Vital Annex: International Journal of Novel Research in Advanced Sciences
ISSN: 2751-756X
Volume 04 Number 06 (2025)
https://innosci.org/IJNRAS

Article

A Discrete Bat Algorithm for Detecting Overlapping Communities in Dynamic Graph-Based Social Networks

Ali Radhi Habeeb[*1]

1. Department of Computer Engineering, Faculty of Engineering and Technology, Islamic Azad University, Iran
* Correspondence: aliradhe2222@gmail.com

Abstract: Community identity is an important task in analyzing social networks, which aims to identify underghier, where nodes are closely connected to internal and are very connected to the rest of the network. These communities can be dissatisfied - where each node belongs to only one society - or overlapping, leaves the nodes related to many communities. The study suggests an unspoilt BAT algorithm (DBA) to detect local communities in symmetrical dynamic networks. Inspired by the eco -location behavior of microbhates, the original BAT algorithm effectively solves continuous adaptation problems, but the application is limited in disconnected domains such as societal identity. To address this, a discreet adaptation has been developed, which represents the status of the bat in an appropriate way to solve combinatory problems. Social identification is designed as a multi -use problem in the stages of a dynamic network time, with two objective functions: First, stable and meaningful social structures mean each time, and the other promotes temporary stability by maximizing the resemblance in continuous time stages. Experimental results suggest that the proposed DBA has rejected existing methods in existing methods, including the Particle Herd-Adaptation (PSO), Genetic Algorithms (GA) and Multi-Lens Biogography-based optimization (MOBO), the lowest bordered rand (ARI) and the lowest error in 98.

Keywords: Optimal Bat Algorithm, Social Structures, PSO, Community Detection

1. Introduction

In the last several years, there has been a proliferation of social networks and, as a result, a proliferation of research into analyzing and understanding intricate relational structures within them. One of the fundamental problems in this field is community discovery—i.e., discovering groups of users or entities that are more interconnected among themselves than they are with the remainder of the network. Detection of communities is central to understanding user behavior, recommendation systems, and information flow. Social network communities can be either distinct (non-overlapping) or overlapping. While traditional approaches have been focused on distinct community structure, social networks in the real world are prone to having overlapping communities, with every member belonging to more than a single group simultaneously, demanding more robust and flexible detection methods [1], [2].

To overcome this issue, graph theory is a strong paradigm to model networked structures with entities as nodes and relationships as edges. However, community detection, especially under dynamic and overlapping conditions, involves computational issues requiring smart optimization methods. Bio-inspired evolutionary algorithms have shown promise to solve such complex problems [3], [4]. In this work, an innovative method for overlapping community detection in dynamic graph-based social networks is

Citation: Habeeb A. R. A Discrete Bat Algorithm for Detecting Overlapping Communities in Dynamic Graph-Based Social Networks. Vital Annex: International Journal of Novel Research in Advanced Sciences 2025, 4(6), 179-188.

Received: 24th May 2025
Revised: 31th May 2025
Accepted: 8th Jun 2025
Published: 17th Jun 2025

proposed based on a Discrete Bat Algorithm (DBA). Although the traditional bat algorithm, motivated by the echolocation ability of bats, is initially proposed for continuous optimization problems, its extension to discrete problem domains—like community detection—is still nascent. To overcome this drawback, we suggest a discrete version of the bat algorithm which is capable of handling the combinatorial space of social graph community structures. The suggested approach formulates the dynamic community detection issue as a multi-objective optimization problem. It develops two objective functions: one providing optimality and stability of social structures at every time step of the dynamic network, and the other providing temporal consistency by maximizing the similarity between community structures at adjacent time steps. The two-objective approach improves the accuracy and continuity of community detection along time.

Experimental research on benchmark datasets validates the performance of the newly proposed DBA method. The result demonstrates an obvious enhancement in accuracy of community detection over state-of-the-art algorithms including Particle Swarm Optimization (PSO), Genetic Algorithms (GA), and Multi-objective Biogeography-Based Optimization (MOBBO). The proposed method, in particular, achieves an improvement of 14.54% in ARI (Adjusted Rand Index) and obtains the lowest error rate, which is evidence of its stability and feasibility for dynamic network analysis in real-world problems.

Related Work

Social network community detection has been among the forefront research problems in the past few years owing to its importance in revealing the hidden structure of complex systems. Community detection methods can be generally classified into two types: detection of disjoint (disjoint) communities and overlapping communities. Although initial methods, i.e., maximization of modularity (Girvan–Newman algorithm), were able to discover disjoint communities effectively, they were unable to cope with the reality that, in overlapping communities, members are part of more than one group at a time. Metaheuristic algorithms have received much attention due to their flexibility and efficiency in exploring huge search spaces. Of these, evolutionary algorithms like Particle Swarm Optimization (PSO), Genetic Algorithms (GA), and Ant Colony Optimization (ACO) have been modified for community detection. These methods transform the detection problem into an optimization problem with objective functions over modularity, density, or community similarity. Most of these algorithms are, nevertheless, meant for static networks and discrete community structures [5], [6]. The Bat Algorithm (BA), which mimics microbat echolocation, has demonstrated excellent potential in a wide range of optimization issues. Initially intended for continuous search spaces, BA has been generalized and discretized ever since to be used for graph-based issues. Recent research has modified BA for discrete spaces, but none, or perhaps very few, have addressed overlapping community detection in dynamic networks—a field that is further complicated because of the time-based nature of network structures [7]. For instance, The CoDeSEG approach, introduced by Xian et al., discovers groups by decreasing the 2D underlying entropy of the network in a probable game setting. The nodes in the game optimize the 2D structural entropy utility function to decide whether they should stay in their current community or switch to another one. They also suggest a structural entropy-based node overlapping heuristic to identify overlapping communities with approximate linear temporal complexity. Experimental evaluation on real-world networks shows that CoDeSEG is the most efficient and has the best performance in ONMI and F1 score. Aslan suggested COOT method (MCOOT), the first update process is used to produce a new position of the current COOT particle, and then the proposed update process. There were three modifications required to adapt to the new update process: To facilitate exploration, the coordinates of the current coot individual are randomly selected between 1 and the problem size, updated based on the suggested update rule, and a genetic mutation operator is implemented based on a mutation probability [8], [9]. As the CD problem is a discrete problem, the suggested MCOOT method converts continuous values of current coot positions to discrete values. After adapting the MCOOT, it was tested on ten various network issues of varying sizes to ascertain its performance. Finally, empirical findings of MCOOT methodology are compared with the state-of-the-art optimization algorithms on solution quality as well as time analysis. Our study found that the suggested methodology outperformed 22 available algorithms for all community finding problems. The suggested methodology achieves the same or better solution quality and resistance, as proved by general results. The new strategy has the possibility to be more competitive, particularly for discrete problems. Abdulrahman presented the recent advancements in multi-objective algorithms to find communities in signed networks under modularity and frustration minimization [10]. Our study explores applications of ant colony optimization, genetic algorithm, and memetic algorithm for identifying community topology. The study concentrates on comparing positive and negative relationships of social networks in order to finish their structural analysis. Under these limitations, this study proposes a Discrete Bat Algorithm for overlapping society discovery in active social networks. The method leverages the power of evolutionary optimization by integrating discrete representation mechanisms and multi-objective functions for maintaining community structure over time [11]. In comparison to other metaheuristic methods, the discrete BA is more appropriate for managing the discrete and overlapping nature of memberships in different networks.

2. Materials and Methods

Taking into consideration the many definitions of a dynamic network, the purpose of this investigation is to determine which communities (community structures) on a dynamic network are the most effective at respectively time step. In order for the proposed strategy to be successful in achieving this objective, it must possess the two key traits listed below:

- For each graph i in the network G, the proposed method should determine community structures that partition the graph into stable and meaningful communities. Additionally, the proposed method should refrain from overly dividing the graph into numerous communities.
- The community structures identified at each time step should be related to and consistent with those found in earlier time steps."The flow chart of the simulation run and community structure acquisition process in every network of time steps of a dynamic network is shown in Figure 1."

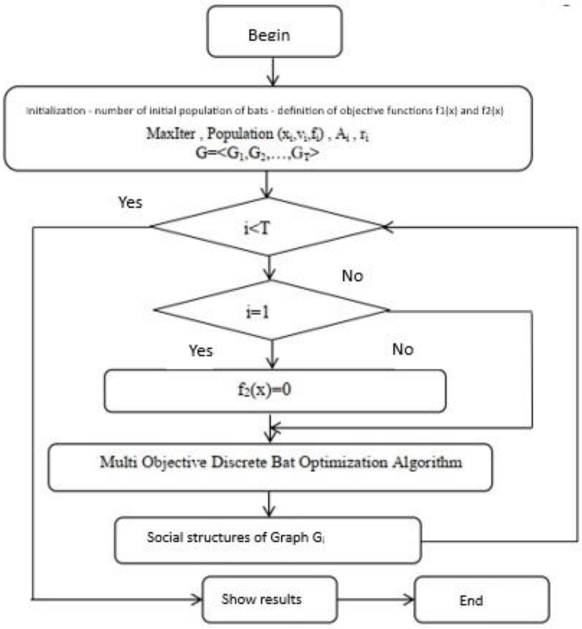

Figure 1. flowchart of the proposed algorithm for community structure detection in a complex dynamic network.

Taking into account T time steps in a dynamic network, the flowchart that follows demonstrates how to apply the improved Bat Optimization Algorithm (BOA) that was discussed in the part that came before it in order to identify community structures at each respective time step. The second objective function is one that is based on the community structures that were discovered in the time step that came before it, which was t−1. Because of this, the objective function is set to zero when the time is equal to one (1), see Figure 2.

```
Inputs:
        Define dynamic network G=<G_1,G_2,…,G_T› with T time step (T network)
        Define objective function 1 as f_1(x)using equation 4-1
        Define objective function 2 as  f_2(x)using equation 4-4
        Define Maximum Iteration as MaxIter
        Define Population size (Number of bats) as PopSize
        Define Variable size (length) of bats propotion to number of nodes in network Define puls rate of bats as A_i
        Define   frequency
of bats as f_i Outputs:
        Communities of network G
Begin
        //foreach network do
        For i=1≤T {
        If (i==1) {
        F_٢(x)=0//because we in thiscondition we don't have a community for comparison Else{
        Communities(i)=OptimizedBatAlgorithm(G_i,f_1(x),f_2(x),MaxIter,Popsize,A_i,f_i)
        }
        Plot results
End
```

Figure 2. pseudo code of the proposed method in detecting community structures in a complex, dynamic network.

Improved Bat Optimization Algorithm (IBOA) Parameters

Many parameters of the developed algorithm need initialization, including the total of repetitions and the initial bat inhabitants size. These parameters greatly impact the accuracy and execution time of the evolutionary algorithm. Table 1 below displays the set parameter values for the improved Bat Optimization Algorithm. These values were determined by trial-and-error and repeated testing.

Table 1. Set parameters for the improved Bat Optimization Algorithm

Value	Description	Parameter
20	Maximum iteration	MaxIter
150	Bat Population Size (Search Agents)	Bats
9.0	Loudness (A)	A_i
8.0	Pulse Rate (r)	r_i
2	Maximum Frequency (f_max)	F

3. Results and Discussion

3.1 Detecting community structures in a dynamic, real-world network

3.1.1 Datasets

The suggested improved Bat Optimization Algorithm was also employed to detect community structures in a real-world dynamical network. The active network was derived from a complex system of football competitions between various states in the United States., In this system, the entities are football teams, each of which competes with others and earns points. Therefore, the match between two teams forms the connection between two entities (teams) in the network. The football competitions between teams have taken place in different years. Consequently, football games between teams in a specific year generate a network at a specific time step. For simulation purposes, we used a dataset of these matches from five consecutive years (2002-2006). The number of matches and the teams participating in the tournaments varied each year. We chose 911 teams that took part in each of the five years to normalize each graph from the time steps. The data regarding each year is contained in a .txt file. In the preprocessing stage, the scores of the match between the two teams are deleted, and only the names of the teams (i.e., nodes or entities) are preserved. Following the selection of the 911 teams, a MATLAB function generates the adjacency matrix for every year. Then, The enhanced Bat Optimization Algorithm uses this matrix as input to identify community structures at each time step. In this dataset, each team belongs to a specific conference. As a result, it is possible to group the teams based on the conference to which they belong. In the dynamic representation of this network, the community structures correspond to the conferences, and each team belongs to one of them. The 911 selected teams belong to 21 conferences. A list of these conferences is provided in Appendix 1. This grouping, as ground truth, can be very useful in evaluating the community structures detected by the algorithm. Moreover, it can estimate the error in detecting the community structures The competition-year dataset is available at http://www.jhowell.net/cf/scores, and the 911 teams' classification across 21 conferences is available at http://www.jhowell.net/cf/scores/byConf.htm.

3.1.2 Evaluation criteria for community detection

One way to measure the quality of a method in detecting community structures of a network (dynamic or static) is to align the detected communities by the method with the real ground-truth community structures of the network. The ground-truth community structures are the fundamental and real structures present in a network. In fact, the success of the proposed method in identifying the community structures is quantified according to how detected community structures are compared to real ground-truth shapes. That is, the method is more successful in identifying the community structures of the network if the detected community structures are nearer to the real ground-truth structures and the gap (error) between them is smaller.

In the present study, we evaluate the performance of the proposed method for discovering community structures in a dynamic network by comparing the discovered community structures with the actual ground-truth community structures. Specifically, we utilize the Adjusted Rand Index (ARI), which was explained in the preceding chapter, to measure the degree of similarity between the discovered community structures and their corresponding ground-truth structures. The ARI is a number between 0 and 1 and indicates the similarity between the two sets of data. As a preliminary step to perform the comparison according to the index defined above, the suggested approach first identifies the community structures in a dynamic network. That is, the nodes making up the network are clustered (i.e., each node is placed in a different community). Then the similarity between the recovered community structures (the clustered nodes) and the ground-truth structures is measured by the Adjusted Rand Index (ARI). Observe that in all such comparisons, the more accurate and stable the method is in uncovering community structures, the closer the value of ARI to 1. for comparison and validation with other methods, we also applied the Normalized Mutual Information (NMI) index and the error measure. The error and NMI measure computation of two clusters is defined in equations 5-1 and 5-2, respectively.

$$\text{Error } \|ZZ^T - GG^T\| \qquad (5\text{-}1)$$

In the above equation, Z is a matrix with n rows and k columns. This matrix specifies how n nodes are assigned to each of the k community structures that the algorithm found. Additionally, G is a matrix with the same dimensions that shows the nodes' assignment to their real community structure (ground-truth structure). We use the error rate to calculate the distance involving the two matrices, which reflect the ground-truth community constructions and the identified community formations.

$$NMI = \frac{-2\sum_{i=1}^{CA}\sum_{i=1}^{CB} C_{ij}\log\left(\frac{C_{ij}N}{C_i.C_j}\right)}{\sum_{i=1}^{CA} C_i.\log\left(\frac{C_i}{N}\right)+\sum_{j=1}^{CB} C_j\log\left(\frac{C_j}{N}\right)} \qquad (5\text{-}2)$$

In the equation above, **A** and **B** represent the identified community constructions and the ground-truth community structures, respectively. Additionally, **C** is the confusion matrix, where each element **C**$_{ij}$ represents the animated of nodes from the **i**th communal structure in **A** that belong to the **j**th community structure in **B**.

3.1.3 Simulation results

The flow chart of the simulation run and community structure acquisition process in every network of time steps of a dynamic network is shown in Figure 1. The processes involved in the simulation and community structure acquisition process in the dataset are shown in the flow chart. The suggested algorithm is first applied to the football game dataset to find the community structures, as shown by the flow chart. That is, the dynamic network is split into five distinct datasets, one for each year from 2005 to 2009. The adjacency matrix of each of the resultant datasets is computed. The ith dataset's adjacency matrix is represented as Ai. The adjacency matrices are hence obtained from the set A = {A1, A2, A3, A4, A5}. This version utilizes a loop that is executed five times since the size of set A is five as well. In each iteration of the loop, the adjacency matrix Ai is utilized while it passes through the bat optimization algorithm for the number I for the purpose of finding community structures [12]. That is, the algorithm is run five times, once for each of the five networks making up the dynamic network used in the analysis of the new method. The figure being referenced in the "Simulation Results" section is actually Figure 3.

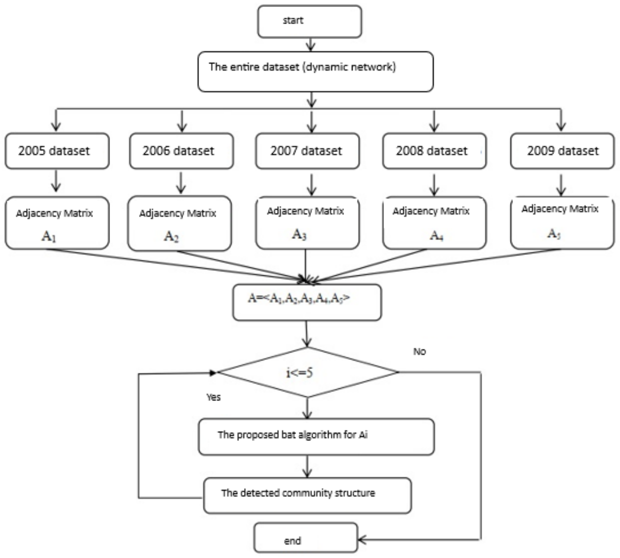

Figure 3. Simulation flowchart.

3.1.4 Results

As is evident in picture 3, the proposed algorithm starts the community structure identification process in every graph of the dynamic network according to the parameters that are described in Table 2.

Table 2. comparison of the graph of the tenth detected community structure.

Detected Community Structure	Ground-Truth Community Structure	Incorrect Detection	ARI Criterion
Alabama	Alabama	-	1
Arkansas	Arkansas		
Auburn	Auburn		
Florida	Florida		
Georgia	Georgia		
Kentucky	Kentucky		
Louisiana State	Louisiana State		
Mississippi	Mississippi		
Mississippi State	Mississippi State		
South Carolina	South Carolina		
Tennessee	Tennessee		
Vanderbilt	Vanderbilt		

"As is evident in Figure 4, the proposed algorithm starts the community structure identification process.

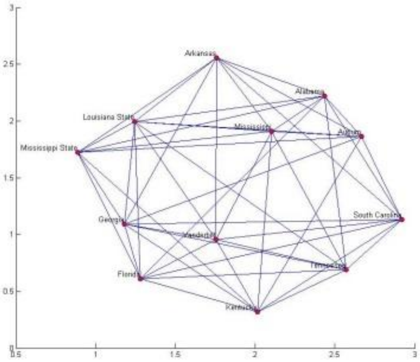

Figure 4. the graph of the tenth detected community structure and its comparison with the corresponding ground truth.

Now we will discover the community structures in the first graph and then compare them with the ground truth structures in a different manner. A sample of the graph structures learned and a comparison of the structures with the ground truth is provided . One hundred percent of the time, each of the graph structures learned that are shown in these figures is identical to the structure of the ground truth that it corresponds to. A convergence plot of the fitness value of the algorithm is presented in Figure 5. The plot is provided for each graph after 20 iterations. Based on the evidence presented in figure 5, the graph for year 2005 (graph 1) illustrates a greater level of convergence in terms of fitness compared to the graphs for the remaining time steps. But when compared with the other time steps, the graph for 2006 has the least fitness convergence. The community structures revealed in graph 1 (of 2005) should therefore be far more stable than those revealed in the graphs for the other years. This is because graph 1 has a greater fitness than the other graphs. To quantify the stability of discovered community structures in each time step graph, we employed the ARI index, which was described in the preceding chapter. We utilized the index to compare the discovered structures to the ground-truth, which were real structures, which were conferences the teams were members of [13], [14].

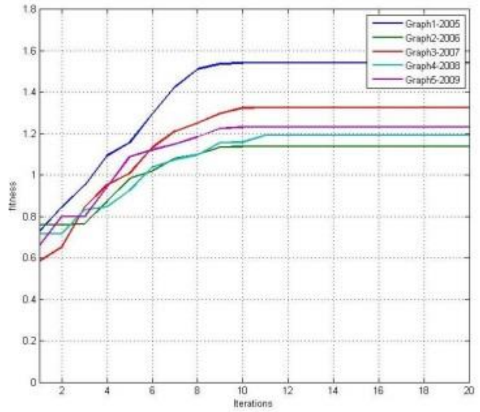

Figure 5. the convergence graph of the community detection algorithm across the 5 graphs of the dynamic network.

Adjusted Rand Index (ARI) is a standard metric for comparing the similarity of clustering results and ground truth labels with a normalized score that corrects for chance clusters. In this research work, ARI scores of the identified communities in every graph of the dynamic network are plotted in Figure 6. It displays a comparative study of how closely the identified communities by the algorithm match the real, predefined communities. Through a more detailed inspection of Figure 6, we can see that the community structures achieved in the first graph of the temporal evolution network are indeed much nearer to the ground truth communities, reflected by a relatively higher ARI value. This tells us that the presented algorithm is especially good at identifying meaningful community structures at the very beginning of the temporal evolution process of the network.

In contrast, the ARI values for the second graph are considerably lower, which means that the communities found in this case are unstable and further away from the assumed ground truth. Such instability may occur because of transitional structural evolution of the network during the time period. In addition, the simultaneous observation of Figure 6 indicates the close relationship between the convergence trend of the fitness function of the algorithm and the stability of identified communities [15]. That is, whenever the algorithm exhibits quicker or better convergence with respect to fitness value, it also tends to provide more stable and closer-to-ground-truth community structures on each graph. This observation reveals the strength of the optimization process in improving the quality of community detection in dynamic network settings.

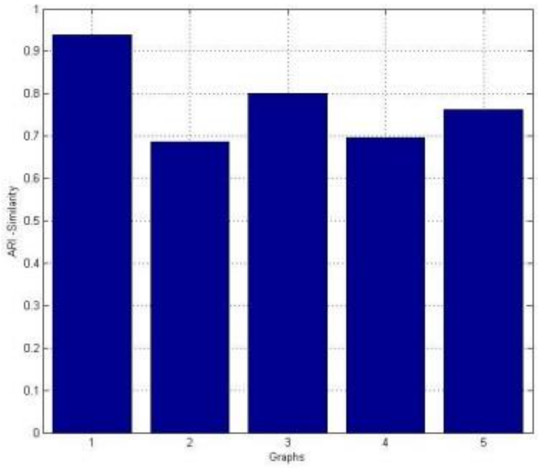

Figure 6. The degree of similarity between the detected community structures in each graph and the ground truth.

4. Conclusion

Networks, nodes (or points) connected by lines or edges, can describe systems in many scientific domains. These include the Web, community networks, neural networks, and distribution and communication networks. Although network systems have been researched for millennia, attention has grown in the past decade. Community structures in complex networks are a popular topic of study. Community structures in networks are nodes with highly connected edges and surface connections to other nodes. Complex networks' community structures hold significant information, thus they must be detected. Complex static and dynamic networks can be used to detect community structure. Node connections are defined at a specific period in static networks. Thus, community structure identification research has focused on this network. Real-world complex networks are dynamic. System entity interactions form and disappear at different time steps. It's significantly more important to recognize community structures in dynamic, complicated networks to find meaningful links and information. Because dynamic networks are dynamic, standard algorithms struggle to identify community structures. Thus, creative strategies in this field should be prioritized. This paper introduces an optimized Bat Algorithm to maximize community structure detection in dynamic complex networks. Community structures for dynamic networks can be generated by the optimization process. Refining these solutions may determine the best stable community structures in each dynamic network time step.

REFERENCES

[1] M. Madadyar, V. Majidnezhad, J. Bagherzadeh Mohasefi, and F. Soleimanian Gharehchopogh, “An enhanced hybrid optimization algorithm for community detection in social network,” Int. J. Data Sci. Anal., pp. 1–23, 2025.

[2] I. Messaoudi and N. Kamel, “A multi-objective bat algorithm for community detection on dynamic social networks,” Appl. Intell., vol. 49, pp. 2119–2136, 2019.

[3] I. Messaoudi and N. Kamel, “Overlapping community detection with a novel hybrid metaheuristic optimisation algorithm,” Int. J. Data Min. Model. Manag., vol. 12, pp. 118–139, 2020.

[4] Z. Boulouard, A. El Haddadi, F. Bouhafer, A. El Haddadi, L. Koutti, and B. Dousset, “Bat-cluster: a bat algorithm-based automated graph clustering approach,” Int. J. Electr. Comput. Eng., vol. 8, p. 1122, 2018.

[5] X. Zhang, Q. Hu, and J. Wang, “A comprehensive review of computational intelligence to community detection,” in Proc. CCF Conf. Comput. Supported Cooperative Work Soc. Comput., 2024, pp. 35–49.

[6] M. Bhattacharya, Design of Authentication Protocols for Mobile Online Social Networks, 2023.

[7] I. Deutschmann, Disentangling Ecological Networks in Marine Microbes, 2021.

[8] I. Koc, “A fast community detection algorithm based on coot bird metaheuristic optimizer in social networks,” Eng. Appl. Artif. Intell., vol. 114, p. 105202, 2022.

[9] S. Taibi, L. Toumi, and S. Bouamama, “Complex network community discovery using fast local move iterated greedy algorithm,” J. Supercomput., vol. 81, p. 182, 2025.

[10] C. S. Siew, “Community structure in the phonological network,” Front. Psychol., vol. 4, p. 553, 2013.

[11] B. S. Rees and K. B. Gallagher, “Overlapping community detection using a community optimized graph swarm,” Soc. Netw. Anal. Min., vol. 2, pp. 405–417, 2012.

[12] K. S. Essa, O. A. Gomaa, M. Elhussein, Y. Géraud, M. Diraison, and Z. E. Diab, “A prosperous and thorough analysis of gravity profiles for resources exploration utilizing the metaheuristic Bat Algorithm,” Sci. Rep., vol. 15, p. 5000, 2025.

[13] Y. Xian, P. Li, H. Peng, Z. Yu, Y. Xiang, and P. S. Yu, “Community detection in large-scale complex networks via structural entropy game,” in Proc. ACM Web Conf., 2025, pp. 3930–3941.

[14] M. Aslan and İ. Koç, “Modified Coot bird optimization algorithm for solving community detection problem in social networks,” Neural Comput. Appl., vol. 36, pp. 5595–5619, 2024.

[15] M. M. Abdulrahman, A. D. Abbood, and B. A. Attea, “Exploring signed social networks: Algorithms for community detection and structure analysis,” Khwarizmia, vol. 2023, pp. 37–45, 2023.