A LOCALLY AND GLOBALLY TUNED METAHEURISTIC OPTIMIZATION FOR OVERLAPPING COMMUNITY DETECTION

***Chandrakant Mallick*[1]*, Parimal Kumar Giri*[2*]*, Sarojananda Mishra*[3]**

[1]Department of Computer Science & Engineering, Biju Patnaik University of Technology,
Rourkela-769015, India

[2]Department of Computer Science & IT, GITA Autonomous College, Bhubaneswar,
Odisha-752054, India

[3]Department of Computer Science Engineering & Application, Indira Gandhi Institute of Technology,
Sarang, Orissa-759146, India

Email: ckmallick@gmail.com[1], parimal.6789@gmail.com[2*](corresponding author), sarose.mishra@gmail.com[3]

DOI: https://doi.org/10.22452/mjcs.vol36no2.4

ABSTRACT

Many people use online social networks to share their opinions and information in this digital age. The number of people engaged and their dynamic nature pose a major challenge for social network analysis (SNA). Community detection is one of the most critical and fascinating issues in social network analysis. Researchers frequently employ node features and topological structures to recognize important and meaningful performance in order to locate non-overlapping communities. We introduce a locally and globally tuned multi-objective biogeography-based optimization (LGMBBO) technique in this research for detecting overlapping communities based on the number of connections and node similarity. Four real- world online social network datasets were used in the experiment to assess the quality of both overlapping and non-overlapping partitions. As a result, the model generates a set of solutions that have the best topological structure of a network with node properties. The suggested model will increase their productivity and enhance their abilities to identify significant and pertinent communities.

Keywords: Community Detection, Biogeography-based Optimization, Topological Structure, Social Networks, multi-objective Optimization, Node Attributes, evolutionary algorithm.

1.0 INTRODUCTION

Millions of people around the world are connected through social networks such as Facebook, Google+, Twitter, and others, thanks to the rapid advancement of technology and the growing popularity of online social networks. A social network is commonly represented as a directed graph G = (V, E), with V denoting nodes (users) and E denoting edges between nodes. A vertex in this model represents an individual, whereas an edge represents a link between two individuals. Nodes in online social networks can impact each other by interacting with one another. As a result, nodes can be active or inactive. The active node is one that receives and extends an idea or piece of information to its neighbors'; otherwise, the node is inactive. The participation of frequent users, as well as the resulting changes in the structure and character of networks, Allows for social network analysis (SNA) on the internet [1]. SNA aims to comprehend the structure and participants of a dynamic social network. The entire networks structure is learned through the community identification method in social network analysis. A community can be defined as a group of members who share similar viewpoints.

Finding meaningful social groupings in online networks is an important problem that has drawn the attention of a number of scholars. In cyberspace, a community can be defined as a social network containing one or more groups of people as members. The important challenges of online social networks include locating communities, significant actors, and forecasting relationships or qualities [2]. Moreover, communities can be classified as group-centric, node-centric, hierarchy-centric, and network-centric. In general, major community detection techniques can be classified as local structure-based methods; approach based on graphs, and constructed tree-based methods [3].

As a result, the majority of study is focused on this topic and has concentrated on developing a range of approaches for non-overlapping community detection, where each node tends to just a single community. There are numerous real-world networks, on the other hand, that have communities that frequently overlap to some extent. Because nodes in these networks may play different functions in the network, they may belong to more than one community. As a result, the majority of community detection research has relied on social network graph architectures to discover communities [4]. However, in many real-world social networks, nodes with similar qualities are more likely within the same communities. Today's online social networks save a wide range of information, including shared items, comments, as well as traits like age, education, gender, and occupation [5]. As a result, the process of community discovery can be improved by taking into account the contents of social networks.

On the contrary, a social network with dynamic character, as well as the association of a large number of users, increases the accuracy of the community detection process. Divy and Ramalakshmi [6] have developed a novel hybrid machine learning strategy for detecting communities and overlapping communities using the particle swarm optimization algorithm. Xu et al. [7] suggested a discrete cuckoo search-based multi-objective community detection algorithm, where the location of nodes is detected and is performed by combined with non-dominated sorting and the crowd distance method. Link strength was considered as a significant function in an effective community detection model using graph based mining technology [8].

With reference to Bringmann et al. [2], most of the community detection in social networks is based on cluster density. However, because limited metrics are only examined for overlapping community detection, this strategy is unable to attain higher accuracy. Furthermore, overlapping community detection (OCD) has been brought up in the literature, but no content analysis has been carried out [4]. The modularity score is important in OCD, whereas the OCD problem with the structure of node characteristics remains unsolved. Fuzzy clustering was used to implement fuzzy semantic overlapping community detection (FuSeO) approach [9]. Furthermore, several heuristic techniques to solve the OCD problem have been presented in recent years due to the high efficiency of metaheuristic algorithms. Using cat swarm optimization, genetic algorithm, and simulated annealing models, Sarswat et al. [10] has presented a two-step hybrid detection strategy for overlapping communities. Rees et al. [11] presented the graph swarm optimization technique, which consists of three steps: group discovery, non- propagating node identification, and propagation of assigned group IDs. This technique, however, is unable to take into account important informational and structural indicators.

Researchers mainly concentrate on the investigation of social networks and the discovery of models and patterns using machine learning [12]. Evolutionary approaches, on the other hand, are extremely fast and efficient in solving large-scale optimization issues. Simon et al. [13] shows that BBO works better as compared to other EAs. He has signified BBO algorithms for gaining certain parameters to tune-up and it has proven itself a competitive heuristic on a large set of EAs. One of the successful optimization algorithms, biogeography-based optimization (BBO), offers a number of advantages [13]. Recently, the applicability of BBO to SNA was increased [14]. The results of the investigation reveal that the BBO algorithm performs effectively in SNA for a variety of applications. It can detect overlapping communities automatically, taking into account the number of connections and the similarity of nodes. Because attribute similarity and link density can be thought of as separate and sometimes contradictory factors [4].

Usually, the traditional multi-objective evolutionary algorithms (MOEAs) do not meet the requirement for online data processing because of their high computational costs [15]. In this study, we propose a novel metaheuristic method, locally and globally tuned multi-objective BBO, for the overlapping community discovery problem, taking into account the number of connections and the similarity of nodes. In order to attain a balance, evolutionary strategies can be applied. In terms of the topological structure (i.e., connection density) of a network with node attributes, the suggested method's result analysis produces a group of solutions with the best performance.

The remainder of the paper is outlined as follows: Section 2 introduces the related works. The research's historical context is discussed in Section 3. Section 4 presents our proposed method and explains it in detail. Extensive experiments are carried out on four real-life data sets in order to evaluate the performance. Section 5 contains a description of the data sets, as well as the outcomes of the experiment and their interpretation. Finally, Section 6 concludes the paper.

2.0 RELATED WORK

This section delves into the previous academics' concerns about detecting community and overlapping communities in online social networks.

Online social networks (OSNs) have expanded into a major platform for people's daily contact and information sharing in recent years, thanks to rapid advancements in Internet technology and widespread adoption of smart mobile terminals. Simultaneously, social network analysis, as a projection of real-world human social ties, contains a significant amount of data. Many works use OSNs to assess public opinion, monitor emergencies, and track social interactions, among other things. Community detection work is one of the most essential studies among them. Some researchers have attempted to improve the greedy technique by developing a more efficient variant. The cost effective lazy forward (CELF) algorithm was introduced by Leskovec et al. [16]. It enhances the greedy method by introducing the sub-modularity property into the node selection procedure. Furthermore, the majority of research focuses on the network's topological features or linking patterns [4]. Non-overlapping (separated or disconnected) and overlapping community identification methods are the two types of community detection methods. The most important studies about optimum modularity to assess the partition's quality have been introduced by Newman et al. [17]. In their paper, Palla et al. [18] looked at the statistical characteristics of overlapping communities.

A innovative method for identifying the smallest and most precise multi-profiled cross community to which an active user belongs was introduced by Taha et al. [44]. A link partition was used by Xia et al. [19] to show how the community structure overlapped. DeMeo et al. [20] demonstrated COPRA, very large networks, an approach for finding overlapping community structures has been developed. Zhang et al. [21] has presented a unique model called OSLOM model that can locate edge orientations, edge weights, overlapping communities, hierarchies, and community dynamics all contribute to network clustering. Atzmueller et al. [22] presented the model SLPA dynamic interaction process, which allows for the detection of efficient and effective overlapping communities.
As more and more complicated data from the real world are represented in graphs with attributed nodes, attributed graph clustering (AGC) has become a significant issue in graph mining. Kamhoua et al. [45] suggested an expanded general graph convolution framework for attributed graph clustering (GRACE), often known as an AGC tasks. Zhang et al. [21] suggested a community affiliation graph model that can capture overlapping, non-overlapping, and hierarchical layered overlapping communities successfully. Rather of assuming an a priori correlation structure between qualities and communities, Contisciani et al. [43] suggested a probabilistic technique that infers this from data rather than making any assumptions about it.

Furthermore, several heuristic techniques to solve the community detection problem have been developed in recent years because of the high efficiency of metaheuristic algorithms. In order to solve community detection issues in a reasonable length of time, Zhang et al. [21] presented clustering optimization and swarm-intelligence-based algorithms. Many single-objective evolutionary algorithms, as well as multi-objective ones, have been used to solve the community detection problem in complex social networks in recent years, according to Pizzuti et al. [23]. To locate non-overlapping social network groups with node attributes, Reihanian et al. [14] developed a multi-objective discrete BBO model. Similarly, in order to tackle the challenge of attributed graph clustering, Zhu et al. [24] developed MOEA-SA, a multi-objective evolutionary method based on similarity in structure and attributes. A multi-objective BBO-based overlapping community detection algorithm (MOBBO-OCD) has been proposed by Reihanian et al. [25] to automatically identify overlapping communities in a social network with node attributes while taking into account both the density of connections and the similarity of nodes' attributes in the network.

It is clear from the preceding literature that most of the methods presented are only capable of finding disjoint communities and not overlapping ones. In a real graph, however, the number of communities is usually unknown ahead of time [24]. The key difference between community and overlapping community is demonstrated in Fig. 1. To address this issue, this research proposes a unique community recognition algorithm that can automatically detect overlapping communities while taking into account in social networks, the density of connections and the similarity of nodes are important factors.

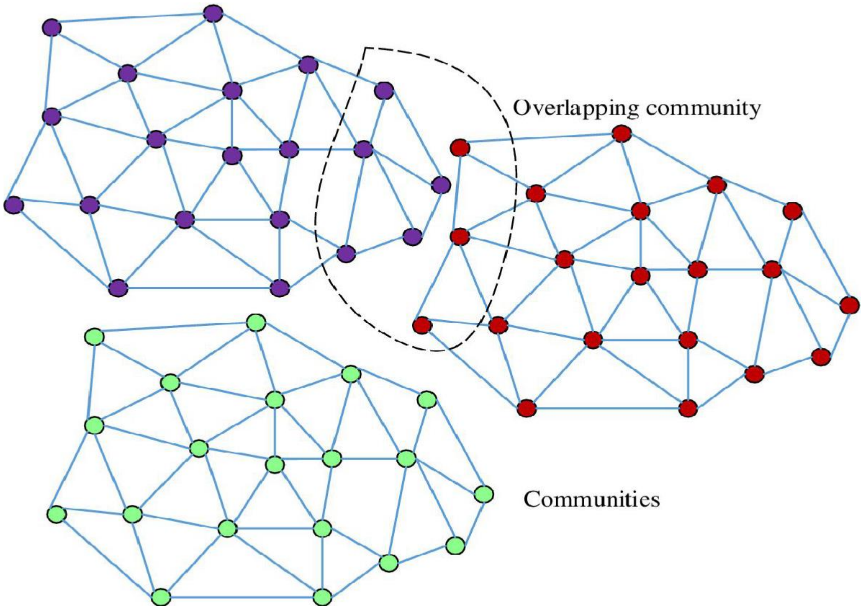

Fig. 1: Illustration of overlapping community (Divy and Ramalakshmi [6])

3.0 BACKGROUND

In this section, we'll go through a novel multi-objective biogeography-based optimization (BBO) technique for detecting overlapping communities in a social network.

3.1 Biogeography-based Optimization

The study of how species migrate from one island or environment to another, as well as how new species originate and disappear is known as biogeography. A habitat is a geographically separated island from the rest of the ecosystem. A habitat with a high suitability index is geographically well adapted for the species that live there. Simon et al. [13] presented Biogeography-based Optimization (BBO), a metaheuristic algorithm biologically inspired that employs the idea of animal or other species migration strategy to solve complicated global optimization issues. The BBO algorithm is quick and gives the best results by mimicking natural patterns such as species distribution. Candidates are referred to as habitats in BBO, and the fitness value is referred to as the habitat suitability index (HSI). Moreover, all habitats have their own probabilistic operators which are proportional to HSI called an emigration (μ) and an immigration rate (λ). For the next generation solution, migration operators refer to the immigration and emigration of species within a habitat.

In Fig. 2, S_1 denotes a low HSI solution, whereas S_2 represents a high HSI solution. As a result, λ_1 denote immigration rate for S_1 will be higher than λ_2 immigration rate S_2. Similarly, for S_1, the μ_1 emigration rate will be lower than the μ_2 emigration rate.

Furthermore, in BBO the migration process is a probabilistic operator. This can update each habitat H_i by applying suitability index variables (SIVs) from higher HSI habitats [26]. The general migration model can be defined as in Eq. 1. In general, the emigration and immigration rates can be evaluated by using Eq. 2, where k is the number of species in the island.

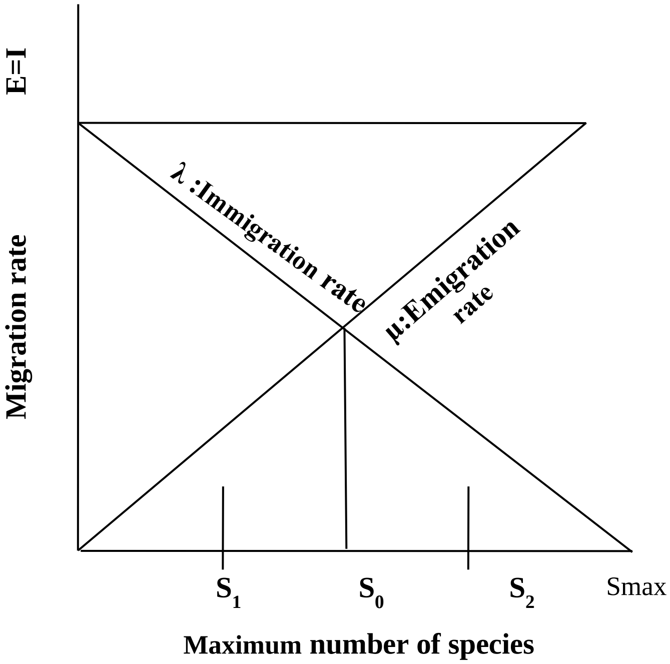

Fig. 2: The S_1 candidate solution is a relatively bad solution, while the S_2 candidate solution is a comparatively good one Simon et al. (2008).

$$H_i(SIV) \leftarrow H_j(SIV) \tag{1}$$

$$\mu_k = \frac{E_k}{S_{max}} \text{ and } \lambda_k = I(1-\mu_k) \tag{2}$$

where E and I are the maximum emigration and immigration rates, respectively. The maximum number of species that the habitat can host is S_{max}.
Similarly, given prior probability of existence Pi and the mutation probability is used to modify one or more randomly selected SIVs of a solution. This operator alters the SIVs in the habitat at random based on the mutation rate mi, which may be determined using Eq. 3.

$$m_i = M_{max}\left(1 - \frac{P_i}{P_{max}}\right) \tag{3}$$

where M_{max} as the maximum mutation rate and m_i are user-defined parameters. Similarly, P_{max} represents the highest probability of a species count.

According to the literature review, researchers offered numerous BBO migration models with noteworthy performance outcomes. They do, however, have their own set of advantages and disadvantages. BBO's success is based on global exploration [13], but it can become caught in local optima and hence be unable to perform the global search. It's difficult to strike a balance between exploration and exploitation because of the low population diversity and convergence rate. An extensive study of BBO algorithms can be found in [28].

To avoid some pitfalls of the existing BBO, Giri et al. [27] have proposed a novel modified BBO algorithm known as locally and globally tuned BBO (LGBBO) that outperforms some variants of original algorithms. The proposed LGBBO inherits attributes from the local best students nearest neighbor, which are then transferred to the pool's globally best individual. The concept is based on a student in a lower class's learning mechanism. In nature, it is frequently observed that a weaker student is influenced directly by a student who is better in a local setting rather than a global context. In other words, instead of adapting qualities from the top individual in the pool, a weak individual tries to adapt the finest features from his or her nearest neighbor (i. e., global best). The model is presented in Eq. 4 as:

$$H_i(SIV) \leftarrow \alpha \, NN\big(H_i(SIV)\big) + (1-\alpha)H_j(SIV) \tag{4}$$

The parameter α is known as a geographical system's immaturity index [28]. This is inversely related to the system's invasion resistance. As a result, the habitats nearest neighbor NN (H_i (SIV)) can be specified in Eq. 5 as:

$$NN\big(H_i(SIV)\big) \leftarrow H_{(i\leq r?1:i-r)}(SIV) \tag{5}$$

where r is the neighborhood's radius. The best nearest neighbor habitat may be discovered at $(i-r)$ since the HSIs are sorted in a certain way. When $i \leq r$, the best habitat has been selected as the best locally.

As compared to various evolutionary algorithms like genetic algorithm (GA), particle swarm optimization (PSO), ant colony optimization (ACO), and differential evolution (DE), BBO requires information sharing between solutions [28]. As a result, BBO is applicable to a wide range of high-dimensional data applications. The justification for using LGBBO in the suggested technique (MLGBBO) is one of the nature-inspired algorithms for tackling challenges in the domain of community detection. The benefits come in the form of ease of use, flexibility, and computing efficiency. De et al. [29] proposed a set of statistical techniques and locally globally tuned biogeography-based optimization are used to identify end vertices for the links recommendation.

3.2 Multi-objective Optimization Problem

The objective of optimization problems is to determine the optimum values for decision variables that achieve the objective while staying within limitations. The majority of real- world problems involve multiple objectives that must be maximized and minimized at the same time, and they are often incompatible and equally significant [30]. Finding a solution that is acceptable for one goal may be incompatible with other goals. As a result, the aim of a multi-objective optimization problem is to find the optimum compromise between all of the objective functions. Traditional multi-objective evolutionary optimization algorithms (MOEAs) do not frequently meet the requirements for online data processing because of their high computational costs. In multi-objective situations, evolutionary algorithms can be effective. Multi-objective evolutionary algorithms can perform well on multi-objective problems to get the optimum value of each objective simultaneously. The optimal trade-off solutions are those in which no additional improvement in any objective function is feasible without surrendering any objectives. Pareto optimum solutions denote these types of the trade-off solutions [31].

Many multi-objective evolutionary algorithms have been proposed in recent decades. The capacity to generate several Pareto optimum solutions in a single run is the major reason. In this research, differential evolution-based non-dominated sorting algorithms such as non-dominated sorting genetic algorithms (NSGA) are popular. For multi-objective optimization, NSGA-II is a prominent non-domination-based genetic algorithm [32]. Non-dominated sorting genetic algorithm the third version (NSGA-III) is a relatively new evolutionary algorithm that may solve complex optimization issues with reasonable computational demands [15]. Similar to NSGA-III, it does not perform well when addressing problems with a large number of objectives. Gu and Wang [33] have introduced the information feedback models to improve the ability of NSGA-III to solve large-scale optimization problems.

As a result, the following is a definition of multi-objective optimization:

Definition1: Assuming that m objectives to be minimize, thus a multi-objective optimization problem is of the form:
Optimize (maximize or minimize) the objective function F(x) = {f_1(x), f_2(x), f_3(x), ..., f_n(x)}.
Subject to condition x ∈ S
where S is the decision space. Let $x = x_{1,}x_2\ldots,x_n \in S$ is n-dimensional decision variable vector; $y = y_{1,}y_2\ldots,y_m \in$ Y is an objective of a m-dimensional vector and Y is an objective space.
Dominance and Pareto-optimality are two fundamental ideas connected to multi- objective solutions and we shall present them in a formal definition such as:

Definition2: A decision vector s ∈ S is said to be strictly dominate with respect to another decision vector t ∈ S denoted by $s \propto t$ if and only if ∀i ∈ {1, 2, 3, , m} :

fi(u) ≤ fj(t), and ∃j ∈ {1, 2, 3, , m} : fi(u) < fj(t)

Definition3: A decision variable s ∈ S is said to be dominant with respect to another decision vector t ∈ S (denoted $s \propto t$) iff the decision vector s is not worse than t in all objectives, that is fi(u) ≤ fj(t), and ∀i ∈ {1, 2, 3, , m}.

If either of the preceding requirements is broken, the solution s does not take precedence over the solution t. In general, several multi-objective optimization (MO) techniques have been published in the literature to solve real-world issues involving several objects that must be optimized at the same time, such as Non-Dominated Sorting Genetic Algorithm-II (NSGA-II), Vector Evaluated Genetic Algorithm (VEGA), Multiple Objective Genetic Algorithm (MOGA), Pareto-Archived Evolution Strategy (PAES), Pareto Envelope-based Selection Algorithm-II (PESA-II) [34]. Many solutions in multi-objective optimization problems are dominated in real-world scenarios, as proven.

Furthermore, the Pareto-based technique does not produce a unique answer. The most essential benefit of the Pareto-based method is that it presents a decision-maker with several alternative solutions. It is his or her obligation to choose the finest viable solution from a group of non-dominated candidate solutions that are all considered equally ideal [12]. In multi-objective space, the selection is effectively a trade-off between one complete solution x and another. Xu et al. [7] suggested a multi-objective cuckoo search- based community detection approach that combines non-dominated sorting and the crowd distance method to discover communities. Multi-objective evolutionary approaches solve large-dimensional optimization problems quickly and efficiently [32].

4.0 PROPOSED WORK

Our goal in this section is to talk about how overlapping communities with node properties in a social network can share comparable attributes. When the node attributes are known, a multi-objective LGBBO-based overlapping community discovery (LGMBBO) algorithm can be proposed to achieve this goal. In addition, the suggested approach is used to identify overlapping communities by taking into account two factors: topological structure and node properties. The majority of methods for detecting communities and overlapping communities rely on a single examination procedure that is neither informative nor structural. However, because the data created by social networks is big in volume and complicated in structure, a single reviewing procedure is not adequate for large social networks. Furthermore, numerous research studies detect either a community or an overlapping community, but none of the methods can detect both at the same time. As a result, by maximizing the objective functions, we can solve the problem of community detection individually, and the method's output could be a set of solutions. The end result of LGMBBO's Pareto-based technique in social network, a set of non-dominated solutions (partitions) to its input input. We've used each solution to represent a particular habitat in our research. In Algorithm 1, the pseudo-code of LGMBBO is shown as follows.

The input of LGMBBO, according to Algorithm 1, is a social network graph with NG = < N, A, E >, where N, A, and E denote nodes of this network, their accompanying attribute values, and graph of networks, edges, respectively. This adjacency matrix and node properties of the social network are contained in NG, which is the goal of this research work.

4.1 Social Network Data Modeling

The community discovery method was used to locate communities using a network graph generated from Twitter data. Outliers are nodes that are odd or noise, and they can be found in communities built from any source of data. We pre-processed social network data sets to remove impurities at first. The data sets are examined and the relationships between their nodes are assessed in the form of an adjacency matrix. After that, if any nodes in the data set network include some properties, they are examined and added in the network's final format. As a result, the data set could be analyses and some of the node's properties extracted.

Algorithm 1: Proposed LGMBBO Algorithm for Overlapping Community Detection

```
Initialize: E = I = 1, mmax = 1, Population size (Np), and Maximum iteration;
```

```
Generate initial habitats ($H1, . . . , H_{Np}$) in algorithm 2 and store them in the set Sh ;
Using the fitness functions, compute HSI values for each habitat from the collection Sh.;
Sort the Sh set;
while Termination criteria is not satisfied do
    Select HSI (fitness);
    Evaluate $\lambda$, $\mu, p_{mut}$ and mutation rate $m_i$;
    Make a random number rand $\in$ (0, 1);
    for each habitat is ranked from best to worst, according to their HSI values do
        Select a habitat $H_i(SIV)$ probabilistically $\propto \lambda_i$ ;
        If rand < $\lambda_i$ and $H_i(SIV)$ are selected then
            Select $H_k(SIV)$ as locally best to $H_i$ (SIV) using NN ($H_i$ (SIV));
            Select habitat $H_j(SIV)$ probabilistically $\propto \mu_j$;
            If rand < $\mu_j$ and $H_j$ (SIV) selected then make a constant $\alpha \in [0, 1]$;
                for each SIVs (solution features) do
                    $H_i(SIV)$ □ $\alpha$ NN ($H_i(SIV)$) + (1- $\alpha$)$H_j(SIV)$;
                end
            end
    end
    Select an $H_i(SIV)$ based on a proportionate mutation probability $p_i$;
    if NA is unweighted then
        CLS □ Decode $H_i(SIV)$ in order to obtain a cluster label for its SIV's;
        NN ($H_i(SIV)$) □ Find all the neighbors of NA node i;
        l □ Find a CLS cluster to which the majority of neighbors from Neighbors to;
        r □ Choose one neighbor at random from the Cluster Labeled  Neighbors l;
        $H_i(SIV)$ = r
        end
    else   if NA is weighted then
        NN ($H_i(SIV)$) □ Find all of the node k's        neighbors of NA;
        NN ($H_k(SIV)$)□ In Neighbors, find the neighbors who has the most powerful   connections to node k
        (highest edge weight);
        r □Select one of your neighbors at random from the Neighbors list ;
        $H_i(SIV)$ = r
        end
    end
```

4.2 Habitat representation

The suggested method generates a solution for each created habitat, which generates a network partition. Each solution (or person in a population) in this network is made up of N variables, where N is the number of nodes in the related network. The adjacency matrix of the network may be used to infer a set of probable values for each of these N variables. Each habitat is encoded using a locus-based adjacency representation in this section Zhou et al. (2015). The network's adjacent matrix can be used to identify the possible values for each of these n variables. The nodes of the network are represented by the adjacency representation based on loci, variables, and their values in general. Following the encoding of the solutions, a decoding process is required to determine all of the groups that each habitat generates. In this stage, all nodes belonging to the same attribute are denoted as the same cluster.

4.3 Initialization

The proposed method generates the initial population during its startup procedure. For this purpose of solving the community detection problem, the suggested solution uses independent initialization methods for both the weighted and unweighted networks. Here the proposed method's initialization is shown in Algorithm 2.

Algorithm 2. Initialization of Habitats (a network having properties of node NA)

```
    if NA is unweighted then
```

```
        for i is 1 to Np;
            for j is 1 to N;
                Neighbors evaluate node j’s neighbors [from the NA adjacency matrix];
      r □Choose one of your neighbors at random from the Neighbors list ;
         $H_j(j)$ = r;
         Return;
    end
    else if NA is weighed then
             for i is 1 to N
               for j is 1 to N ;
               Evaluate node j’s neighbors;
             p(neighbors)   Using Eq. 6, compute each neighbor’s chance of being chosen. ;
              r□Choose one neighbor from list of neighbors depending on the   probability   p(neighbors)
             (through the use of a roulette wheel);
               $H_j(j)$ = r;
               Return H1, H2… HNp;
     end
```

The value of the jth variable in Hi(j) is chosen at random from the network’s during the initialization process of the unweighted graph. The neighbors’ of node j in the network have a connection to∈node k, may defined as neighbors’ (j) = {k ∈ V, A_{jk}=1}, where V is the collection of nodes and A is the network’s adjacency matrix. After that, depending on the connection weight between node j and each of its neighbors’ in Eq. 6, for each of node j’s neighbor, the probability of selection is calculated as follows:

$$P\left(N_j(k)\right) = \frac{A_{ij}}{\sum_{i\in N(j)} \quad A_{jk}} \ , k \in \ neighbour(j) \tag{6}$$

where N_j(k) stands for neighbors’ of k for node j, and A_{jk} and A_{jl} stand for elements of a weighted network’s adjacency matrix. They are all integers. The weight of the node-to- node edge j and k is represented by A_{jk}. Node j and node l’s relative weight in the network is represented by A_{jl} .

4.4 Fitness functions

In this section, we’ve looked at two fitness functions in our proposed work: modularity and similarity attribute (SimAtt). We assumed that the goal of our proposed approach is to determine the topological structure of the nodes as well as their shared properties. In terms of topological structure, the bigger the Modularity, the more evenly divided the communities are, the better. A network split with a modularity value close to 1 (the highest level of Modularity) denotes a strong sense of community. The term "modularity" is defined in Eq 7 as follows:

$$Modularity = \frac{1}{2m}\sum \quad \left[A_{ij} - \frac{k_i k_j}{2m}\right] \delta c_i c_j \tag{7}$$

where A is the associated network’s adjacency matrix. In this case, the maximum modularity, the more prosperous the communities are in terms of topological organization. SimAtt, on the other hand, measures the similarity of node attributes in a community. This metric is calculated by using Eq. 8 as follows:

$$SimAtt = \frac{1}{NC}\sum_{i=1}^{NC} \quad max_{1\le j_1 \le j_2 \le \cdots \le j_k} \frac{\sum_{k=1}^{m} \quad n_{i_{j_k}}}{m_{n_i}} \tag{8}$$

where NC , the total community counts, ‘m’ is the total number of attributes assign the community i and $n_{i_{j_k}}$denotes node numbers where h^{th}attribute has the value k_j in the community ‘i’. Besides, n_i is the i^{th} number of nodes in the community? In general, k_1 is the first attribute of nodes having distinct values and k_m is the m^{th} number attributes for distinct values. The higher value of SimAtt, from the standpoint of attribute similarity, that means, the better communities are partitioned.

4.5 Sorting strategy

In this subsection, the sorting strategy is discussed. Each habitat is a solution, and the value of its objective function can be used to estimate its goodness. After determining the value of each habitat's objective function, they are ordered in descending order with the value of their decision function. In contrast to single-objective issues, multi-objective problems include many objective functions and the optimum value for each function provides a list of values.

As a result, the single decision function BBO sorting approach may not be used in multi- objective issues. A sorting algorithm that uses crowding distance for community detection was presented by Deb et al. [32]. The pseudo code for the sorting process in the suggested idea is shown in Algorithm 3.

Algorithm 3: Arrange the Habitats (H_i)

```
Since the value of objective function is given, so determine the rank of each habitat $H_i$.
Determine each habitat's $H_i$ crowding-distance;
Sort the habitat $H_i$ by the value of its habitats' Crowding-distance;
Sort the habitat $H_i$ by the order in which its habitats are ranked;
Return $H_i$;
```

4.6 Selection mechanism

Each habitat (solution) has a Crowding-distance value when the proposed method's sorting technique is completed. The following are the procedures for selecting the proposed method:

1. If the ranks of two solutions disagree, the person with the lowest rank is selected.
2. If two solutions are equal in rank, then one with greater Crowding distance is chosen.

Elitism is performed with this mechanism; it compares the current formed population to the best population of previous generations to find the finest solutions.

4.7 Migration

We believe that during the migration process, the initial habitats are sorted. In BBO, the migration operator may be used to modify and change existing solutions. Furthermore, we employed the rank-based migration technique to locate overlapping communities in this work. As a result, the starting position solution will be the lowest λ as well as the largest μ value. Similarly, the solution at the end position has the maximum λ value and the minimum μ value. Here emigrating habitat Hj is chosen with respect to emigration rate (μ_j). After that, each habitat Hi for those who are emigrating is chosen probabilistically using the roulette wheel process. A habitat with a maximum μ value will have a larger chance of importing its related SIV to habitat Hi using this technique.

4.8 Mutation

Overall, mutation represents to improve the population diversity. It is a probabilistic operator in BBO that SIVs are changed at random in each habitat. Different mutation rates will have a significant influence on the output of a metaheuristic optimization algorithm. If the mutation probability is properly approximated, an evolutionary algorithm will produce effective solutions. As a result, we chose a suitable mutation rate for a set of factors in this paper, which is roughly 0.1 [35].

5.0 EXPERIMENT RESULTS

The findings of our investigation are reported in this section by using four data sets from the actual world. The following three community detection techniques are compared with the results of our proposed methods.

1. Le Martelot and Hankin's Stability Optimization algorithm [36]
2. Modularity maximization BBO Algorithm [25]

3. SimAtt maximization BBO Algorithm [37]

According to Amiri et al. [38], the development of initial solutions is so important to the success of a stochastic algorithm, therefore all outcomes are averaging 10 distinct runs on each data set, and the findings are displayed. The experiment employed an Intel Core 2 Duo 2.20 GHz processor with 2 GB of RAM, and Matlab 7.11.0 was used to create the proposed algorithm (R2010b).

5.1 Parameters Setting

The initialization of LGMBBO includes parameter initialization and population initialization. The Parameter initialization includes setting the population size, the maximum number of species Smax, the maximum immigration rate I, the maximum emigration rate E. To prove the superiority of the proposed framework, the performance metrics like modularity and SimAtt are used to evaluate the experimental results. The proposed model LGMBBO is simulated carefully with different parameters offered in Table 1.

Table 1: Parameters and its value for simulation.

SL.	Parameters	Values
1	Population size: NP	100
2	Maximum immigration rate: I	1
3	Maximum emigration rate :E	1
4	Maximum number of species: S_{max}	50
5	Maximum mutation rate : M_{max}	1
6	Random value rand between	(0,1)
7	Probability of mutation P_{mut}	0.01
8	k is the number of species	25
9	Immaturity index α	0.75
10	Neighborhood's radius r	5%
11	Maximum number of iterations	100-200
12	Running time	10
13	Weighting parameter η	0.5, 1, 1.5
14	Total community counts (NC)	20
15	Total number of attributes (m)	4-11

5.2 Data Sets

For different features (Attrs.), we looked at four real-life data sets of Social Interaction Between Humans (HSI) networks (UKfaculty, Workplace, School1, School2). Table 2 summaries the information presented in this section concerning the data sets.

Table 2: Four real-world data sets were employed in our research work.

SL. No.	Data Sets	Type of Network	No. of Nodes	No. of Edges	Node Attrs.	No. of Attrs.
1	Dataset I: Workplace [39]	HSI	92	755	Department Name	5

2	Dataset II: Primary School–Day1 [40]	HSI	236	2197	Class name	11
3	Dataset III:Primary School–Day2 [40]	HSI	238	2419	Class name	11
4	Dataset IV: U. K. Faculty [41]	HSI	81	577	School Affiliation	4

5.3 Metric for measuring performance

The proposed strategy produces a a collection of non-dominant solutions, as we have discussed earlier in this study. It's a Pareto-based strategy that presents a person who makes decisions with a number of candidates to choose from as prospective solutions. So, taking into account the two elements of node attributes and linkage structure, we proposed a metric for determining the best compromised solution from a collection of non-dominant ones. It indicates that the metric must be able to solve the problem (community partitioning) in which members share comparable characteristics and have a lot of connections. High modularity and similarity of attribute values are a well-detected community indication. As a result, the strength of a community of interests are determined by the difference between the modularity and similarity of attribute metrics, i.e., the larger the difference value, the higher the standard of the discovered community.

We introduce the performance metric, Alpha_Sim, to choose the finest compromise solution from a variety of options. In Eq. 9, the matrices Alpha_Sim may define as follows:

$$Alpha\_Sim = \frac{(1+\eta^2)(SimAtt*Modularity)}{\eta^2*SimAtt+Mudularity} \tag{9}$$

where η is a weighting parameter for SimAtt and Modularity. The value of η should be set to 1 if the strength of relationships and the similarity of qualities are both equally relevant. Also, if we wish to give SimAtt more attention than Modularity, the value of η should be set to an integer in the range [0, 1). If we want to emphasis Modularity over SimAtt, set the value of η to a positive integer in the range (0,∞). As a result, η alters the weighting of two factors: similarity of node properties with strength of connections. In a nutshell, Alpha_Sim is capable of striking a balance between node features that are comparable and the strength of connections.

6.0 RESULT ANALYSIS

In order to test the suggested model's superiority, we compare its results to those of three other relevant models in this section. The outcomes of their objective functions were provided after these models were developed and used four sets of data described in sub- section 5.1. The experimental findings were then evaluated using the performance measure established in sub-section 5.2.

6.1 Comparison of Results

Since, the production of first solutions is so important to the performance of a heuristic algorithm. As a result, the outcomes of 10 different runs of the algorithms used in the experiments on each data set are averaged to provide the results, which are in Table 3.

Table 3: Modularity's mean and standard deviation after the ten times have been averaged method was performed on each set of data.

Data Sets	**Modularity**	**Model 1**	**Model 2**	**Model 3**	**MOBBO _OCD**	**Proposed Model**
Dataset I	Mean	**0.38570**	0.37478	0.08351	0.38570	**0.38870**
	Std	0.00000	0.01253	0.03435	0.00000	0.00000
Dataset II	Mean	0.68290	0.68209	0.02132	0.68487	**0.69487**
	Std	0.00592	0.00112	0.00730	0.00000	0.00120
Dataset III	Mean	0.68077	0.67654	0.04654	0.68166	**0.68546**
	Std	0.00267	0.00405	0.08009	0.00000	0.00021
Dataset IV	Mean	**0.44881**	0.44140	0.23568	0.44881	**0.54881**
	Std	0.00000	0.00381	0.01968	0.00000	0.00001

Table 4: SimAtt's mean and standard deviation after the ten times have been averaged algorithm was performed on each set of data.

Data set	**SimAtt**	**Model 1**	**Model 2**	**Model 3**	**MOBBO _OCD**	**Proposed Model**
1	Mean	0.92153	0.91041	0.95903	0.99804	**0.99874**
	Std	0.00000	0.04754	0.00662	0.00063	**0.00163**
2	Mean	0.63945	0.54932	0.93087	0.95331	**0.96431**
	Std	0.00050	0.03888	0.00534	0.01128	**0.02128**
3	Mean	0.67651	0.61590	0.93216	0.95164	**0.97164**
	Std	0.00062	0.02580	0.01160	0.11063	**0.01063**
4	Mean	0.91574	0.93847	0.99255	1.00000	**1.00000**
	Std	0.00000	0.03623	0.00211	0.00000	**0.00000**

Each algorithm was executed ten times on each set of data and computed the average. After averaging the ten times each algorithm was run on each data set, Table 3 and Table 4 calculate the mean and standard deviation of modularity and SimAtt, respectively. The numbers allocated when the data sets and comparison methods were introduced in the study are used to represent them. For instance, Model 1 is Le Martelot and Hankin's Stability Optimization algorithms and data set 1 is the Workplace data set. Model 2 and Model 3 reflect the Modularity maximization BBO Algorithm and SimAtt minimization BBO Algorithm, respectively. In Table 3 and Table 4, each data set's best result is bolded.

According to the simulation study, the outcomes in Table 3 and Table 4 suggest that the proposed strategy performs better. On each data set, the suggested technique has the greatest mean values among all rivals when SimAtt is taken into account. On the other hand, when considering Modularity's Importance, the suggested technique surpasses the most circumstances; there are options with equal results in others (except workplace set of data, and U. K. Faculty data). The suggested method achieves these improved results because it takes Modularity into account and optimizes it alongside SimAtt as a multi-objective work that employs good static mutation algorithms. Some considerations should be made when it comes to standard deviation.

Since, the proposed technique is a heuristic optimization method. Therefore initial solutions have a significant impact on algorithm's outcomes. Moreover, the algorithm may continue with different starting solutions each time, thus the standard deviation of the proposed algorithm has been exceeded in some circumstances. However, when comparing the outcomes of its standard deviations in comparison to other peers' performance. Overall, the performance of the proposed method is sufficient. The suggested multi-objective optimization model can be considered an appropriate and important heuristic for discovering optimal solutions to community discovery with node attributes in social networks, based on the results provided in Table 3 and Table 4.

As a result, the quality of a network's discovered communities is determined by its modularity value. When compared to other community detection methods, the LGMBBO approach produces better results in terms of community modularity values and attributes similarity.

The mean and standard deviation after each algorithm has been averaged ten times was run on each Modularity data set are shown Fig 3. We can see from Figure 3 that our proposed model performs better for datasets II and III. However, the performance of data set I was comparable to our proposed model. Our proposed model outperforms the other four models in general.

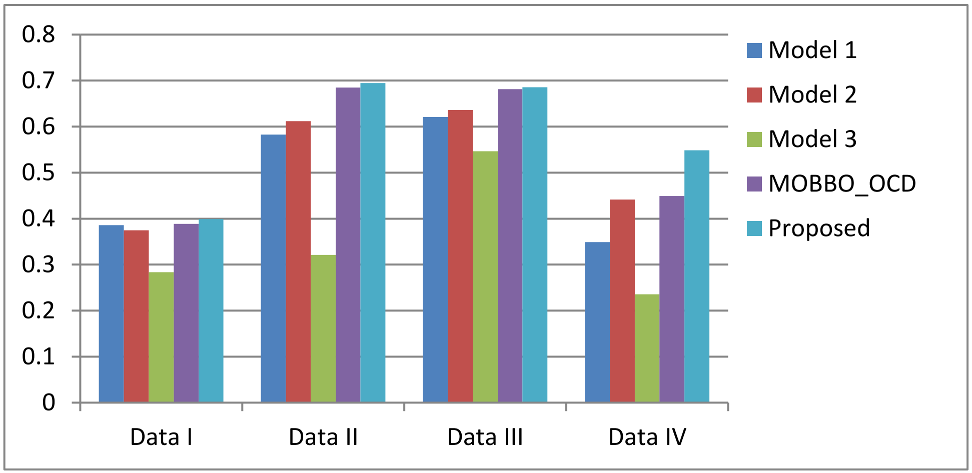

Figure 3(a): Mean of Modularity

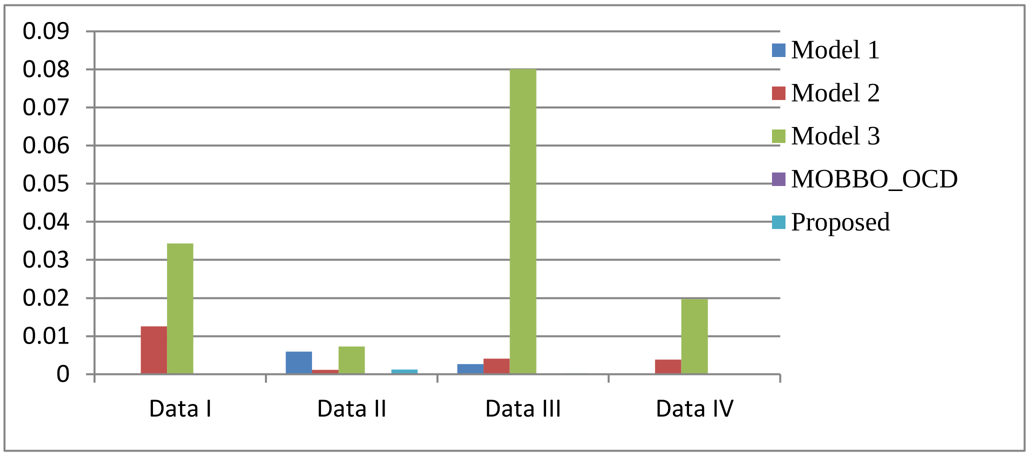

Figure 3(b): Standard Deviation of Modularity

Fig. 3: The average and variation of Modularity after each algorithm has been averaged ten times was performed on every data set are shown in Figures 3(a) and 3(b).

Similarly, after each algorithm has been averaged ten times was run on each data set for similarity of attributes, Fig 4 displays the mean and standard deviation. We can see from Fig 4 that the suggested model outperforms the previous four models.

The values are calculated using these results of Alpha_Sim after getting the SimAtt and Modularity outcomes for all the algorithms. Table 5 and Table 6 show the corresponding Alpha_Sim results for all of the algorithms for each set of data (for each piece of data, the best possible outcome is selected as boldface).

The Alpha_Sim is more concerned with the similarity of node properties. Set η to less than 1 to make them more effective (0< η < 1). When η approaches zero, Alpha_Sim approaches SimAtt, according to Eq. 10:

$$Alpha\_Sim = \frac{(1+\eta^2)(SimAtt*Modularity)}{\eta^2*SimAtt+Mudularity} = SimAtt \quad (10)$$

However, because the Modularity and the connectivity are more significant to us and have a greater impact on *Alpha_Sim*, we set $\eta > 1$. $\eta \rightarrow \infty$, and *Alpha_Sim* approaches to exact value of Modularity, according to Eq. 11:

∞

$$Alpha_{Sim} = \frac{(1+\eta^2)(SimAtt*Modularity)}{\eta^2*SimAtt+Mudularity}$$

$$= \frac{(SimAtt*Modularity)}{SimAtt} = Modularity \qquad (11)$$

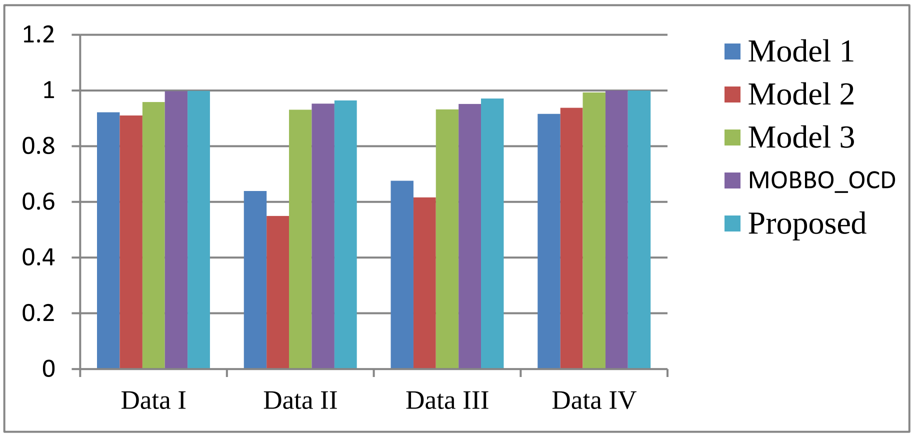

Fig. 4(a): Mean of SimAtt

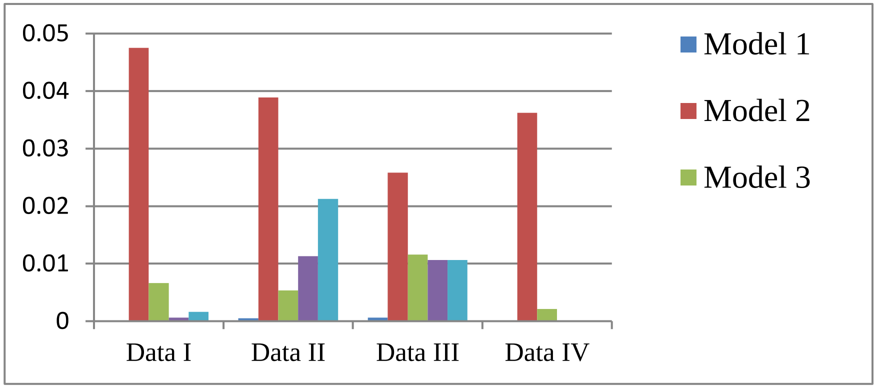

Fig. 4(b): Standard Deviation of SimAtt

Fig. 4: The mean and standard deviation of SimAtt after each algorithm has been averaged ten times was performed on each data set are shown in Fig 4(a) and 4(b).

$$\text{Alpha\_Sim}= \frac{(1+1^2)(SimAtt*Modularity)}{1^2*SimAtt+Mudularity} = \frac{(2)(SimAtt*Modularity)}{SimAtt+Mudularity} \qquad (12)$$

Table 5: After each algorithm has been averaged ten times as performed on each data set, the mean values of *Alpha_Sim (for η = 0.5 and 1.5) were calculated.*

Data set	**SimAtt**	**Model 1**	**Model 2**	**Model 3**	**MOBBO _OCD**	**Proposed Model**
1	$\eta = 0.5$	0.72116	0.70773	0.30021	0.74486	**0.75486**
	$\eta = 1.5$	0.46974	0.45752	0.11545	0.46987	**0.46977**
2	$\eta = 0.5$	0.64769	0.57119	0.09677	0.80464	**0.85464**
	$\eta = 1.5$	0.66890	0.63417	0.03045	0.70021	**0.73221**
3	$\eta = 0.5$	0.67736	0.62700	0.14465	0.80940	**0.85940**
	$\eta = 1.5$	0.67945	0.65643	0.06186	0.69932	**0.69987**
4	$\eta = 0.5$	0.75802	0.76574	0.60332	0.79713	**0.79874**
	$\eta = 1.5$	0.53233	0.52724	0.30774	0.53431	**0.55451**

Table 6: After each algorithm has been averaged ten times was performed on each data set, the mean values of Alpha_Sim (for η = 1) were calculated.

Data set	SimAtt	Model 1	Model 2	Model 3	MOBBO _OCD	Proposed Model
1	$\eta = 1$	0.54379	0.53078	0.15180	0.54590	**0.55590**
2	$\eta = 1$	0.66045	0.60783	0.04157	0.72928	**0.76928**
3	$\eta = 1$	0.67863	0.64456	0.07847	0.73051	**0.73751**
4	$\eta = 1$	0.60239	0.60024	0.38050	0.61180	**0.65480**

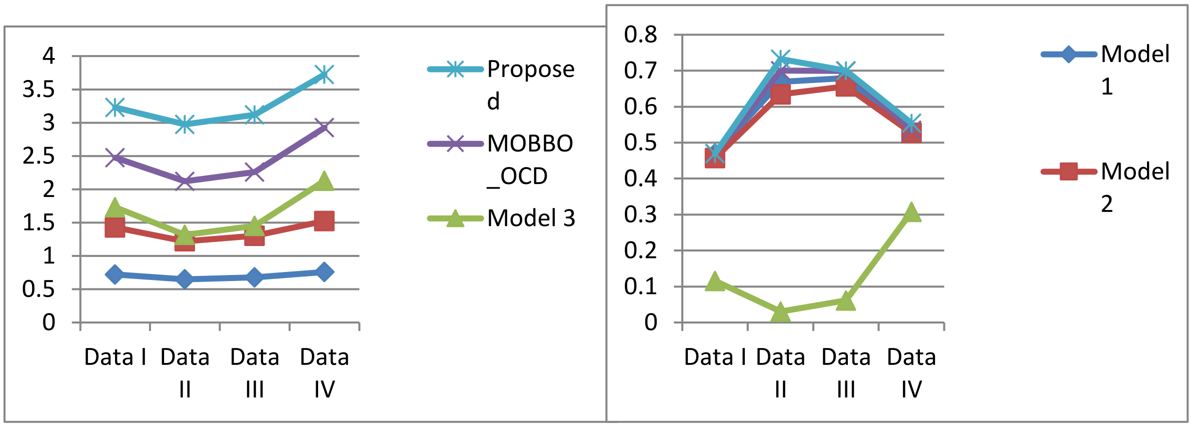

Fig. 5(a): Mean values of Alpha_Sim for η= 0.5

Fig. 5(b): Mean values of Alpha_Sim for η=1.0

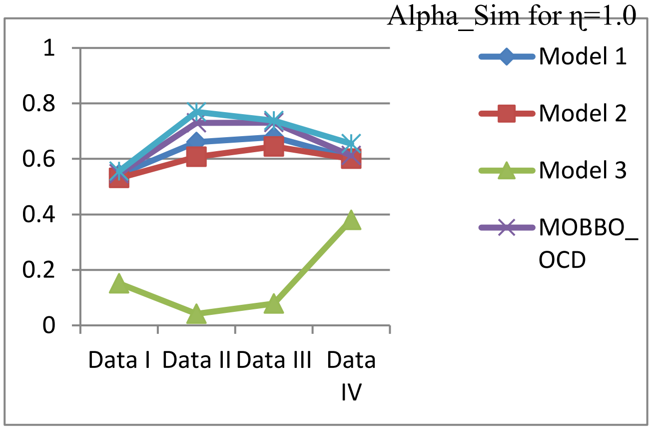

Fig. 5(c) : Mean values of Alpha_Sim for η= 1.5

Fig. 5: Fig. 5(a), 5(b) and 5(c) are the average values of *Alpha_Sim* (where η = 0.5, 1.0 and 1.5)

Accordingly, the mean values of Alpha_Sim for different parameters such η = 0.5, 1.0, and 1.5 after each algorithm has been averaged ten times on each data set are presented in Fig. 5, which illustrates the distinct strengths for node attributes and network connection structure.

We find that the suggested technique has higher SimAtt values due to node similarity characteristics and Modularity due to network connection topologies; as a result, all three η values have a greater Alpha_Sim value. It is now clear that the proposed method outperforms all three of these scenarios:

a. Because node attribute similarity is crucial for the link ($\eta = 1$)

b. As similarity of node attributes are more important at the link ($0 < \eta < 1$)

c. As the similarity of node properties ($\eta > 1$) is more relevant to us than the link after each algorithm has been averaged ten times on each data set.

6.2 Statistical Analysis

To ensure the efficiency of the LGMBBO algorithm, we do statistical analysis on the proposed and other methods. We used Friedman′s test to statistically validate the superiority of the proposed LGMBBO based on the findings of Table 5 and Table 6, and the Friedman statistics are provided in Table 7. To confirm the statistical results, we compare the proposed LGMBBO algorithm with three other algorithms using Friedman′s test. LGMBBO ranks top among all the algorithms in the experiment, according to the results of Friedman′s ranking, which are displayed in Table 6, with the mean rank value for four datasets at $\eta =$ 0.5, 1.0, and 1.5.

6.3 Scalability Analysis

Since the size of a network has a significant impact on an evolutionary algorithm’s running time [42]. The scalability of the proposed method is tested on a network with a variety of numbers of nodes because it is based on an evolutionary process. For the evaluation, one test network of 92, 236, 238, and 81 nodes has been chosen. The proposed method running time increases from 45s to 64s, 110s, and 178s, respectively, when the number of nodes increases from 92 to 236, and 238. As a result, it can be shown that as the network size increases, the proposed method’s running time grows substantially longer. To address this problem, we plan to alter the SimAtt concept in future studies to reduce its computational complexity. The rationale for this is that the main computational cost of the suggested method appears to be linked to SimAtt calculations. SimAtt denotes a higher level of complexity, based on the number of characteristics seen in an experiment. Furthermore, rather than network parameters, it is based on network node characteristics (adjacency matrix).

7.0 CONCLUSION AND FUTURE WORK

In order to disclose hidden relationship patterns among people and better understand human behaviour from a social-economics perspective, community detection in online social networks is a challenging but crucial phenomenon.
This study presents a multi-objective metaheuristic optimization approach to discover communities in social networks utilizing specified node properties. The suggested method favors’ node attribute similarity above the density of links in recognized communities. A new simulation model known as SimAtt to be established for this purpose and it has been combined with Modularity as the proposed algorithm’s goal functions. SimAtt analyses the similarity of node attributes in a community, whereas Modularity approximates community quality from a topological standpoint. Because the proposed strategy is based on the pareto principle, the end result is a set of optimal solutions that are not dominated. As a result, this work introduces Alpha_Sim, a novel evaluation criterion for determining the best compromise solution among non-dominated solutions. In the suggested approach, a sorting strategy and Crowding-distance were utilized to sort the population. In this study, extensive experimentation on four real-world data sets to assess the performance of the proposed technique. In addition, three approaches to relevant techniques were devised so that they can be compared to the one that is being suggested. When compared to the other three approaches, the proposed method outperforms them all, according to the results of experiments.

In the future, we plan to improve the proposed method so that it may be used to find overlapping communities and detect communities in dynamic networks. Similarly, misinformation may be propagated by several sources, each of which has its own community, and these communities may overlap with legal users. As a result, another issue to be addressed is the discovery of overlapping communities involved in information diffusion.

Table 7: The average and final rankings of the four algorithms used in the study according to Friedman′s test, using different values of *Alpha_Sim* on each data set for η = 0.5, 1.0, and 1.5.

Algorithms		Rank (η = 0.5)		Rank (η = 1.0)		Rank (η = 1.5)	
		Mean	Final	Mean	Final	Mean	Final
	Model 1	3.2452	4	3.6753	4	3.7523	3
	Model 2	3.2134	5	4.1211	2	4.2312	2
	Model 3	4.2134	3	3.6754	3	3.5453	4
	MOBBO_OCD	4.3234	2	3.3254	5	3.3243	5
	Proposed	4.5342	1	4.3231	1	4.6541	1
	Model 1	2.5434	2	3.6754	2	2.6211	2
	Model 2	2.2121	4	2.5342	3	3.4532	4
	Model 3	3.2312	3	3.1232	4	2.8564	3
	MOBBO_OCD	3.5124	4	3.2154	3	3.7621	3
	Proposed	3.8345	1	3.7323	1	3.8212	1
	Model 1	3.4212	3	3.3223	2	2.7343	4
	Model 2	2.5334	5	2.3221	5	2.9678	3
	Model 3	2.6332	4	2.6654	4	2.4221	5
	MOBBO_OCD	3.4310	2	3.1001	3	3.0121	2
	Proposed	4.2212	1	3.9987	1	3.1121	1
	Model 1	3.3543	2	2.7865	3	3.2123	3
	Model 2	2.6545	4	2.4342	4	3.1121	4
	Model 3	2.4323	5	2.2121	5	3.0121	5
	MOBBO_OCD	3.2231	3	3.0112	2	4.2134	2
	Proposed	3.8767	1	3.9786	1	4.4323	1

REFERENCES

[1] J. Scott, “Social network analysis: developments, advances, and prospects”, *Social Network Analysis and Mining*, Vol. 1, 2011, pp. 21–26.

[2] B. Bringmann, M. Berlingerio, F. Bonchi, and A. Gionis, “Learning and predicting the evolution of social networks”, *IEEE Intelligent Systems*, Vol.25, No. 4, 2010 pp. 26–35.

[3] R. Parvathi, and M. Monica, “Survey on social network community detection and ranking”, *International Journal of Control Theory and Applications*, Vol. 10, No. 25, 2011, pp. 159–164.

[4] J. Xie, S. Kelley, and B. Szymanski, “Overlapping community detection in networks: the state-of-the-art and comparative study”, *ACM Computer Survey*, Vol. 45, No. 4, 2013 pp. 1–35.

[5] X. Bai, P. Yang, and X. Shi, “An overlapping community detection algorithm based on density peaks”, *Neurocomputing*, Vol. 226, 2017, pp. 7–15.

[6] M. Divy, and R. Ramalakshmi, “Hybrid machine learning approach for community and overlapping community detection in social network”, *Transactions on Emerging Telecommunication Technologies*, vol 33, no 10, 2020, e4161

[7] Z. Xu, Y. Liu, B. Li, and H. Li, “A multi-objective discrete cuckoo search algorithm for community detection in dynamic networks”, *Soft Computing*, Vol. 21, No. 22, 2017, pp. 6641–6652.

[8] S. Zhao, L. Yu, and B. Cheng, ’Probabilistic community using link and content for social networks”, *IEEE Access*, Vol. 5, 2017, pp. 27189–27202.

[9] S. Kianian, M. Khayyambashi, and N. Movahhedinia, “FuSeO: fuzzy semantic overlapping community detection”, *Journal of Intelligent Fuzzy System*, Vol. 32, No. 6, 2017, pp. 3987–98.

[10] A. Sarswat, V. Jami, and R. Guddeti, “A novel two-step approach for overlapping community detection in social networks”, *Social Network Analysis Mining*, Vol. 7, No. 1, 2017, pp. 47–58.

[11] B. Rees and K. Gallagher, “Overlapping community detection using a community optimized graph swarm”, *Social Network Analysis Mining*, Vol. 2, No. 4, 2012, pp. 405–417.

[12] C. Liu, J. Liu, and Z. Jiang, “A multi-objective evolutionary algorithm based on similarity for community detection from signed social networks”, *IEEE Transactions on Cybernetics*, Vol. 44, No. 12, 2014, pp. 2274–2287.

[13] D. Simon, “Biogeography-based optimization”, *IEEE Transaction Evolutionary Computing*, Vol. 12, No. 6, 2008, pp. 700–713.

[14] A. Reihanian, B. Minaei, and H. Alizadeh, “Topic oriented community detection of rating based social networks”, *Journal King Saud University Computing Information Science*, Vol. 28, No. 3, 2015, pp. 303–310.

[15] J-H. Yi, L-N. Xing, G-G Wang,. J. Dong, A. V. Vasilakos, A. H. Alavi, and L. Wang, “Behavior of Crossover Operators in NSGA-III for Large-Scale Optimization Problems”, *Information Sciences*, Vol. 509, 2020, pp. 470–487.

[16] J. Leskovec, J. Kleinberg, and C. Faloutsos, “Graph evolution: Densification and shrinking diameters”, *ACM Transactions on Knowledge Discovery from Data*, Vol. 1, No. 1, 2007, pp. 1- 41.

[17] M. Newman, and M. Girvan, “Finding and evaluating community structure in networks”, *Phys. Rev. E.*,arXiv:cond-mat/0308217 Vol. 69, No. 2, 2004, pp. 1–16.

[18] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, “Uncovering the overlapping community structure of complex networks in nature and society”, *Nature*, Vol. 435, 2005, pp. 814–818.

[19] Z. Xia, and Z. Bu, “Community detection based on a semantic network”, *Knowledge Based System* , Vol. 26, 2012, pp. 30–39.

[20] P. DeMeo, A. Nocera, G. Terracina, and D. Ursino, “Recommendation of similar users, resources, and social networks in a Social Internetworking Scenario”, *Information Science*, Vol. 181, 2011, pp. 1285–1305.

[21] X. Zhang, C. Wang, S. Yansen, L. Pan, and H-F. Zhang, “A fast overlapping community detection algorithm based on weak cliques for large-scale networks”, *IEEE Transactions on Computational Social Systems*, Vol. 4, No. 4, 2017, pp. 218–230.

[22] M. Atzmueller, S. Doerfel, and F. Mitzlaff, “Description-oriented community detection using exhaustive subgroup discovery”, *Information Science*, Vol. 329, 2016, pp. 965–984.

[23] C. Pizzuti, “’Multi-objective genetic algorithm to find communities in complex networks”, *IEEE Transaction Evolutionary Computing*, Vol. 16, No. 3, 2012, pp. 418–430.

[24] Z. Zhu, T. Zhou, C. Jia, J. Liu, and J. Cao, “Community detection across multiple social networks based on

overlapping users", *Transaction Emerging Telecommunication Technology*, vol 33 No 26, 2022 *arXiv*:1909.09007v2.

[25] A. Reihanian, , F-D. Mohammad-Reza and S. A. Hadi, "Community detection in social networks with node attributes based on multi-objective biogeography-based optimization", *Engineering Applications of Artificial Intelligence*, Vol. 62, 2017, pp. 51–67.

[26] P. K. Giri, S.S. De, and S. Dehuri, "A Novel Locally and Globally Tuned Biogeography- based Optimization Algorithm", *Soft Computing: Theories and Applications (SoCTA*), Vol. 583, 2017, pp. 635–647.

[27] P. K. Giri, S. S. De, S. Dehuri, and C. S. Bae, "Locally and Globally Tuned Chaotic Biogeography-based Optimization Algorithm", *IEEE International Conference on Information Technology, ICIT*, Bhubaneswar, 2018 pp. 152–158.

[28] P. K. Giri, S. S. De, and S. Dehuri, "Adaptive Neighbourhood for Locally and Globally Tuned Biogeography Based Optimization Algorithm", *Journal of King Saud University–Computer and Information Sciences,* Vol. 33, No. 4, 2021, pp. 453–467.

[29] S.S. De, P. K. Giri, and S. Dehuri, "Link Recommendation for Social Influence Maximization", *In: Biologically Inspired Techniques in Many Criteria Decision Making*, Vol. 271, No. 2, 2022, pp. 71–93.

[30] E. Zitzler, and L.Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach", *IEEE Transaction Evolutionary Computing*, Vol. 3, No. 4, 1999, pp. 257–271.

[31] Z. Yuxin, L. Shenghong, and J. Feng, "Overlapping community detection in complex networks using multi-objective evolutionary algorithm", *Computational and Applied Mathematics*, Vol. 36, No. 1, 2015, pp. 1–20.

[32] K.. Deb, A. Pratap, S. Agarwaand, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II", *IEEE Transactions Evolutionary Computing*, Vol. 6, No. 2, 2002, pp. 182–197.

[33] Z. M. Gu and G-G. Wang, "Improving NSGA-III algorithms with information feedback models for large-scale many-objective optimization", *Future Generation Computer Systems*, Vol. 107, 2020, pp. 49–69.

[34] Q. Zhang, and H. Li, "A multi-objective evolutionary algorithm based on decomposition", *IEEE Transactions on evolutionary computation*, Vol. 11, No. 6, 2007, pp. 712–731.

[35] P. K. Giri, S. S. De, S. Dehuri, and C. S. Bae, "Biogeography based optimization for mining rules to assess credit risk", *Intelligent Systems in Accounting, Finance and Management*, Vol. 28, No. 1, 2020, pp. 35–51.

[36] E. LeMartelot, and C. Hankin, "Multi-scale community detection using stability optimization within greedy algorithms', *arXiv preprint arXiv*, Vol. 1201, No. 3307, 2012.

[37] X. Zhou, Y. Liu, B. Li, and G. Sun, "Multi-objective biogeography-based optimization algorithm with decomposition for community detection in dynamic networks", *Physica A: Statistical Mechanics and its Applications*, Vol. 436, 2015, pp. 430–442.

[38] B. Amiri, L. Hossain, J. W. Crawford, and R. T.Wigand, "Community detection in complex networks: multi-objective enhanced firefly algorithm", *Knowledge Based System*, vol. 46, No. 2, 2013, pp. 1—11.

[39] M. Genois, C. L. Vestergaard, J. Fournet, A. Panisson, I. Bonmarin, and A. Barrat, "Data on face-to-face contacts in an oce building suggest a low-cost vaccination strategy based on community linkers", *Network Science*, Vol. 3, 2015, pp. 326–347.

[40] J. Stehle, N. Voirin, A. Barrat, C. Cattuto, L. Isella, J.F. Pinton, M. Quaggiotto, W. V. Broeck, C. Regis, and B. Lina, "High-resolution measurements of face-toface contact patterns in a primary school", *PLoS One*, Vol. 6, no. 8, 2011 e23176.

[41] T. Nepusz, A. Petroczi, L. Negyessy, and F. Bazso, "Fuzzy communities and the concept of bridgeness in complex networks", *Physics and Society*, Vol. 77, 2008, pp. 16–107.

[42] Y. Zhang, G-G. Wang, K. Li, W-C. Yeh, M. Jian, and J. Dong, "Enhancing MOEA/D with Information Feedback Models for Large-Scale Many-Objective Optimization", *Information Sciences*, Vol. 522, pp-343—365, 2020.

[43] M. Contisciani, E. A. Power, and C. D. Bacco, "Community detection with node attributes in multilayer networks", Scientific Reports, vol. 10, No. 15736, 2020.

[44] K. Taha, P. Yoo, and F. Z. Eddinari, "Detecting implicit cross-communities to which an active user belongs", PLoS ONE, Vol. 17, No. 4, 2022, e0264771.

[45] B. F. Kamhoua, L. Zhang, K. Ma, J. Cheng, B. Li, and B. Han, "GRACE: A General Graph Convolution Framework for Attributed Graph Clustering", ACM Transactions on Knowledge Discovery from Data, vol 17, no 3, 2023, pp 1 – 31.