IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.14, No.3, July 2020, pp. 219~230
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.57834

Determining Community Structure and Modularity in Social Network using Genetic Algorithm

Taufan Bagus Dwi Putra Aditama*[1], Azhari SN[2]
[1]Master Program of Computer Science, FMIPA UGM, Yogyakarta, Indonesia
[2]Department of Computer Science and Electronics, FMIPA UGM, Yogyakarta, Indonesia
e-mail: ***[1]taufanbagusdpa@gmail.com**, [2]arisn@ugm.ac.id

Abstrak

Penelitian mengenai menentukan struktur komunitas dalam jaringan yang kompleks telah menarik banyak perhatian diberbagai aplikasi, seperti jaringan e-mail, jaringan sitasi, jaringan sosial, jaringan metabolisme, jaringan maskapai penerbangan, jaringan biologis, jaringan informasi, jaringan teknologi, dan jaringan komputer. Kepopuleran menentukan struktur komunitas disebabkan karena dapat menganalisis struktur, dan fungsional sebuah jaringan, yang mana jaringan atau komunitas itu sendiri dapat diartikan sebagai suatu node yang terkait erat didalam suatu jaringan informasi.

Sedangkan, untuk menentukan struktur komunitas dengan memaksimalkan nilai modularitas adalah hal yang sulit. Oleh karena itu, banyak penelitian memperkenalkan algoritma-algoritma baru untuk memecahkan masalah dalam menentukan struktur komunitas dan memaksimalkan nilai modularitas tersebut. GA dapat memberikan solusi yang efektif dengan menggabungkan eksplorasi dan eksploitasi. GA menggunakan metode komputasi berbasis populasi, dimana populasi terbaik didapatkan dari proses penyeleksian populasi secara acak, crossover, dan mutasi.

Penelitian ini berfokus pada Algoritma Genetika yang ditambahkan fitur clean up didalamnya. Hasil akhir penelitian ini merupakan hasil perbandingan nilai modularitas berdasarkan penentuan struktur komunitas dari Algoritma Genetika, Algoritma Girvan and Newman dan Algoritma Louvain. Hasil nilai modularitas terbaik diperoleh dengan menggunakan Algoritma Genetika yang mendapatkan hasil 0,6833 untuk dataset Zachary's karate club, 0,7446 untuk dataset Bootlenose dolphins, 0,7242 untuk dataset American college football, dan 0,5892 untuk dataset Books about US politics.

Kata kunci— *Community detection, genetic algorithm, social networks, community structure, modularity*

Abstract

Research on determining community structure in complex networks have attracted a lot of attention in various applications, such as email networks, social networks, social networks, metabolic networks, airline networks, biological networks, information networks, technology networks, and computer networks. The popularity determines the structure of a community because it can analyze the structure, and functionality of a network, in which the network or community itself can be interpreted as a node that is closely related to an information network.

Meanwhile, to determine the structure of the community by maximizing the value of modularity is difficult. Therefore, a lot of research introduces new algorithms to solve problems in determining community structure and maximizing the value of modularity. Genetic Algorithm can provide effective solutions by combining exploration and exploitation. Genetic Algorithm uses population-based computing methods, where the best population is obtained from the process of selecting random populations, crossovers, and mutations.

This study focuses on the Genetic Algorithm which added a cleanup feature in process. The final results of this study are the results of a comparison of modularity values based on the

Received July 14th,2020; Revised July 28th, 2020; Accepted July 30th, 2020

determination of the community structure of the Genetic Algorithm, Girvan and Newman Algorithm, and the Louvain Algorithm. The best modularity values were obtained using the Genetic Algorithm which obtained 0.6833 results for Zachary's karate club dataset, 0.7446 for the Bottlenose dolphins dataset, 0.7242 for the American college football dataset, and 0.5892 for the Books about US politics dataset.

Keywords—*Community detection, genetic algorithm, social networks, community structure, modularity*

1. INTRODUCTION

Research on determining community structure and modularity in complex networks have attracted much attention in various applications [1, 2], such as e-mail networks, citation networks, social networks, metabolic networks, airline networks, biological networks, information networks, technology networks, and computer networks [3]–[8]. Matters on popularity regarding determining modularity in the network are reinforced by Pattanayak et al. [9] and Zhang et al. [1], which states that popularity determines modularity in complex networks because it can analyze the structure and function of a community.

The community structure is often described as a node closely related to other nodes in a network, determining the community structure of the network is important for analyzing individual traits [10, 11]. Determining community structure in a network is the key to extracting useful information from a network, while the most popular algorithms for determining community structure are Girvan-Newman, Kernighan-Lin, Louvain [12], where most of the algorithms are based on determining the structure of the community and the value of modularity in a certain time because modularity (Q) is often used to measure the quality of the community, where the greater the value of modularity the better the structure of a community [13].

Twitter, Facebook, Google+, Sina Weibo, and Vkontakte are categorized as social networks and have grown rapidly in recent years [14, 15]. Social networks are one type of complex networks that are in harmony with the real world and can produce very large volumes of data [14, 16]. According to [17], social networks are not only networks that are in harmony with the real world but also have a lot of information and have the potential to provide us with more accurate and implicit knowledge.

The genetic algorithm discovered by Holland is an algorithm that combines exploration and exploitation, where exploration itself is to find new solutions from the solution area and exploits to find the most effective solution from previous searches [18, 19]. Genetic algorithm is very popular for search problems, optimization, and solutions to very complex problems because the genetic algorithm uses population-based computational methods obtained from random population selection processes, crossovers, and mutations [20]–[25].

Based on the explanation above, this research proposes a genetic algorithm to determine community structure in social networks, where the Genetic Algorithm will be modified by inserting a cleanup process and eliminating the selection process. The cleanup process in the Genetic Algorithm was chosen because in previous studies it was able to provide good performance in determining community structure [26, 27]. By utilizing the cleanup process in the genetic algorithm, it is expected to determine the community structure precisely.

Research on detecting communities in complex networks has attracted a lot of attention in recent years. Several methods of community detection have been developed. Community detection is carried out to analyze the structure and function of a network. Several studies have been conducted relating to detecting communities using Genetic Algorithms including [12], and [28]. The two studies have in common the method used is the genetic algorithm method, and the two studies also have differences in the dataset used. In research conducted by [28], the study uses the Kyoto Encyclopedia of Genes and Genomes (KEGG) biological network dataset, Biocarta, BBID, EC Number, and Reactome Pathway. Meanwhile, in research conducted by [12], the study uses a social network dataset that has been made by previous studies, namely

Zachary's karate club network [29], Bottlenose dolphin network [30],American college football network [31], and Books about US politics network [32], many studies using the four social networks and biological networks of the real world have been described above.

Still in the use of Genetic Algorithms to determine community structure, in contrast to the research of [12], and [28] research conducted by [26] has a different process from Genetic Algorithms in general, where Genetic Algorithms are generally population generation, selection, crossover, mutations, which will then get a new population. Meanwhile, research conducted by [26] has a process of generation, mutation, clean-up, crossover, clean-up, which will then get a new population. However, the research conducted by [26] has a similar process with the research conducted by [27], while in terms of the dataset used research conducted by [26] used three datasets namely Zachary's karate club network [29], American college football network [31], and Enrol Email Networks. Meanwhile, research conducted by [27] only used two datasets Zachary's karate club network [29], American college football network [31].

2. METHODS

In this paper, a genetic algorithm is proposed to determining community structure in social and complex networks, and genetic algorithm uses as the search engine and employs the network modularity as the fitness function to evolve the population. The genetic algorithm will be modified by inserting a cleanup process and eliminating the selection process. Next, genetic algorithm is described in detail.

3.1 Population initialization

The general understanding of individuals according to [33] is an individual who is in a group or an individual, while the understanding of individuals in genetic algorithms according to [34] to express one solution, individuals can be said to be the same as a chromosome, which is a collection of genes. This gene can be in the form of binary, float, and combinatorial.
According to [35] initial population, generation is a process that produces a number of individuals randomly (randomly). The size of the population depends on the problem to be solved and the type of genetic operator that will be applied. After the population is determined, initialization is carried out on the chromosomes in the population. Chromosome initialization is done randomly, while still paying attention to the solution domain and problem constraints. The formula below is for generating a random population in binary representation.

$$IPOP = \text{round}\{\text{random}(N_{ipop}, N_{bits})\} \qquad (1)$$

IPOP is a gene that will contain rounding from random numbers generated by N_{ipop} (*population number*) x N_{bits} (*number of genes in each chromosome*).

The population generation plan begins by declaring the pop_size variable, the generation array, and the graph. Where the pop_size variable is used to declare the number of nodes to be raised, the generation array is used to store the results of random node sampling, and the graph is used to find out how many nodes will be raised.

3.2 Fitness function

According to [36] fitness value is a value that shows the quality of chromosomes in the population, where the value of fitness is used as a measurement tool, the greater the fitness value, the better the individual is to be a potential solution, whereas according to [37] fitness value is a value that states whether or not a solution (individual), fitness value is used as a reference in achieving optimal value in genetic algorithms. Modularity is a measure of a network structure or graph, where networks with high modularity have solid connections between nodes in a module or community and have rare connections between nodes in different modules or communities, and modularity is often also used to measure structures detecting structures community in the network, while the most popular modularity function is the modularity created by [38]. Newman has made the formula of modularity Q. Formula 2 is a formula for undirected and unweighted networks [38].

$$Q = \sum_{c=1}^{n_c} \frac{l_c}{m} - \left(\frac{d_c}{2m}\right)^2 \qquad (2)$$

Where n_c is the number of communities (clusters), l_c the total number of edges in community c, and d_c is the total number of degrees node in the community c. Where the design is initiated by declaring a graph variable and then calculating modularity for each node.

3.3 Mutation

According to [39], mutations are an important part of genetic algorithms because they minimize the chances of searching trapped in local optima, whereas according to [40] mutations play a role in replacing genes lost from populations due to the selection process that allows reappearance of genes that do not appear at population initialization, where the chromosomes of children are mutated by adding a very small random value (mutation step size), with a low probability. There are several opinions about the value of this mutation rate, one of which is that the mutation rate of 1/n will give a pretty good result, those who argue the mutation rate does not depend on the size of the population. The mutation process does not have to be like that process, but there is another process that is by mutating the gene as much as the probability of mutation * the number of genes, where the position of the gene to be mutated is randomly selected [18, 40, 41].

A simple way to get binary mutations is to replace one or several gene values from a chromosome, the mutation steps are as follows [42]:

Step 1) Count the number of genes in the population (length of chromosomes multiplied by population size).

Step 2) Randomly select the gene to be mutated.

Step 3) Determine the chromosomes of the genes chosen to be mutated.

Step 4) Change the gene value (0 to 1, or 1 to 0) of the chromosome to be mutated.

The mutation process begins by declaring variables such as graphs, and adjacency matrix offspring, which are used for making adjacency matrix for offspring, after the adjacency matrix has been made, the next step is to repeat it for random chromosomes and genes, and if the chromosome index is the same with a random gene index, random genes will be re-selected, and if the chromosome index is different from the random gene index, then the contents of the chromosome index will be checked on the gene random index. If the contents of the chromosome index in the gene random index is equal to 1, then the value will be changed to 0, as well as the chromosome random index in the gene index. Meanwhile, if the chromosome index on the gene random index is equal to 0, then the value will be changed to 1, as well as the chromosome random index on the gene index.

3.4 Crossover

According to [43] crossover is a very important process in producing a new chromosome by crossing two or more parent chromosomes and is expected to create a new chromosome that is more efficient, whereas according to [44] mating (crossover) is operators of genetic algorithms that involve two parents to form new chromosomes, and allow new offspring to contain part of their parents and will result in much better performance compared to their parent. The one-point crossover made by [18] is a crossover that swaps the value of genes from a chromosome after certain points and is usually for chromosome representation in binary. At a one-point crossover, the crossover position k (k = 1, 2, ..., N-1) with N = the length of the chromosome selected randomly. Variables are exchanged between chromosomes at this point to produce children. Figure 1 is an illustration of one point crossover for the probability of a crossover = 0.9 [18].

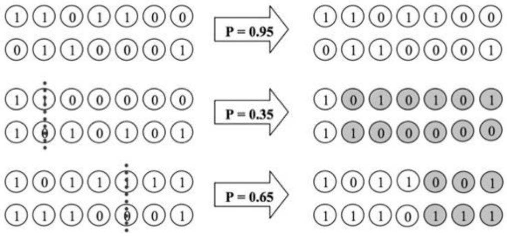

Figure 1 Illustration of One-Point Crossover

The crossover process begins by declaring offspring and probability variables, after that repeating as many nodes or individual i in a generation, then repeating as many probability values as specified j, if i and j are determined then crossbreeding between nodes or individuals to i with nodes or individuals to i + 1 in the gene to j, and if the index i + 1 is found to be an error then the process of interbreeding is done between nodes or individuals to i with nodes or individuals to i - 1 in the gene to j.

3.5 Clean-up step

According to [27] the clean-up process created by [26] is an efficient process for correcting errors in nodes that occupy the wrong community, where the node consists of parent and child vectors. Errors in the placement process in the clean-up process are detected from the fitness evaluation on the genetic algorithm. However, even though the overall fitness value is quite good, there may still be several misplaced nodes, but it does not necessarily affect the value of fitness in the whole community. The clean-up process is based on a new metric called community variance which aims to reduce all placement errors.

According to [26] community variance is a metric based on nodes in a community, where a community must contain more internal links in the community than external links with other communities or it can be concluded that the neighbors of a node are mostly must be in the same community. [26] define community variance $CV(i)$ where node i is the number of communities that are between neighbors and the node itself, where $CV(i)$ must be low for a good community structure, while the equation for finding community variance is as follows:

$$CV(i) = \frac{\sum(i,j) \in E^{f(i,j)}}{deg(i)}$$

$$\text{where } f(i,j) = \begin{cases} 1, commID(i) \neq commID(j) \\ 0, commID(i) = commID(j) \end{cases} \quad (3)$$

Where $f_{(i,j)}$ will be 1 if community i is not the same as community j and 0 if otherwise, whereas $deg(i)$ is the relationship of node i, E is edges, and community i is the community of node i.

The process of making community variance by randomly selecting nodes, if the node value is greater than the threshold value, where the threshold value is obtained from the constant calculation of a set of old nodes, then the node chosen randomly will be included in the same community, whereas if the threshold value is not met there are no operations performed on the nodes in the community [26].

3.6 Genetic algorithm framework

In general, the community determination step using genetic algorithm starts from generating population, where the results of the generation are in the form of an array which will then be converted into adjacency matrix, then the adjacency matrix will be used as chromosomes and genes to process the genetic algorithm, after that The modularity value is calculated based on the preprocessing graph data, then after the modularity value has been completed, the next step is to carry out the process of mutation, clean-up, crossover, clean-up, generation update, and if the update process of the modularity value has not exceeded the X_{best}variable value then it will return to the mutation process until the modularity value is more than the value of the X_{best}variable, and if the X_{best}variable value has been fulfilled then the system will automatically stop. The design of community structure determination using genetic algorithm is shown in Figure 2.

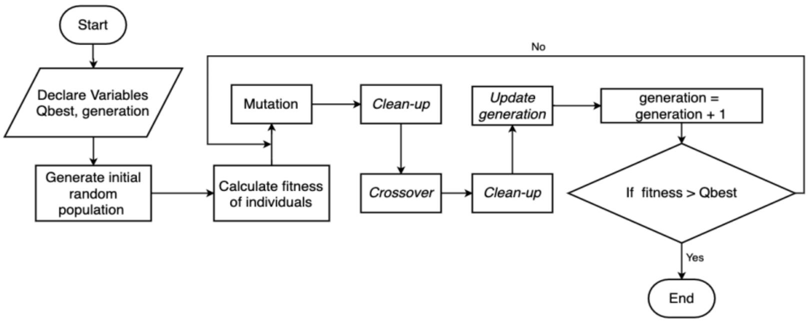
Start
Declare Variables Qbest, generation
Generate initial random population
Calculate fitness of individuals
Mutation
Clean-up
Crossover
Clean-up
Update generation
generation = generation + 1
If fitness > Qbest
No
Yes
End

Figure 2 Framework of genetic algorithm

Finally, the framework of genetic algorithm is described as follows:

Step 1) Set $t = 0$ where t denotes the generation number.

Step 2) Generate the initial population $P_0 = \{x_1, \dots, x_{np}\}$ by randomly sampling NP points from the search space S.

Step 3) Compute the network modularity value $Q(x_i)$ of each individual x_i in P_0.

Step 4) Perform the mutation operation (see Section 2C for details) on each individual x_i in P_t and obtain the mutant vectors $V_t = \{v_1, \dots, v_{np}\}$.

Step 5) Correct the mistakes in each mutant vector v_i in V_t by executing the cleanup operation (see Section 3E for details).

Step 6) Execute the modified one-point crossover (see Section 2D for details) on each mutant vector v_i in V_t and generate the trial vectors $U_t = \{u_1, \dots, u_{np}\}$.

Step 7) Correct the mistakes in each trail vector u_i in U_t by executing the clean-up operation (see Section 2E for details).

Step 8) Calculate the network modularity value $Q(u_i)$ of each trial vector u_i in U_t.

Step 9) Compare x_i with u_i (i = 1, . . . , NP) $\{i = 1, \dots, NP\}$ in terms of the network modularity value by following the equation (2), and put the winner into the next population P_{t+1}.

Step 10) Set $t = t + 1$.

Step 11) If the termination criterion is not satisfied, go to Step 4; otherwise, stop and output the best individual x_{best} in P_t.

3.7 Dataset

3.7.1 Zachary karate club networks

Zachary karate club is a network obtained from karate clubs which has 34 members, then becomes an internal problem between administrators and coaches of karate clubs, which causes club coaches to create new clubs with members of the original club. If represented in the graph there are 34 nodes, 78 edges, and four communities in the Zachary club karate network [29].

3.7.2 Bottlenose dolphins networks

In the 2003, Lusseau et al. [30] create a network by observing communities of dolphin species consisting of 62 different dolphin specials, where each dolphin represents a node, if two dolphins communicate frequently, they are connected by a line or edge, and on warming, there are 62 nodes and 160 edges. Where dolphins with female sex are pink, blue is male and green (unknown). Most links (70%) connect dolphins of the same sex.

3.7.3 American college football networks

American college football is a network created by Girvan and Newman in 2002 [31]. American college football network built from observations of university soccer leagues, nodes represent teams and edges represent play between the two teams, where the network has 115 nodes and 616 edges. Where teams are divided at the conference, and each team at the conference plays on average 4 times from the same conference and 7 other teams [45].

3.7.4 Books about US politics networks

Books about US politics is a network created by Newman in 2006 [32], the dataset was obtained from observations of a collection of social networking data, where the data are political book data purchased by Americans, consisting of 105 nodes and 441 edges. Where each node shows a political book, and the edge shows two books if bought by the same person [46].

3. RESULTS AND DISCUSSION

The implementation of this system uses the Python programming language. The equipment and materials used in this implementation are as follows:

4.1 Hardware

The hardware used in this study are presented in Table 1.

Table 1 Hardware

No.	Hardware	Information
1.	Laptop	Macbook Pro (Retina, 13-inch, Early 2015)
2.	Processor	2.7 GHz Intel Core i5
3.	Memory	8 GB 1867 MHz DDR3
4.	Resolution	13.3 inch (2560x1600)
5.	GPU	Inter Iris *Graphics* 6100 1536 MB

4.2 Software

The software used in this study are presented in Table 2.

Table 2 Software

No.	Software	Information
1.	Operating System	macOS High Sierra version 10.13.6
2.	Programming Language	Python
3.	Text editor	Jupyter Notebook, Atom, dan Microsoft Excel

4.3 Discussion

Genetic algorithm is stochastic optimization algorithms, we perform the experiments 10 times on these four networks among of them is Karate, Dolphins, Football, and Books, each test will be taken 100 times iteration to get N_{pr}, T_{avg}, T_{best} Q_{avg}, dan Q_{best} Where N_{pr} is the average number of communities, T_{avg} is the average amount of collection time, T_{best} is the best supporting time, Q_{avg} is the average value of modularity, and Q_{best} is the best modularity value, where the results are obtained with a 0.8 crossover probability parameter, the threshold for clean-up is 0.8, and the X_{best} limit is 0.9.

The best modularity results for the Karate dataset after testing were 0.6833, which results are shown in Figure 3 and the community structure is shown in Table 3.

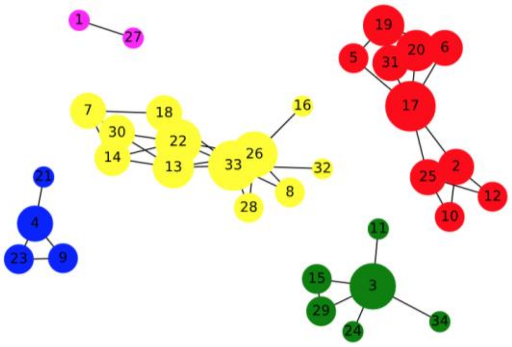

Figure 3 Result graph for Karate dataset

Based on Table 3, the nodes that occupy community 0 are nodes 2, 5, 6, 10, 12, 17, 19, 20, 25, and node 31, the nodes that occupy community 1 are nodes 7, 8, 13, 14, 16, 18 , 22, 26, 28, 30, 32, and 33, nodes occupying community 2 are nodes 3, 11, 15, 24, 29, and 34, nodes occupying community 3 are nodes 4, 9, 21, and 23, nodes that occupy community 4 are nodes 1 and 27.

Table 3 Community structure for Karate Dataset.

Community	Members	Modularity
0	2, 5, 6, 10, 12, 17, 19, 20, 25, 31	0,2269
1	7, 8, 13, 14, 16, 18, 22, 26, 28, 30, 32, 33	0,2424
2	3, 11, 15, 24, 29, 34	0,1134
3	4, 9, 21, 23	0,0794
4	1, 27	0,0212

The best modularity results for the Dolphins dataset after testing were 0.7446, which results are shown in Figure 4 and the community structure is shown in Table 4.

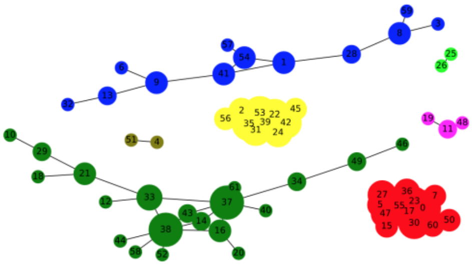

Figure 4 Result graph for Dolphins dataset

Based on Table 4, the results of the genetic algorithm show 8 communities for the Bootlenose dolphins dataset, while for nodes that occupy community 0 of them are nodes 0, 5, 7, 15, 17, 23, 27, 30, 36, 47, 50, 55, and 60, nodes that occupy community 1 of which are nodes 2, 22, 24, 31, 35, 39, 42, 45, 53, and 56, nodes that occupy community 2 of which are nodes 10, 12, 14, 16, 18, 20, 21, 29, 33, 34, 37, 38, 40, 43, 44, 46, 49, 52, 58, and 61, nodes that occupy community 3 are nodes 1, 3, 6, 8, 9, 13 , 28, 32, 41, 54, 57, and 59, nodes occupying community 4 are nodes 11, 19 and 48, nodes occupying community 5 are nodes 4, and 51, nodes occupying community 6 are nodes 25 and 26.

Table 4 Community structure for Dolphins Dataset.

Community	Members	Modularity
0	0, 5, 7, 15, 17, 23, 27, 30, 36, 47, 50, 55, 60	0,2233
1	2, 22, 24, 31, 35, 39, 42, 45, 53, 56	0,1996
2	10, 12, 14, 16, 18, 20, 21, 29, 33, 34, 37, 38, 40, 43, 44, 46, 49, 52, 58, 61	0,1740
3	1, 3, 6, 8, 9, 13, 28, 32, 41, 54, 57, 59	0,1074
4	11, 19, 48	0,0199
5	4, 51	0,0100
6	25, 26	0,0100

The best modularity results for the Football dataset after testing were 0.7242, which results are shown in Figure 5 and the community structure is shown in Table 5.

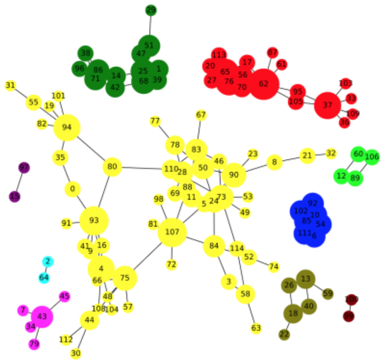

Figure 5 Result graph for Football dataset

Based on Table 5, nodes that occupy community 0 of which are nodes 17, 20, 27, 33, 36, 37, 56, 61, 62, 65, 70, 76, 87, 95, 103, 105, 109, and 113, the nodes that are occupying community 1 of which are nodes 0, 3, 4, 5, 8, 9, 11, 16, 19, 21, 23, 24, 28, 30, 31, 32, 35, 41, 44, 46, 48, 49, 50, 52, 53, 55, 57, 58, 63, 66, 67, 69, 72, 73, 74, 75, 77, 78, 80, 81, 82, 83, 84, 88, 90, 91, 93, 94, 98, 101, 104, 107, 108, 110, 112, and 114, nodes that occupy the community 2 of which are nodes 1, 14, 25, 29, 38, 39, 42, 47, 51, 68, 71, 86 , and 96, nodes that occupy community 3 of which are nodes 6, 10, 54, 85, 92, 102, and 111, nodes that occupy community 4 of which are nodes 7, 34, 43, 45, and 79, nodes that occupy community 5 of which are nodes 13, 18, 22, 26, 40, and 59, nodes that occupy community 6 of which are nodes 12, 60, 89, and 106, nodes that occupy community 7 of which are nodes 2, and 64, nodes that occupy community 8 of them are nodes 99, and 100, nodes that occupy community 9 of which are nodes 15, and 97.

Table 5 Community structure for Football Dataset.

Community	Members	Modularity
0	17, 20, 27, 33, 36, 37, 56, 61, 62, 65, 70, 76, 87, 95, 103, 105, 109, 113	0,1434
1	0, 3, 4, 5, 8, 9, 11, 16, 19, 21, 23, 24, 28, 30, 31, 32, 35, 41, 44, 46, 48, 49, 50, 52, 53, 55, 57, 58, 63, 66, 67, 69, 72, 73, 74, 75, 77, 78, 80, 81, 82, 83, 84, 88, 90, 91, 93, 94, 98, 101, 104, 107, 108, 110, 112, 114	0,2482
2	1, 14, 25, 29, 38, 39, 42, 47, 51, 68, 71, 86, 96	0,1245
3	6, 10, 54, 85, 92, 102, 111	0,0933
4	7, 34, 43, 45, 79	0,0270
5	13, 18, 22, 26, 40, 59	0,0399
6	12, 60, 89, 106	0,0270
7	2, 64	0,0068
8	99, 100	0,0068
9	15, 97	0,0068

The best modularity results for the Football dataset after testing were 0.5892, which results are shown in Figure 6 and the community structure is shown in Table 6.

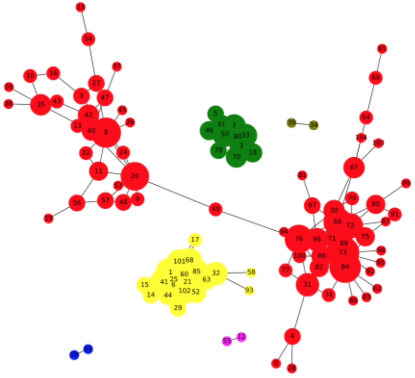

Figure 6 Result graph for Books dataset

Based on the results of Table 6, where the results from the Books about US politics dataset are divided into 6 communities. In community 0 it is occupied by nodes 0, 3, 4, 8, 9, 10, 11, 13, 16, 19, 20, 22, 23, 24, 26, 27, 28, 30, 31, 35, 36, 37, 38, 40, 42, 43, 45, 47, 48, 49, 53, 54, 56, 57, 62, 64, 65, 66, 67, 69, 71, 72, 73, 74, 75, 76, 77, 79, 81, 82, 83, 84, 86, 87, 88, 89, 90, 91, 92, 94, 95, 96, 97, 98, 99, 100, 103, and 104, community 1 is occupied by node 1, 6, 14, 15, 17, 21, 25, 29, 32, 41, 44, 52, 58, 60, 63, 68, 85, 93, 101, and 102, community 2 is occupied by 2, 5, 7, 18 , 33, 46, 50, 51, 70, 78, and 80, community 3 is occupied by node 59, and 61, community 4 is occupied by node 12, and 55, community 5 is occupied by nodes 34 and 39.

Table 6 Community structure for Books Dataset.

Community	Members	Modularity
0	0, 3, 4, 8, 9, 10, 11, 13, 16, 19, 20, 22, 23, 24, 26, 27, 28, 30, 31, 35, 36, 37, 38, 40, 42, 43, 45, 47, 48, 49, 53, 54, 56, 57, 62, 64, 65, 66, 67, 69, 71, 72, 73, 74, 75, 76, 77, 79, 81, 82, 83, 84, 86, 87, 88, 89, 90, 91, 92, 94, 95, 96, 97, 98, 99, 100, 103, 104	0,2460
1	1, 6, 14, 15, 17, 21, 25, 29, 32, 41, 44, 52, 58, 60, 63, 68, 85, 93, 101, 102	0,2040
2	2, 5, 7, 18, 33, 46, 50, 51, 70, 78, 80	0,1224
3	59, 61	0,0052
4	12, 55	0,0052
5	34, 39	0,0052

Experimental results of Genetic Algorithm tested on Zachary’s karate club network, Bottlenose dolphins networks, American college football network, and Books about US politics. We perform the experiments 10 times, and each experiment will be taken 100 times iterations to get N_{pr} is the average number of communities, T_{avg} is the average amount of collection time, T_{best} is the best supporting time, Q_{avg} is the average value of modularity, and Q_{best} is the best modularity. The test results are shown in Table 7 would be compared with Girvan and Newman algorithm and Louvain algorithm.

Table 7 The experimental results of Zachary’s karate club network, Bottlenose dolphins networks, American college football network, and Books about US politics.

Dataset	Algorithm	N_{pr}	T_{avg}	T_{best}	Q_{avg}	Q_{best}
Karate	Genetic	4,202	1,9240	1,3584	**0,4761**	**0,6833**
	Girvan and Newman	18	0,1850	0,1644	0,1879	0,4012
	Louvain	4	0,0336	**0,0207**	0,4151	0,4151
Dolphins	Genetic	6,572	5,6149	4,6281	**0,5749**	**0,7446**
	Girvan and Newman	32	0,6995	0,6643	0,2660	0,5193
	Louvain	4	0,0572	**0,0426**	0,5233	0,5233
Footballs	Genetic	5,641	26,4314	17,0402	0,4200	**0,7242**
	Girvan and Newman	36	9,1943	8,9122	0,3816	0,5996
	Louvain	10	0,1032	**0,0867**	**0,6044**	0,6044
Books	Genetic	4,198	32,5925	14,7021	0,4805	**0,5892**
	Girvan and Newman	36	4,9811	0,4860	0,3536	0,5168
	Louvain	4	0,1135	**0,0901**	**0,5265**	0,5265

Based on the test results Table 7, genetic algorithm tested using the karate, dolphins, football, and books dataset with 10 times the test, where each test iterates as much as 100 times. genetic algorithms get an average number of 4,202 communities for the karate dataset, 6,572 for the dolphins dataset, 5,641 for the football dataset, and 4,198 for the Books dataset. Whereas, for the test results using Girvan and Newman algorithm is 18 for the average number of karate dataset communities, 32 for the average number of dolphins dataset communities, 36 for the average number of football dataset communities, and books dataset. Meanwhile, for the results of testing using the Louvain algorithm, the average number of communities for the karate, dolphins, footballs, and books datasets is 4, 4, 10, and 4. Meanwhile, for the best processing time results are all obtained using the Louvain algorithm with a result of 0.0207 seconds for the karate dataset, 0.0426 seconds for the dolphins dataset, 0.0867 seconds for the football dataset, and 0.0901 seconds for the dataset books. The highest average modularity value for the karate and dolphins dataset was obtained using the genetic algorithm, which obtained 0.4761 and 0.5749 results. Whereas, for the football and books dataset, the highest average modularity values were obtained using the Louvain algorithm with the results of 0.6044 and 0.5265.

5. CONCLUSIONS

In this paper, we have introduced genetic algorithm to determining community structure in complex networks. The proposed genetic algorithm use clean-up process, which effectively corrects the mistakes of putting nodes into wrong communities in both mutant and trial vectors and improves the search ability of. Determining community structure with genetic algorithm can be applied with the results of 5 communities for the Zachary's karate club dataset, 7 communities for the Bootlenose dolphins dataset, 10 communities for the American college football dataset, and 6 communities for the Books about US politics dataset based on the best modularity values. Genetic Algorithms can be applied to increase the value of modularity, where testing uses Zachary's karate club dataset, Bootlenose dolphins, American college football, and Books about US politics get the best modularity values of 0.6833, 0.7446, 0.7242 and 0.5892. Where, the best modularity value of Genetic Algorithm is higher than Girvan and Newman Algorithm and Louvain Algorithm. Genetic Algorithms take a considerable amount of time when determining community structure, the best processing time is 1.3584 seconds for processing using the Karate dataset, 4.6281 seconds for the Dolphins dataset, 17.0402 seconds for the Footballs dataset, and 14.7021 seconds for the Books dataset. The processing results are much longer compared to Louvain Algorithm and Girvan and Newman Algorithm.

REFERENCES

[1] J. Zhang, X. Ding, and J. Yang, “Revealing the role of node similarity and community merging in community detection,” *Knowledge-Based Systems*, vol. 165, pp. 407–419, Feb. 2019.

[2] X. Chen and J. Li, “Community detection in complex networks using edge-deleting with restrictions,” *Physica A: Statistical Mechanics and its Applications*, vol. 519, pp. 181–194, Apr. 2019.

[3] H. Jin, W. Yu, and S. Li, “Graph regularized nonnegative matrix tri-factorization for overlapping community detection,” *Physica A: Statistical Mechanics and its Applications*, vol. 515, pp. 376–387, Feb. 2019.

[4] E. Jokar and M. Mosleh, “Community detection in social networks based on improved Label Propagation Algorithm and balanced link density,” *Physics Letters A*, vol. 383, no. 8, pp. 718–727, Feb. 2019.

[5] Y. Xu, “Community detection based on network communicability distance,” *Physica A: Statistical Mechanics and its Applications*, vol. 515, pp. 112–118, Feb. 2019.

[6] X. Zhou, K. Yang, Y. Xie, C. Yang, and T. Huang, “A novel modularity-based discrete state transition algorithm for community detection in networks,” *Neurocomputing*, vol. 334, pp. 89–99, Mar. 2019.

[7] M. Arasteh and S. Alizadeh, “A fast divisive community detection algorithm based on edge degree betweenness centrality,” *Appl Intell*, vol. 49, no. 2, pp. 689–702, Feb. 2019.

[8] Y. Lei, Y. Zhou, and J. Shi, “Overlapping communities detection of social network based on hybrid C-means clustering algorithm,” *Sustainable Cities and Society*, vol. 47, p. 101436, May 2019.

[9] H. S. Pattanayak, A. L. Sangal, and H. K. Verma, “Community detection in social networks based on fire propagation,” *Swarm and Evolutionary Computation*, vol. 44, pp. 31–48, Feb. 2019.

[10] M. Lu, Z. Qu, Z. Wang, and Z. Zhang, “Hete_MESE: Multi-Dimensional Community Detection Algorithm Based on Multiplex Network Extraction and Seed Expansion for Heterogeneous Information Networks,” *IEEE Access*, vol. 6, pp. 73965–73983, 2018.

[11] B. Rao, A. Mitra, and J. Mondal, “Algorithm for Retrieval of Sub-community Graph from a Compressed Community Graph Using Graph Mining Techniques,” *Procedia Computer Science*, vol. 57, pp. 678–685, 2015.

[12] H. Liu, F. Yang, and D. Liu, “Genetic algorithm optimizing modularity for community detection in complex networks,” in *2016 35th Chinese Control Conference (CCC)*, Chengdu, China, Jul. 2016, pp. 1252–1256.

[13] B. S. Khan and M. A. Niazi, “Network Community Detection: A Review and Visual Survey,” p. 39, 2017.

[14] G. Gadek, A. Pauchet, N. Malandain, K. Khelif, L. Vercouter, and S. Brunessaux, “Topical cohesion of communities on Twitter,” *Procedia Computer Science*, vol. 112, pp. 584–593, 2017.

[15] A. F. Hidayatullah and Azhari SN, “Analisis Sentimen dan klasifikasi kategori terhadap tokoh publik pada data Twitter menggunakan Naive Bayes Classifier,” Universitas Gadjah Mada, 2014.

[16] A. Chianese and F. Piccialli, “A Service Oriented Framework for Analysing Social Network Activities,” *Procedia Computer Science*, vol. 98, pp. 509–514, 2016.

[17] S. Guesmi, C. Trabelsi, and C. Latiri, “CoMRing: A Framework for Community Detection Based on Multi-relational Querying Exploration,” *Procedia Computer Science*, vol. 96, pp. 627–636, 2016.

[18] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1992.

[19] A. A. AbdulHamed, M. A. Tawfeek, and A. E. Keshk, “A genetic algorithm for service flow management with budget constraint in heterogeneous computing,” *Future Computing and Informatics Journal*, vol. 3, no. 2, pp. 341–347, Dec. 2018.

[20] A. Amirjanov, “Modeling the Dynamics of a Changing Range Genetic Algorithm,” *Procedia Computer Science*, vol. 102, pp. 570–577, 2016.

[21] M. Imran, N. A. Pambudi, and M. Farooq, “Thermal and hydraulic optimization of plate heat exchanger using multi objective genetic algorithm,” *Case Studies in Thermal Engineering*, vol. 10, pp. 570–578, Sep. 2017.

[22] K. Jankauskas, L. G. Papageorgiou, and S. S. Farid, “Fast genetic algorithm approaches to solving discrete-time mixed integer linear programming problems,” *Computers & Chemical Engineering*, vol. 121, pp. 212–223, Feb. 2019.

[23] G. Nagarajan, R. I. Minu, B. Muthukumar, V. Vedanarayanan, and S. D. Sundarsingh, "Hybrid Genetic Algorithm for Medical," *Procedia Computer Science*, vol. 85, pp. 455–462, 2016.

[24] P. Rai, A. Agrawal, M. L. Saini, C. Jodder, and A. G. Barman, "Volume optimization of helical gear with profile shift," *Procedia Computer Science*, vol. 133, pp. 718–724, 2018.

[25] M. Vizcaíno-González, J. Pineiro-Chousa, and M. Á. López-Cabarcos, "Analyzing the determinants of the voting behavior using a genetic algorithm," *European Research on Management and Business Economics*, vol. 22, no. 3, pp. 162–166, Sep. 2016.

[26] M. Tasgin and H. Bingol, "Community Detection in Complex Networks using Genetic Algorithm," Apr. 2006.

[27] G. Jia *et al.*, "Community Detection in Social and Biological Networks Using Differential Evolution," in *Learning and Intelligent Optimization*, vol. 7219, Y. Hamadi and M. Schoenauer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 71–85.

[28] M. B. M'Barek, A. Borgi, W. Bedhiafi, and S. B. Hmida, "Genetic Algorithm for Community Detection in Biological Networks," *Procedia Computer Science*, vol. 126, pp. 195–204, 2018.

[29] W. W. Zachary, "An Information Flow Model for Conflict and Fission in Small Groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, 1977.

[30] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, "The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations," *Behavioral Ecology and Sociobiology*, vol. 54, no. 4, pp. 396–405, Sep. 2003.

[31] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002.

[32] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, Jun. 2006.

[33] I. A. Hadi, "Pentingnya Pengenalan Tentang Perbedaan Individu Anak Dalam Efektifitas Pendidikan," *INSPIRASI: Jurnal Kajian dan Penelitian Pendidikan Islam*, vol. 1, no. 1, pp. 71–92, May 2017.

[34] M. Negnevitsky, *Artificial intelligence: a guide to intelligent systems*, 2nd ed. Harlow, England ; New York: Addison-Wesley, 2005.

[35] E. Suhartono, "Optimasi Penjadwalan Mata Kuliah Dengan Algoritma Genetika (Studi Kasus di AMIK JTC Semarang)," *INFOKAM*, vol. 11, no. 5, Dec. 2015.

[36] A. Rianawati and W. F. Mahmudy, "Implementasi Algoritma Genetika Untuk Optimasi Komposisi Makanan Bagi Penderita Diabetes Mellitus," vol. 5, no. 14, p. 12, 2015.

[37] I. M. B. Adnyana and N. K. D. A. Jayanti, "Implementasi Sistem Penjadwalan Ujian Akhir Semester menggunakan Algoritma Genetika," *CSRID*, vol. 6, no. 1, pp. 11–20, Oct. 2015.

[38] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, vol. 74, no. 3, p. 036104, Sep. 2006.

[39] M. B. Bashir and A. Nadeem, "Improved Genetic Algorithm to Reduce Mutation Testing Cost," *IEEE Access*, vol. 5, pp. 3657–3674, 2017.

[40] M. Gen, R. Cheng, and L. Lin, *Network models and optimization: multiobjective genetic algorithm approach*. London: Springer, 2008.

[41] D. E. Goldberg and R. Lingle Jr., "AllelesLociand the Traveling Salesman Problem," in *Proceedings of the 1st International Conference on Genetic Algorithms*, Hillsdale, NJ, USA, 1985, pp. 154–159.

[42] I. Iryanto and E. Ismantohadi, "Optimasi Pemilihan Barang Dagangan bagi Pedagang Keliling dengan Algoritma Genetika," *JTT (Jurnal Teknologi Terapan)*, vol. 3, no. 1, pp. 24-28–28, Mar. 2017.

[43] V. Kumar S G and R. Panneerselvam, "A Study of Crossover Operators for Genetic Algorithms to Solve VRP and its Variants and New Sinusoidal Motion Crossover Operator," *International Journal of Computational Intelligence Research*, vol. 13, pp. 1717–1733, 2017.

[44] A. B. A. Hassanat and E. Alkafaween, "On Enhancing Genetic Algorithms Using New Crossovers," p. 15, 2018.

[45] E. Raju, M. A. Hameed, and K. Sravanthi, "Detecting communities in social networks using unnormalized spectral clustering incorporated with Bisecting K-means," in *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pp. 1–5.

[46] A. Malathi and D. Radha, "Analysis and visualization of social media networks," in *2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, Oct. 2016, pp. 58–63.