led the community detection from a complex network division problem into a complex network optimization.

Recently, investigations on the notion of optimization have resulted in evolutional algorithms such as genetic algorithms [7], ant colony algorithms, simulated annealing algorithms [8, 9] and particle swarm optimization [10] to enhance the accuracy of community identification. This is due to less complexity of these algorithms in comparison with other ones. In many cases, to resolve the issue, greater memory capacity and more powerful processors are required to deal with the computational complexity of some of the algorithms. Chaotian and et al carried out some experiments and arrived at the idea that detection of a community based on particle congestion in a unique way to improve the accuracy amount of modularity [11]. The goal of using the previously presented particle swarm optimization algorithms for community identification is that they are only able to detect non-overlapping societies, and this refers to the genes created in this regard. For instance, in [12], it finds sub-networks in non-overlapping networks. They offer a method for PSO in which there is a leader in every repetition instead of having a gbest. In this method, for decreasing the number of repetitions, the particles follow the leader to reach an optimized solution. In another word, the method provides personal updating based on local information. The main limitation of this method is that it only detects the non-overlapping network.

In recent years, many researchers have investigated influential methods for detecting overlapping communities, so that the method of link clustering has been recognized as one of the effective ways for identifying overlapping communities, these kinds of methods cluster the communication links between nodes instead of clustering the nodes themselves. The benefit of this method is that the cluster provided from the clustering is a subgraph of the main graph, therefore permitting a node to be available in numerous sub-graphs [12, 13].

For instance, at Stanford University in 2011 through 2013, several research papers by Yang and Leskovec [14, 15] have been presented. Their primary axis is to analyze the graph dependency model (AGM), in which a new measure for community detection is presented. In this method, the representation of the graph is a two-part structure consisting of nodes and communities. For the relationship between a node and community, the weight of belonging the node to the community has been used. This is the basis of some of the methods presented in this type of research, which is interpreted from its final approach to Bigclam. This is a new approach to community detection, which is based on model estimations, but this method has a big problem, and it has problems with societies in which nodes become dense.

Particle swarm optimization was proposed by James Kennedy and Russell Eberhart [16] in 1995, whose main idea was to simulate the processing of birds moving. The collective intelligence is derived from the connection between several simple components, each of which adjusts its behaviors and connections with other members of the group based on certain rules [17]. The collective intelligence of particles can converge rapidly and has quantitative parameters for processing, and it can solve problems in a nonlinear way and combines with other optimization algorithms, to not quickly gets stuck in the optimal locale.

Researchers consider two main attributes for collective intelligence: the first characteristic is the impact of the environment, which includes responses to environmental stimuli; this is especially important in communities in which there is no difference between several components of the society, and the second one called self-organization that involves attuning the movement of a component based on both its previous experience and the movement of other components of the group [17, 18].

Asim et al.[19] introduced two methods, LAA and LOA, which are generalized to the Louvain method [20]. The Louvain method for community detection is a method to extract communities from large networks. The method is a greedy optimization method to runs in time $O\left(n \log n^2\right)$ in the number of nodes in the network [20].

The set of all Pareto optimal solutions is called the Pareto optimal set and the corresponding set of the objectives is called the Pareto obverse. The solution of MOPs is a Pareto optimal set instead of just one solution. They use the concept of Pareto dominance to allow the heuristic to handle problems. It employs an external repository of particles that is used by other particles to guide their flight. However, this algorithm is originally designed for continuous MOPs [21].

Many multi-objective evolutionary algorithms (MOEAs) have been proposed and find wide applications. However, no MOEAs are applied to overlapping community detection [22].

C. Contributions of This Paper

The novelty of this paper is using of the multi-agent concept for the detection of overlapping communities and also using a special kind of particle coding for the division of complex communication networks into subnetworks for identifying the number of communities. In this research, the fitness function for particle evaluation is the modularity, and then, using existing community coding, find nodes that lead to the overlapping of the network, and then, by measuring the modularity, we determine their main community. In this end, the binary particle swarm optimization algorithm is used to improve detection.

- Using the multi-agent concept for the detection of overlapping communities.
- For the complex network clustering problem, a discrete PSO framework is proposed.
- For the overlapping community detection problem, a multi-agent PSO framework is proposed.
- By using multi-agent PSO, we can fuzzy approach to calculate the percentage of the membership of each node to the overlapping communities.

In the first part of the paper, the particle optimization algorithm, the concept of multi-agent and modularity are reviewed, then in the second section of the paper, the multi-agent algorithm is brought up and in section three, several experimental outcomes are provided regarding the datasets exploited by the researchers. Finally, the conclusion is presented in section four.

II. Background and Related Works

A. Some definitions about community

In this study, an undirected network can be modeled as an undirected graph. A social network can be modeled as a G = (V, E) graph, where V is a set of nodes or vertices and E is a set of edges that interact between nodes. To explain the community definition and detection, some basic concepts are needed.

Definition 1: In this graph, we form the proximity matrix of the graph edges, the value m_{ij} represents an element of the adjacency matrix that is placed in row i and column j. If the relationship is between the two elements, its value is equal to P_{xy} as defined before. Adjacency matrix A is a zero-one matrix.

$$m_{ij}, m_{ji} = 1, if\ (v_x, v_y) \in E \tag{1}$$

Definition 2: The degree of a node can be represented by $k_i = \sum_{j \in G} A_{ij}$.

Definition 3: For a sub-graph C ⊂ G to which node v_i belongs, $k_i^{in}(C) = \sum_{j \in C} A_{ij}$ is the number of edges connecting the node v_i to the other nodes belonging to C.

Definition 4: For a sub-graph C ⊂ G to which node v_i belongs, $k_i^{out}(C) = \sum_{j \notin C} A_{ij}$ is the number of edges connecting the node v_i to the nodes not belonging to C.

Definition 5: $\{\forall n \in V, \forall C \in G \exists n \in C\}$, So every node belongs to C.

Definition 6: An Overlap community is $\{\forall c \in C, \forall n \in c_1, \forall n \in c_2 \mid c_1 \cap c_2 \neq \varnothing \exists n \in O\}$, O is an overlapping community.

In a strong sense if $k_i^{in}(C) > k_i^{out}(C), \forall i \in C$ the sub-graph C can be regarded as a community.

B. Multi-Agent System

A Multi-Agent System (MAS) is composed of a group of autonomous software agents. They are capable of realizing the desired objectives on a cooperative basis. Learning and functioning autonomously along with having cooperation with other agents are among the major characteristics of the agent. These characteristics show that agents have the capability of independent and asynchronous execution, and can discover relevant knowledge from the environment [23, 24].

Multi-Agent Collaborative Search (MACS) is regarded as a framework for the operationalization of population-based and hybrid approaches for multi-objective optimization. In this framework, some intelligent technologies are combined to get local and global objectives.

With the help of individual actions, each agent has the room for the convergence to Pareto optimal set. Individualistic actions include local moves based on memetic algorithms [25].

The objective of the multi-agent system is to make multiple autonomous agents work together efficiently to attain collective group behavior through local interaction [26].

C. Particle Swarm Optimization

Particle swarm optimization was proposed by Kennedy and Eberhart [16] and inspired by flying birds. This method is one of the most outstanding methods for achieving optimal global solutions. At first, the particle swarm optimization algorithm generates the initial population of particle swarm by creating a completely random population and begins to search for optimized particles using particle updates.

$$v_i^{k+1} = wv_i^k + \underbrace{c_1 r_1\left(pbest_i^k - x_i^k\right)}_{(1)} + \underbrace{c_2 r_2\left(gbest_i^k - x_i^k\right)}_{(2)} \tag{2}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{3}$$

$$v_{ij} = \begin{Bmatrix} v_{ij} = v_{\max}, v_{ij} \geq v_{\max} \\ v_{ij} = -v_{\max}, v_{ij} \leq -v_{\max} \end{Bmatrix} \tag{4}$$

$$v_i = \left(v_{i1}, v_{i2}, v_{i3}, ..v_{iq}\right) \tag{5}$$

$$x_i = \left(x_{i1}, x_{i2}, x_{i3}, ...x_{iq}\right) \tag{6}$$

Equation (1) is the vector of particle velocity and equation (2) is the vector of particle position, and r_1, r_2 are two random numbers in the interval [0,1], k also indicates the k^{th} generation of particles, w is a coefficient for particle velocity as the inertia of particle, and c_1 is the acceleration coefficient for pbest and c_2 is considered as the acceleration coefficient for gbest and $V_{\max}$ is an upper bound on the velocity of a particle. Part 1 of equation (1) expresses the particle's self-cognition mechanism and Part 2 of equation (1) represents the mechanism of the particle's socialization [27].

The above mechanism is proposed as the standard particle swarm optimization (SPSO), which is suitable for solving problems with continuous fitness function. To solve hybrid optimization problems, a binary PSO is proposed [28], with a sigmoid function, the particle position values are mapped to zero or one, but their velocities change according to the standard PSO.

$$x_i^k = \begin{Bmatrix} 1, \rho \leq S \\ 0, \rho > S \end{Bmatrix}, S = Sigmoid\left(v_i^k\right) = \frac{1}{\exp\left(-v_i^k + 1\right)} \tag{7}$$

In equation (7), ρ is a random number between [0,1].

D. Modularity

Modularity [6] could be defined as an indicator of evaluation for communities, the main idea behind this indicator is to compare the density of community with the density of a random network based on equation (8).

$$Q = \frac{1}{W} \sum_{C \in P_i, j \in C} \sum \left[A_{ij} - \frac{K_i K_j}{W}\right] \tag{8}$$

where C is a community of all communities, *i* and *j* are two nodes of community, A_{ij} is the value of the item of adjacency matrix, and W is the sum of the weights of the inputs and outputs communicators in the graph resulting from the grid, every node calculated twice in an undirected graph. And K_i is the degree of the *i*-th node.

The Q value is defined as a number between -1 and 1, and the larger the number, the better the community is detected on the network, and it can be said more accurately.

III. Description of the Proposed Method

In this section, the proposed algorithm for overlapping community detection problem is offered. We select the random approach because it can promote good population variety as well as having small time complexity and what is most important is that it has a slight impact on the objective dimensions since the dimensions of our objectives are rather high.

First, a particle representation scheme and its updating rules used in the proposed DPSO framework are given, and then the swarm initialization is described. This algorithm witch functions on the base of PSO and multi-agent concept can be regarded as fuzzy detection overlapping community detection, so we name it FOCDMAPSO (Fuzzy Overlapping Community Detection Multi-Agent Particle Swarm Optimization). The proposed algorithm adopts modularity to classify communities. This algorithm is made up of four main parts, and we answered four questions in this respect. Firstly, what is the coding of the optimization and, secondly, how is the mechanism of updating the particles and, thirdly, how the fitness function is defined to determine the society and fourth, how the pattern of overlapping communities is detected?

In this method (FOCDMAPSO), each agent executes independently and acquires the knowledge of its environment. As an outcome, each of the agents provides the best of its particles as the best solution for detecting social communities, which is a non-overlapping community. The main objective is the detection of overlapping communities, so a coordinator analyzes the results of the agents and discovers overlapping communities.

A. Encoding of particle swarm optimization and particle splitting

In our proposed method (FOCDMAPSO), first, it is assumed that each node is only in one community, and communities do not overlap.

To solve the complex network clustering problem, in this paper, we redefine the term position and velocity used in PSO in discrete form. The definitions are as follows:

1) Definition of position

In PSO, the position vector represents a solution to the optimized problem. For the network clustering problem, the position permutation of a particle i is defined as $X_i = \{x_1, x_2, \ldots, x_n\}$. Each dimension of position is a random integer between 0 and 1, $x_i \in [0,1]$, where n is equal to the total vertices number of the network. If $x_i = x_j$, then we take it that node i and j belong to the same cluster.

Encoding is based on assigning a tag to each community and is defined as follows:

$$P_i = (p_{i1}, p_{i2}, ..., p_{ij}, ..., p_{iv}), p_{ij} \in \{l_1, l_2, ..., l_k, ..., l_m\} \qquad (9)$$

P_i is a particle and P_{ij} shows the community label of node j in particle P_i and l_k is the label of community k.

Given the above, if $P_{ij} = L_k$, then the node j belongs to the community k.

The method can automatically determine the number of communities based on community labeling without having any prior knowledge.

To this end, our algorithm splits the network into two communities in the first stage.

The main challenge emerges after the first stage because the process for community detection has not yet been completed and the nodes of society are limited to just two communities.

Figure 2 gives an illustration of how the discrete position of a particle is coded and decoded.

The motive behind the definition of the position vector is that it is straightforward and easy to decode so that it will lower the computational complexity. The reason behind the definition of the position vector is that it is forthright and easy to decode so that it will lower the computational complexity.

For instance, if there is a network with 34 nodes, after the first stage, we will have a particle with 34 members, which are located at community 0 or 1.

So, this network is divided into two subnetworks, community 0 and community 1 groups.

2) Definition of velocity

Velocity works on the position sequence and it is rather vital. A good velocity gives the particle a leadership and controls whether the particle can reach its destination and by how fast it could. The discrete velocity of particle i is defined as $V_i = \{v_1, v_2, \ldots, v_n\}$. V_i is real-coded.

The network may have more uninvestigated subnetworks. To divide the network into other subnetworks, a divisional scenario is needed, so that we use a recursive division scenario. It means that at first a community is divided into two communities, then each of its communities is divided into two other communities and this process will stop when the following requirements are met:

1) The number of repetitions will be completed

2) In each step, the value of the fitness function is less than the previous one

The first motivation of the velocity definition, in the style, is to prevent particles from flying away because, in general, it is necessary to set a threshold V_{max} to inhibit particles from flying out of the boundaries.

From what has illustrated, we see that the proposed DPSO framework has the following features.

1) The definitions of discrete position and velocity are straight forward and very simple.

2) The newly defined arithmetic operators are very easy to realize, which greatly lower the computational complexity.

3) The proposed DPSO framework does not need to know the clusters of a network in advance; it can automatically determine it by itself. The newly designed DPSO framework seems to be very suitable for solving the network clustering problem.

B. Particle swarm update

Particle updates involve the velocity and position of each particle. Particle velocity update is performed based on equation (1), namely standard PSO, but particle position update is not performed based on the standard PSO. According to equation (10), the particle position update is performed using the sigmoid function, which is either 0 or

1. Consequently, we have a new function that specifies the position of a particle in community 0 or community 1.

We define $xMin$ equal zero and $xMax$ equal one for standardization of x. According to equation (10), calculate X_{new}.

The motive behind the definition of the position vector is that it is straightforward and easy to decode so that it will lower the computational complexity.

$$x = min(max(x, xMin), xMax)$$
$$X_{new} = \begin{cases} 0, x < 0.5 \\ 1, x \geq 0.5 \end{cases} \quad (10)$$

C. PSO fitness function

To change the problem of overlapping community detection from a division problem into an optimization problem, a fitness function must be selected to be optimized. In this research, the modularity function is used as a fitness function. Newman and Girvan [6] presented the first modularity function as a quality assessment index for community detection, so that by changing the value of this function from zero to one, the division of the graph as a social network into smaller societies is performing more accurately, and with more quality.

For the case when the ground truth of a network is known, we adopt the so-called normalized mutual information (NMI) index described in [29] to estimate the similarity between the true clustering results and the detected ones. Given two partitions A and B of a network, let C be the confusion matrix whose element C_{ij} is the number of nodes shared in common by community i in partition A and by community j in partition B. The $NMI(A,B)$ is then defined as:

$$NMI = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} C_{ij}\log\left(C_{ij}N / C_i C_j\right)}{\sum_{i=1}^{C_A} C_i \log\left(C_i / N\right) + \sum_{j=1}^{C_B} C_j \log\left(C_j / N\right)} \quad (11)$$

where $C_A(C_B)$ is the number of clusters in a partition $A(B)$, $C_i.(C_{.j})$ is the sum of elements of C in row i(column j), and N is the number of nodes of the network. If $A = B$, then $NMI(A,B) = 1$; if A and B are different, then $NMI(A,B) = 0$. The NMI is a similarity measure proved to be reliable in [30].

For the case when the ground truth of a network is unknown, we use the modularity Q.

Therefore, the fitness function is modularity function Q that defined as equation (8)[6], because when using NMI for fitness function, after finish iteration, the result of every agent is ground truth.

D. The framework of the Proposed Algorithm for Overlapping Community Detection

In the proposed algorithm (FOCDMAPSO), the execution of an agent may produce a particle and each node in the particle is placed in one community. This process will result in the division of a community into two communities or even more ones on a recursive basis. Finally, an agent might detect a non-overlapping community. In the provided algorithm, we see an increase in the number of agents from one to N. A non-overlapping community will be detected for each agent after the execution of the process.

The evaluation and supervision of the responses of the agents will be done by a coordinator to determine the shared nodes between detected subnetworks. These shared nodes comprise the common ground of the overlapped communities.

In the community structure encoding of a particle, the community assignment of each node is determined by a community label. However, the same community label in different particles does not need to represent the same community due to the randomness of execution. Therefore, the decoded community structures from particles should be aligned (matched) before coordinating operations.

The output of each agent is a particle. One particle consists of nodes that are members of several communities, so in each particle, there are a set of nodes that are members of the community I_k. The coordinator finds the nodes that led to the overlapping network by intersection operations between particles. In the intersection operation, the following three states may occur:

All nodes are located in the same particle so that the detected communities are non-overlapped

Some nodes are common points of different particles; these nodes are those active ones that lead to overlapping communities. There is no shared node between a set of particles. According to the fitness function, this state is impossible. Given the second state assume, when the result of agents demonstrate that a node belongs to more than community, this node can be a good candidate to be selected as an active node leading to the creation of overlapping communities. FOCDMAPSO explains in figure 3.

Algorithm 1. Framework of the Proposed Algorithm for Overlapping Community

Parameters:
1- Max Iteration: $maxIter$
2- Swarm size: pop
3- Particle size: Node number of the network
3- The learning factors: c_1, c_2
4- Inertial Weight: w
5- Agent number: $maxAge$
Input: The adjacency matrix A of a network
Output: Result of agent (representation of network cluster)
Step1: **for** $1, 2, \ldots, maxAge$ **do**
  Go to step2
Step2: **Initialization**
2.1- Position initialization: $P = \{x_1, x_2, \ldots, x_{pop}\}$
2.2- Velocity initialization: $V = \{v_1, v_2, \ldots, v_{pop}\}$
2.3- Personal best position initialization
2.4- Personal Best error
2.5- Personal error
2.6- Best position for group
2.7- Best error for group
Step3: **Cycling**
  For $i = 1, 2, \ldots, pop$ **do**

3.1- Calculate new velocity V_i^{t+1} for the i_{th} particle according to equation (1).
3.2- Calculate a new position X_i^{t+1} for the i_{th} particle according to equation (10).
3.3- Evaluation of X_i^{t+1}
3.4- Calculate Best position for the group based on the personal best error
Step4: **Stopping criteria:** if $t < maxIter$, then $t++$ and go to Step3 otherwise, stop the agent and output particle
Step5: Determine the shared nodes between detected subnetworks in all results by the coordinator

Eq (12) indicates a fuzzy approach to calculate the percentage of the membership of each node to the overlapping community:

$$P(n \in c_i) = \frac{\sum_{j=1}^{a} A_j(n,c_i)}{a}; \quad A_j(n,c_i) = \begin{cases} 1 \\ 0 \end{cases} \tag{12}$$

$A_j(n,c_i) = 1$ if the agent j believes that $n \in c_i$, otherwise $A_j(n,c_i) = 0$, where n is a candidate node, c_i is a community label and a is the number of agents.

E. Complexity Analysis

1) Space Complexity

In our algorithm, two main memorizers are needed. The first one is the input data as adjacency matrix memorizer, which needs a complexity of $O(n^2)$ and n is the number of vertices of the network. The second memorizer is for the particles, say there are N particles, and then the complexity is $O(Nn)$ thus, the total space complexity of our algorithm is $O(n^2)$.

2) Computational Complexity

The main time complexity lies in Step 3 of our algorithm since Step 1 can be accomplished in linear time. Here, we use n and m to denote the vertex and edge numbers of the network, respectively.

Step 3.1, 3.2 and 3.4 need $O(n)$ basic operations. Step 3.3 needs $O(n^2)$ basic operations. So, the worst-case time complexity of FOCDMAPSO is $O(n^2)$.

IV. Experimental Results and Analysis

To compare the experimental results with other algorithms in modularity function and another analysis in the multi-agent method, we have taken six standard networks for this experiment. The maximum number of iterations in the evaluation function is 100.

Several experiments show that the division of the society may be different, but they have the same modularity (with a precision of two tenths), so the space of the issue has much modularity and the PSO is supposed to achieve one of the best modularity. The modularities found as gbest are very close to each other, so the problem has several solutions.

Now, when multiple agents are used in the problem, each agent results in an optimally global particle. This article has a modularity value. This value is very close to the modularity of the particles obtained from other agents. The test experience of the proposed algorithm shows that if the response of the two agents is different in terms of division. their difference is in one or two nodes, but their modularity is either the same or with an accuracy of 0.01 difference in value. Four of the current algorithms are compared with the proposed algorithm presented in this article, which details are given in this section.

A. Description of the datasets

In this section, six experiments will be done to validate the performance of the algorithm on different real networks.

We have taken six standard networks for which the number of partitions is known and that have been taken into consideration to evaluate the effectiveness of the proposed technique. These six standard networks are known as the Zachary's Karate Club[31], the American College Football [31], the Dolphin social network [32], the SFI social network [33], the Netscience social network [34] and Powergrid social network [35]. The description of these networks is given in Table I.

Real-world networks do not represent some features, such as the mass distribution of nodes and the size of society in real systems, because in those networks all vertices have the same degree, and all network communities have almost the same construction.

Therefore, the standard graphs or standard networks based on the graph structure called LFR were presented by Mr. Lancichinetti and et al. in 2008[36]. They use adjustable power rules and eventually generate a graph as a complex network. Today, many researchers active in the analysis of social networks use this type of network as a dataset to evaluate their algorithms. In this type of network, $£_1$ and $£_2$, respectively, represent the law of the power of degree distribution and the law of the power of the size of society.

Each vertex shares a fraction of its $\mu - 1$ vertex with the other vertices of its community, as well as a fraction of μ the vertex with the vertices of other communities. Its value is equal to $0 \le \tau \le 1$, in other words, μ indicates a fraction of the edges within the community that is located on each node.

In addition to the real datasets described, we generated five datasets based on LFR and according to the parameters of which are given in the table below and compare the results with the Louvain algorithm.

B. Settings of the experiments

The implementation of the proposed algorithm is performed with the settings in Table IV for each real dataset.

Particle size is the number of nodes in the graph, w is a coefficient for particle velocity as the inertia of particle, and c_1 is the acceleration coefficient for personal best and c_2 is considered as the acceleration coefficient for global best which must be set before running the algorithm. Our experiments are conducted on a personal computer equipped with an Intel Corei7 CPU (2.5 GHz) and 8 Gigabyte memory. The algorithms are validated and implemented under the environment of Python 3.7.0.

In these experiments, the number of executive agents is considered ten. Furthermore, we added another agent which is considered as a coordinator for evaluating common nodes in situations wherein subsection 3.4 was expressed. The coefficients c_1, c_2, and w were obtained experimentally, in this regard, we have chosen each of the parameters c_1, c_2, and w in the range of 0 to 2 with an interval of 0.5 in each step. The experimental results for the selection of these parameters on Zachary's karate club dataset are shown in Table III and figure 4.

Approaches to let the inertia weight dynamically vary and those can be classified with the following definitions: Random adjustments, Linear decreasing, Nonlinear decreasing, Fuzzy adaptive inertia [37] but we are not using dynamically, so we use to constant.

C. Experimental results

Tables V, VI, VII, and VIII compare the modularity of the six datasets using the MR-MOEA algorithm [38], IMOQPSO algorithm [22], the MCMOEA algorithm [39] and the MODPSO algorithm [21] with maximum modularity of FOCDMAPSO algorithm. In table III, the first value from the top represents the American college football dataset, in which the modularity of FOCDMAPSO algorithm is 0.456 and modularity the MR-MOEA algorithm is 0.306, also in Zachary's karate club dataset second row, the modularity of FOCDMAPSO algorithm is 0.419 and modularity the MR-MOEA algorithm is 0.229 and also in the Dolphin social network dataset last row, the modularity of FOCDMAPSO algorithm is 0.485 and modularity the MR-MOEA algorithm is 0.271.

According to the results of the experiments carried out following the tables, the proposed algorithm outlined in this article has improved on the Dolphin social network and Zachary's karate club and American college football datasets in the modularity of community detection. Therefore, it can be said that the use of the FOCDMAPSO algorithm to determine nodes in communities has improved the modularity of community detection. The results are shown in tables V, VI, VII, and VIII.

Our proposed algorithm has been implemented with 10 agents on the six datasets and the condition for stopping each agent is one of the following two:

1) The number of repetitions will be completed
2) In each step, the value of the fitness function is less than the previous one

The results are shown in figure 5, figure 6, figure 7 and figure 11. According to figure 11, four communities have been discovered that some nodes belong to more than one community. The nodes that belong to more than one community have led to overlapping communities. As shown in figure 5 and 6, five nodes have been discovered as nodes that lead to overlapping communities. These five nodes are marked in figure 5, have most swiping in a variety of communities. The number of nodes is 3,10,9,27 and 30.

According to the heat map in figure 6, five nodes belong to four communities and these nodes led to the creation of overlapping communities, as shown in table IX, we will be able to use fuzzy approach to calculate the percentage of the membership of each node to the overlapping communities. So that, nodes of community zero are not a member of another community, therefore community zero is a non-overlapping community but another three communities are overlapping communities.

According to figure 7, maximum modularity is 0.419 and minimum modularity is 0.403 and average modularity is 0.409 and the median modularity is 0.406 in Zachary's karate club.

To measure the difference between the results of each of the agents, we use NMI evaluation metric, in NMI1 given that the assumption is that we have the grand truth of this network, but in NMI2 given that the assumption is that we do not have the grand truth, so we assign grand truth, to the agent result that has the highest modularity.

The results of which are shown in Table X. According to NMI2 in table X, we conclude that the results of the agents are approximately the same and that the nodes are moving between the discovered communities. One of the features of our proposed algorithm is that we can observe a node in multiple communities that are likely to be nodes in that community, and if the nodes were observed only in one community in all of the agents, that node could be moved to No other communities.

According to figure 8, we showed unstable NMI value in the agents' number 4,5,6 but shown the stable Modularity value in those agents, so we conclude that the modularity of the agents is approximately the same but NMI values are different, Therefore, the results of this experiment indicate that there is no direct relationship between NMI and Modularity.

Figure 9 gives an illustration of how the value of the modularity improves per each iteration in the agent. It shows that the repetition time is not completed because the value of the fitness function is not improved comparing the next iterations, so the repetitions were stopped.

According to figure 10, node 8 has been detected in two communities. When the result of agents indicate that a node belongs to more than one community, it can be a good candidate to be selected as an active node leading to the creation of overlapping communities.

To measure the performance of the proposed algorithm, we need to compare it with one of the most efficient community identification algorithms, Louvain, to detect the communities in the LFR-based dataset. The modularity values of the generated social networks are calculated based on the classification provided by ground truth in each of these datasets.

Given the network nodes class label as a community tag, we obtain the value of the NMI evaluation parameter for the Louvain algorithm as well as the FOCDMAPSO algorithm, to obtain a match of the network nodes community with what these two algorithms have detected.

The results of the calculations can be seen in the table below.

As can be seen in the table XI, according to the original cluster, the modularity value in the LFR datasets is -0.3275, -0.091, 0.0251, 0.2132, and 0.0907, respectively.

When we calculate, the modularity value based on the Louvain algorithm on the datasets will be 0.3721, 0.5202, 0.2291, 0.313, and 0.2730, respectively. However, for the proposed algorithm on the datasets, modularity value will be 0.3757, 0.5445, 0.3214, 0.3149, and 0.2759, respectively.

These results indicate the better performance of the proposed algorithm than to the Louvain algorithm. In the next evaluation, we use the NMI evaluation parameter to assess the extent to which the main community matches the community identified by the Louvain algorithms and the proposed algorithm. Note that, the efficiency of the algorithm in the case of non-overlapping communities was evaluated.

The results presented in the table XI show that both the NMI values and the Modularity values are in the proposed algorithm higher than the Louvain algorithm, so it can be concluded that performance of quality of community detection and the degree of matching with the main community of node in the proposed algorithm is better than Louvain algorithm.

V. CONCLUSION AND FUTURE WORK

The proposed algorithm in this research is based on the multi-agent particle swarm optimization. In artificial intelligence, an agent refers to an autonomous entity that acts, directing its activity towards achieving goals, upon an environment using observation through sensors and consequent actuators[40]. This method enables several simple particle swarms as the base components to regulate their behavior and relationships with the rest of the group. The algorithm uses a special type of coding to identify the number of communities without any prior knowledge. In this method, the modularity measure is used as a fitness function to optimize particle swarm. Several experiments show that the proposed algorithm, FOCDMAPSO outperforms four other competitive algorithms over six common datasets in the research literature.

The previous PSO algorithms for community detection in the literature only recognize non-overlapping communities, but this study suggests a combinatory mechanism that uses the collective intelligence of multi-agents to identify nodes in overlapping communities. Our experimental results indicate that once a node belongs to more than one community, this node is a good candidate to be selected as the active node, which leads to the creation of overlapping networks. Using this method, we can calculate the percentage of the membership of each node to the overlapping communities. When the number of agents increases, the execution time will be increased. This could be one of the limitations of the proposed.

Meta-heuristic optimization algorithms inspired by nature, are introduced by various researchers viz. Bat Algorithm [41-43], Cuckoo Search Algorithm [44, 45], Dragonfly Algorithm [46], Polar bear algorithm [47], and so forth. If these algorithms can be modeled discretely, they can find non-overlapping communities, and they can use the idea of this paper to find overlap communities.

In fact, in this research, it was found that each agent identifies the community label of a node; in the future, we want to show that the nodes with the largest number of community labels are identical and similar, and maybe more intimate with each other and may have more and closer connections. Furthermore, we will expand our research to identify nested communities, which is one of the main research challenges in social network analysis and also expand our research on parameter tuning in PSOs dynamically with a fuzzy system because the fuzzy system [37] avoids getting trapped in the local optima and it improves the performance of the PSO algorithm.

In fact, in this research, it was found that each agent identifies the community label of a node; in the future, we want to show that the nodes with the largest number of community labels are identical and similar, maybe more intimate with each other and may have more and closer connections.

REFERENCES

[1] Huang, Y., et al., *Survey on Fractality in Complex Networks*, in *Recent Developments in Intelligent Computing, Communication and Devices*. 2019, Springer. p. 675-692.

[2] Li, K., et al., *Analysis on topological features of deterministic hierarchical complex network.* Physica A: Statistical Mechanics and its Applications, 2019.

[3] Van Lierde, H., G. Chen, and T.W. Chow, *Scalable Spectral Clustering for Overlapping Community Detection in Large-Scale Networks.* IEEE Transactions on Knowledge and Data Engineering, 2019.

[4] Girvan, M. and M.E. Newman, *Community structure in social and biological networks.* Proceedings of the national academy of sciences, 2002. **99**(12): p. 7821-7826.

[5] Fortunato, S., *Community detection in graphs.* Physics reports, 2010. **486**(3-5): p. 75-174.

[6] Newman, M.E. and M. Girvan, *Finding and evaluating community structure in networks.* Physical review E, 2004. **69**(2): p. 026113.

[7] Shang, R., et al., *Community detection based on modularity and an improved genetic algorithm.* Physica A: Statistical Mechanics and its Applications, 2013. **392**(5): p. 1215-1231.

[8] Jin, D., et al., *Ant colony optimization with a new random walk model for community detection in complex networks.* Advances in Complex Systems, 2011. **14**(05): p. 795-815.

[9] Bandyopadhyay, S., et al., *A simulated annealing-based multiobjective optimization algorithm: AMOSA.* IEEE transactions on evolutionary computation, 2008. **12**(3): p. 269-283.

[10] Rahimi, S., A. Abdollahpouri, and P. Moradi, *A multi-objective particle swarm optimization algorithm for community detection in complex networks.* Swarm and Evolutionary Computation, 2018. **39**: p. 297-309.

[11] Wu, C., et al. *An improved PSO algorithm for community detection*. in *2015 10th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*. 2015. IEEE.

[12] Palla, G., et al., *Uncovering the overlapping community structure of complex networks in nature and society.* nature, 2005. **435**(7043): p. 814.

[13] Huang, F., et al., *Overlapping community detection for multimedia social networks.* IEEE Transactions on multimedia, 2017. **19**(8): p. 1881-1893.

[14] Yang, J. and J. Leskovec. *Community-affiliation graph model for overlapping network community detection*. in *2012 IEEE 12th International Conference on Data Mining*. 2012. IEEE.

[15] Yang, J. and J. Leskovec. *Overlapping community detection at scale: a nonnegative matrix factorization approach*. in *Proceedings of the sixth ACM international conference on Web search and data mining*. 2013. ACM.

[16] Eberhart, R. and J. Kennedy. *A new optimizer using particle swarm theory*. in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*. 1995. Ieee.

[17] Schutte, J.F., et al., *Evaluation of a particle swarm algorithm for biomechanical optimization.* Journal of Biomechanical Engineering, 2005. **127**(3): p. 465-474.

[18] Dellermann, D., et al., *Hybrid intelligence.* Business & Information Systems Engineering, 2019: p. 1-7.

[19] Alguliyev, R.M., R.M. Aliguliyev, and I.Y. Alakbarova, *Extraction of hidden social networks from wiki-environment involved in information conflict.* International Journal of Intelligent Systems and Applications (IJISA), 2016. **8**(2): p. 20-27.

[20] Blondel, V.D., et al., *Fast unfolding of communities in large networks.* Journal of statistical mechanics: theory and experiment, 2008. **2008**(10): p. P10008.

[21] Gong, M., et al., *Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition.* IEEE Transactions on evolutionary computation, 2013. **18**(1): p. 82-97.
[22] Li, Y., et al., *Overlapping community detection through an improved multi-objective quantum-behaved particle swarm optimization.* Journal of Heuristics, 2015. **21**(4): p. 549-575.
[23] Ahmad, R., et al. *A multi-agent based approach for particle swarm optimization*. in *2007 International Conference on Integration of Knowledge Intensive Multi-Agent Systems*. 2007. IEEE.
[24] Jiang, J., et al., *Understanding crowdsourcing systems from a multiagent perspective and approach.* ACM Transactions on Autonomous and Adaptive Systems (TAAS), 2018. **13**(2): p. 8.
[25] Ricciardi, L.A. and M. Vasile, *Improved archiving and search strategies for Multi Agent Collaborative Search*, in *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*. 2019, Springer. p. 435-455.
[26] Li, E., Q. Ma, and G. Zhou, *Bipartite output consensus for heterogeneous linear multi-agent systems with fully distributed protocol.* Journal of the Franklin Institute, 2019. **356**(5): p. 2870-2884.
[27] Zhang, C., et al., *A memetic particle swarm optimization algorithm for community detection in complex networks.* International Journal of Pattern Recognition and Artificial Intelligence, 2016. **30**(02): p. 1659003.
[28] Wang, G.-G., et al., *A hybrid method based on krill herd and quantum-behaved particle swarm optimization.* Neural Computing and Applications, 2016. **27**(4): p. 989-1006.
[29] Shahmoradi, M., et al., *Multilayer overlapping community detection using multi-objective optimization.* Future Generation Computer Systems, 2019.
[30] Danon, L., et al., *Comparing community structure identification.* Journal of Statistical Mechanics: Theory and Experiment, 2005. **2005**(09): p. P09008.
[31] Mishra, K.K., et al., *A direction aware particle swarm optimization with sensitive swarm leader.* Big data research, 2018. **14**: p. 57-67.
[32] Lusseau, D., et al., *The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations.* Behavioral Ecology and Sociobiology, 2003. **54**(4): p. 396-405.
[33] Cai, Q., et al., *Greedy discrete particle swarm optimization for large-scale social network clustering.* Information Sciences, 2015. **316**: p. 503-516.
[34] Hu, P., W. Fan, and S. Mei, *Identifying node importance in complex networks.* Physica A: Statistical Mechanics and its Applications, 2015. **429**: p. 169-176.
[35] Ghoshal, A.K., et al. *A Fast Parallel Genetic Algorithm Based Approach for Community Detection in Large Networks*. in *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*. 2019. IEEE.
[36] Lancichinetti, A., S. Fortunato, and F. Radicchi, *Benchmark graphs for testing community detection algorithms.* Physical review E, 2008. **78**(4): p. 046110.
[37] Valdez, F., et al., *Comparative study of the use of fuzzy logic in improving particle swarm optimization variants for mathematical functions using co-evolution.* Applied Soft Computing, 2017. **52**: p. 1070-1083.
[38] Zhang, L., et al., *A mixed representation-based multiobjective evolutionary algorithm for overlapping community detection.* IEEE Transactions on Cybernetics, 2017. **47**(9): p. 2703-2716.
[39] Wen, X., et al., *A maximal clique based multiobjective evolutionary algorithm for overlapping community detection.* IEEE Transactions on Evolutionary Computation, 2016. **21**(3): p. 363-377.
[40] Russell, S.J. and P. Norvig, *Artificial intelligence: a modern approach*. 2016: Malaysia; Pearson Education Limited.
[41] Song, A., et al., "*Community detection using discrete bat algorithm*", IAENG International Journal of Computer Science, vol. 43, no. 1, pp 37-43, 2016.
[42] Rani, S. and M. Mehrotra. *A Hybrid Bat Algorithm for Community Detection in Social Networks*. in *International Conference on Intelligent Systems Design and Applications*. 2018. Springer.
[43] Wang, C., W. Song, and L. Liu, "*An adaptive bat algorithm with memory for global optimization*", IAENG International Journal of Computer Science, vol.45, no. 2, pp 320-327, 2018.
[44] Zhou, X., Y. Liu, and B. Li, *A multi-objective discrete cuckoo search algorithm with local search for community detection in complex networks.* Modern Physics Letters B, 2016. **30**(07): p. 1650080.
[45] Li, S. and J. Wang, "*Improved cuckoo search algorithm with novel searching mechanism for solving unconstrained function optimization problem*", IAENG International Journal of Computer Science, vol. 44, no. 1, pp 8-12, 2017.
[46] Mirjalili, S., *Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems.* Neural Computing and Applications, 2016. **27**(4): p. 1053-1073.
[47] Połap, D., *Polar bear optimization algorithm: Meta-heuristic with fast population movement and dynamic birth and death mechanism.* Symmetry, 2017. **9**(10): p. 203.

Mohammad Akafan
Ph.D. Candidate, Department of Computer Engineering, Islamic Azad University, North Tehran Branch.
He researches on various areas in artificial intelligence and data mining, including text mining, web information extraction.
His research interests include Data Mining, Big Data, Social Network.

Behrouz Minaei-Bidgoli
Associate Professor in the Computer Engineering Department at Iran University of Science and Technology.
He leads the Data Mining Lab (DML) that researches various areas in artificial intelligence and data mining, including text mining, web information extraction, and natural language processing.
He also acts as the Director of Research for the Department. His research interests include Data Mining and Social Network.

Alireza Bagheri
received his B.S. and M.S. degrees in computer engineering from Sharif University of Technology (SUT) at Tehran. He received his Ph.D. degree in computer science from Amirkabir University of Technology (AUT) at Tehran.
Currently, he is an assistant professor in the computer engineering and IT department at Amirkabir University of Technology (AUT) at Tehran.
His research interests include computational geometry, graph drawing, and graph algorithms.

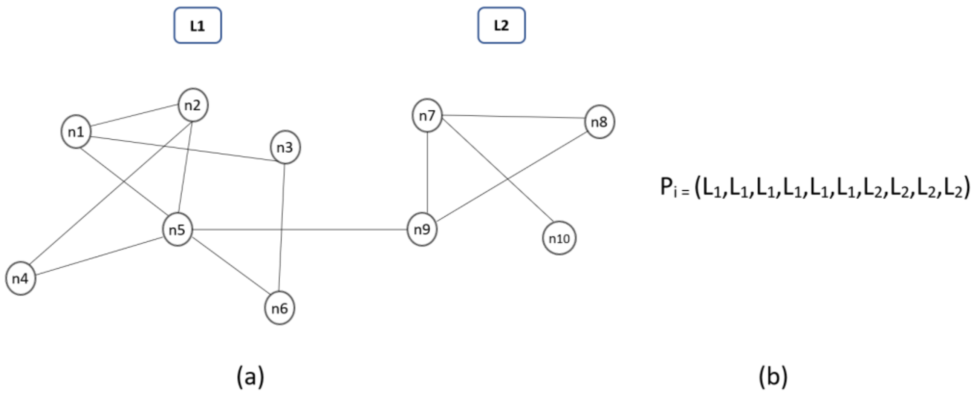

Fig 1: (a) Communities of a social network such that L_1, L_2 are labels of these communities, and n_i are the nodes of the network, (b) Particle coding and P_i is particle i .

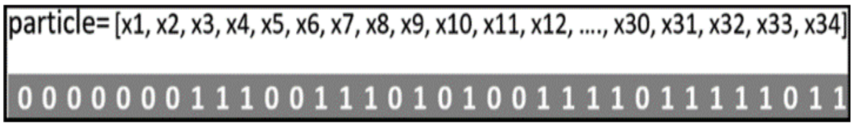

Fig 2: Particle code representation of the network.

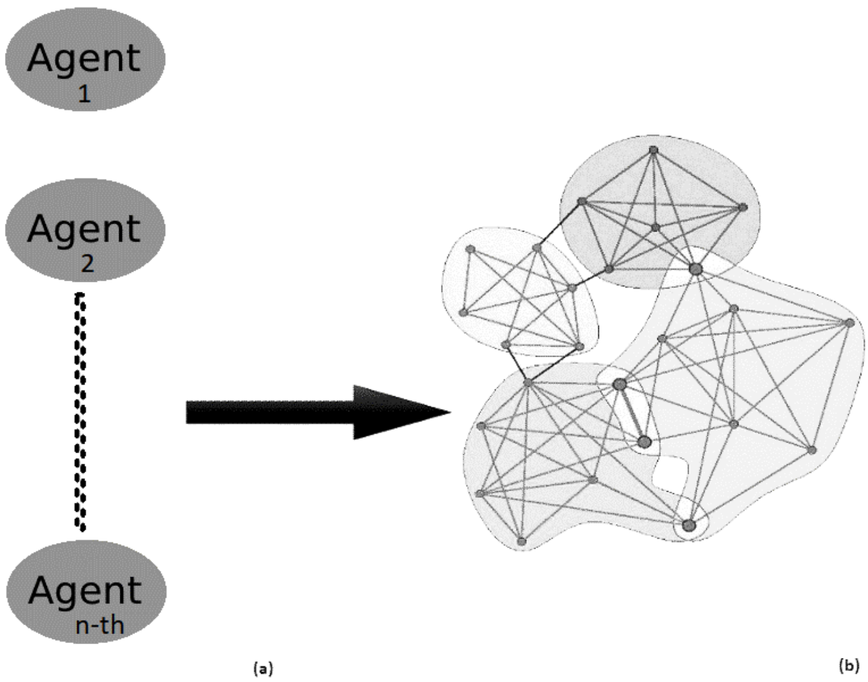

Fig 3: Identification of belongingness of each node to a given community in social networks by FOCDMAPSO algorithm.

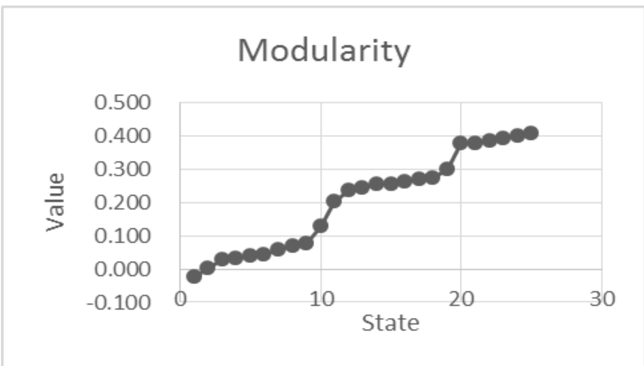

Fig 4: Experimental results to set parameters.

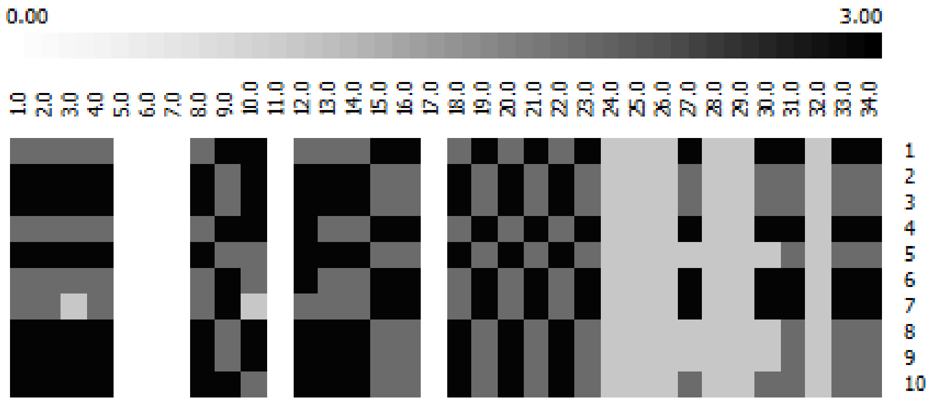

Fig 5: The heat map derived from the proposed method and detecting nodes that lead to the overlapping of the communities in Zachary's karate club. Row value is agent number and column value is node number.

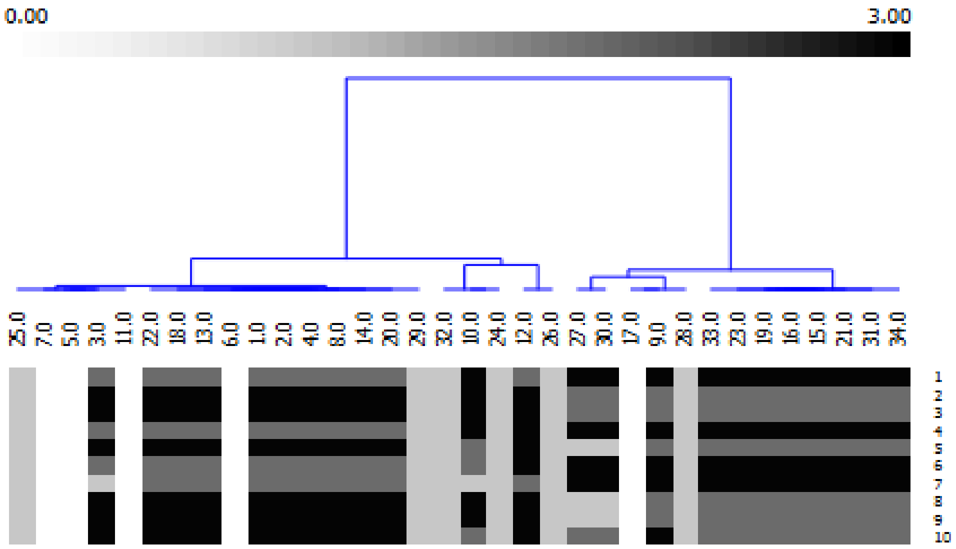

Fig 6: The heat map derived from the proposed method and detecting nodes that lead to the overlapping of the communities in Zachary's karate club, the heat map cluster by node community. Row value is agent number and column value is node number.

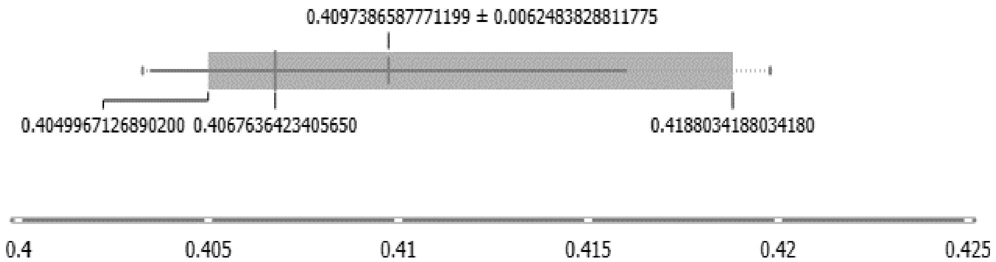

Fig 7: Box plot for modularity of this method in Zachary's karate club.

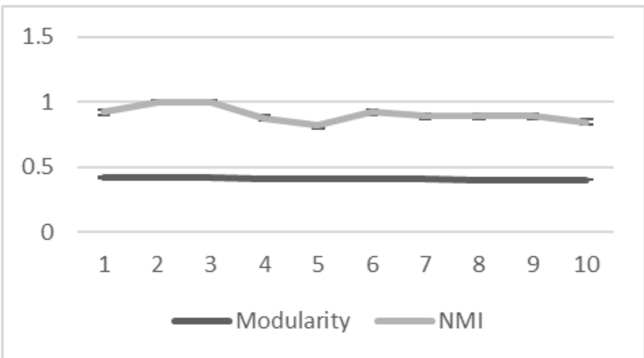

Fig 8. Compare between NMI value and Modularity value.

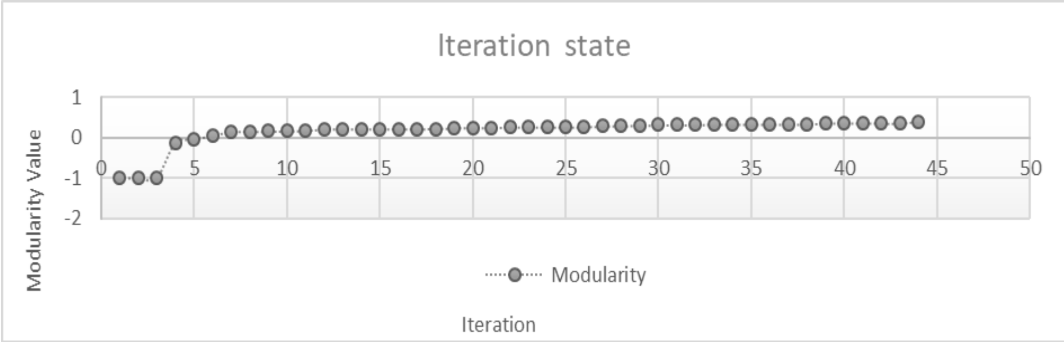

Fig 9: Improve modularity per each iteration.

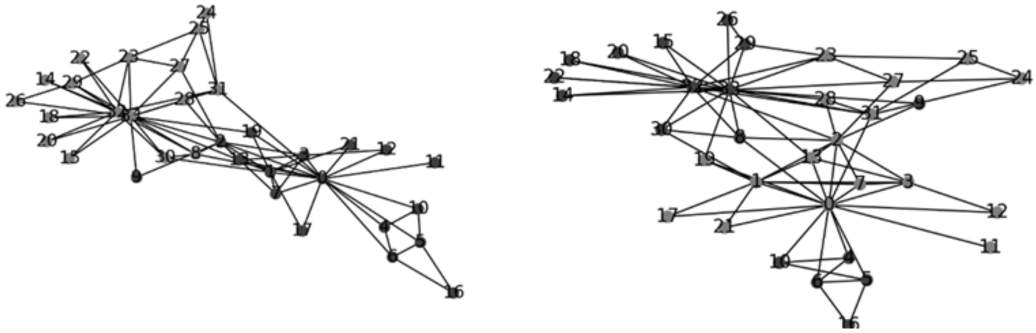

Fig 10: Graphs of agent 1 and 2 from left to right.

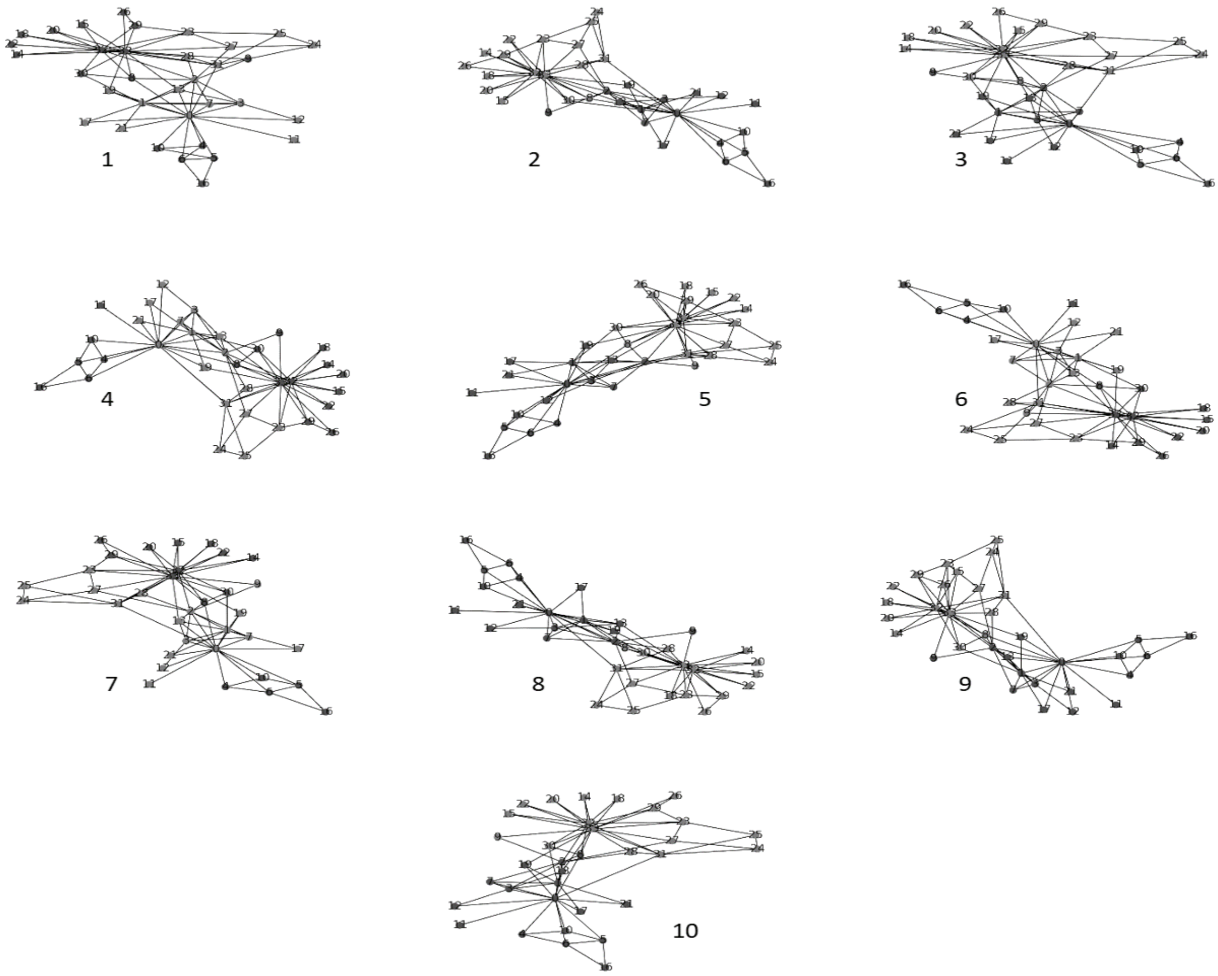

Fig 11: Graphs derived from the proposed method and detecting nodes that lead to the overlapping of the communities by agent number in Zachary's karate club, from 1 to 10, the modularity of agents decreases.

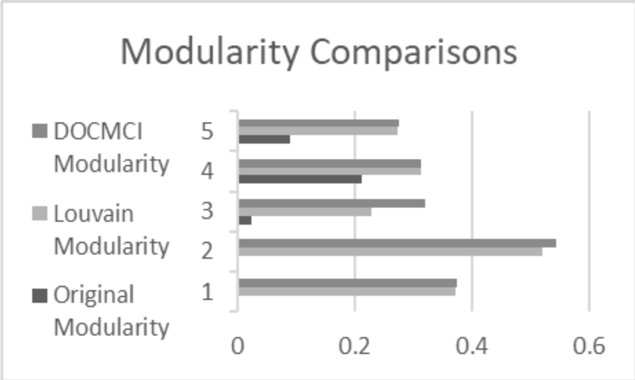

Fig 12: Modularity comparison between Louvain and FODMAPS clustering and original cluster.

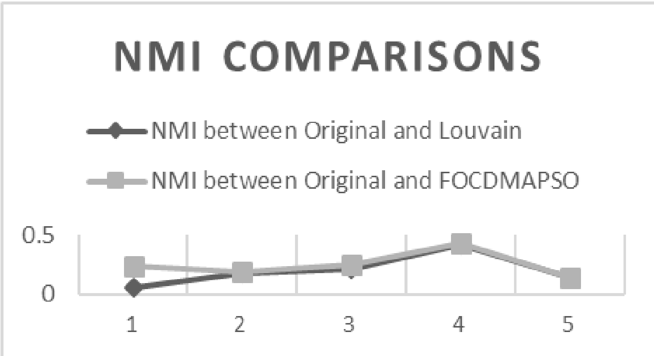

Fig 13: NMI comparison between Louvain and FOCDMAPSO algorithms.

TABLE I- DESCRIPTION OF REAL-WORLD NETWORKS.

Networks	#Nodes	#Edges	#Clusters	Description
Karate	34	78	2	Zachary's karate club
Dolphins	62	159	2	Dolphin social network
Football	115	613	12	American College football
SFI	118	200	Unknown	
Netscience	1589	2742	Unknown	
Power grid	4941	6594	Unknown	

TABLE II- DESCRIPTION OF LFR NETWORKS.

Dataset	Nodes	Edges	$£_1$	$£_2$	µ	Average-Degree	Ground truth cluster
1	34	87	3.09	2.67	0.92	6	3
2	62	104	3.65	5.95	0.82	7	6
3	115	800	5.93	5.97	0.71	14	16
4	118	589	4.34	2.95	0.40	10	7
5	1589	8806	4.98	2.45	0.71	10	80

TABLE III - SETTING OF THE ALGORITHM PER DATASETS.

State	c_1	c_2	w	Modularity
1	1	1	0.5	-0/025307692308
2	0	0	0.5	0/003209730440
3	1.5	1.5	0	0/028357659435
4	2	2	0.5	0/033206443130
5	0	0	0	0/038712689020
6	0.5	0.5	0.5	0/044054569362
7	1	1	0	0/057039447732
8	2	2	0	0/066901380671
9	0.5	0.5	0	0/077502958580
10	1.5	1.5	0.5	0/126730440500
11	0	0	1	0/204064431295
12	0	0	2	0/237019723866
13	0	0	1.5	0/243183431953
14	2	2	2	0/252963182117
15	1.5	1.5	2	0/252963182117
16	0.5	0.5	2	0/259948717949
17	0.5	0.5	1.5	0/268084812623
18	1	1	2	0/273837606838
19	1	1	1.5	0/297752794214
20	1	1	1	0/376976988823
21	0.5	0.5	1	0/377059171598
22	1.5	1.5	1	0/382894148586
23	1.5	1.5	1.5	0/392838264300
24	2	2	1	0/399084155161
25	2	2	1.5	0/405864562788

TABLE IV - SETTING OF THE ALGORITHM PER DATASETS.

Parameters	Dataset	Zachary's karate club	Dolphin social network	American College football	SFI Network	Netscience Network	Power grid Network
Particle Size		34	62	115	118	1589	4941
Initial Population		200	200	200	200	200	200
Iteration		100	100	100	100	100	100
w		1.5	1.5	1.5	1.5	1.5	1.5
c_1		2	2	2	2	2	2
c_2		2	2	2	2	2	2

TABLE V– COMPARE MR-MOEA [38].

Algorithm	MR-MOEA	FOCDMAPSO	
Dataset	Modularity	Community Detected#	Modularity(max)
American College football	0.306	12	0.456
Zachary's karate club	0.229	4	0.419
Dolphin social network	0.271	4	0.485

TABLE VI – COMPARE OF MODULARITY OF IMOQPSO [22] AND FOCDMAPSO.

Algorithm	Modularity	
Dataset	IMOQPSO	FOCDMAPSO (max)
American College football	0.243	0.456
Zachary's karate club	0.213	0.419
Dolphin social network	0.264	0.485

TABLE VII – COMPARE OF MODULARITY OF MCMOEA[39] AND FOCDMAPSO.

Algorithm	Modularity	
Dataset	MCMOEA	FOCDMAPSO (max)
American College football	0.279	0.456
Zachary's karate club	0.210	0.419
Dolphin social network	0.206	0.485

TABLE VIII– COMPARE OF MODULARITY OF MODPSO [21] AND FOCDMAPSO.

Algorithm	Modularity	
Dataset	MODPSO	FOCDMAPSO (max)
SFI	0.748	0.752
Netscience	0.950	0.951
power grid	0.829	0.831

TABLE IX: FUZZY APPROACH TO CALCULATE THE PERCENTAGE OF THE MEMBERSHIP OF EACH NODE TO THE OVERLAPPING COMMUNITIES FOR ZACHARY'S KARATE CLUB DATASET.

Node	Community				Percentage			
	0	1	2	3	0	1	2	3
3	0	1	3	6	0%	10%	30%	60%
9	0	0	5	5	0%	0%	50%	50%
10	0	1	2	7	0%	10%	20%	70%
27	0	3	3	4	0%	30%	30%	40%
30	0	3	3	4	0%	30%	30%	40%

TABLE X. USE NMI EVALUATION METRIC FOR DIFFERENCES BETWEEN THE RESULT OF EACH OF THE AGENTS IN ZACHARY'S KARATE CLUB DATASET.

Agent	Modularity	NMI1	NMI2
1	0.419	0.923	1.0
2	0.418	1	0.923
3	0.418	1	0.923
4	0.407	0.877	0.923
5	0.406	0.819	0.890
6	0.406	0.923	0.848
7	0.405	0.890	0.860
8	0.404	0.892	0.816
9	0.404	0.892	0.816
10	0.403	0.848	0.923

TABLE XI. COMPARISON BETWEEN ORIGINAL CLUSTER AND LOUVAIN CLUSTER AND FOCDMAPSO CLUSTER.

Dataset	1	2	3	4	5
Nodes	34	62	115	118	1589
Edge	86	104	800	589	8808
Original cluster	3	6	16	7	80
Original Modularity	- 0.3275	- 0.091	0.0251	0.2132	0.0907
Louvain Cluster	4	8	7	6	14
Louvain Modularity	0.3721	0.5202	0.2291	0.3138	0.2730
FOCDMAPS O Cluster	4	10	12	12	14
FOCDMAPSO Modularity	0.3757	0.5445	0.3214	0.3149	0.2759
Louvain NMI between Original and	0.055	0.1762	0.2134	0.4131	0.1399
NMI between Original and FOCDMAPSO	0.230	0.1846	0.2432	0.4291	0.1419