



## Advancing Graph Theory with Genetic Algorithms: A Focus on Non-Inclusive Vertex Irregular Labeling

Kiswara Agung Santoso<sup>1,\*</sup>, Febyan Gilang Cristyanto<sup>1</sup>, Ikhsanul Halikin<sup>1</sup>,  
Ridho Alfarisi<sup>1</sup>

<sup>1</sup> *Department Mathematics, Jember University, Jember, Indonesia*

---

**Abstract.** Non-inclusive irregular vertex labeling is a labeling on a graph where the vertex labels are real numbers that have weights. The weight is defined as the sum of the labels of the connected nodes. The main problem in labeling graphs is how to find the formula so that the required labeling rules can be applied. To find this formula, researchers must try to label various kinds of graphs to find labeling patterns. A Heuristic Algorithm is an algorithm that can always provide solutions and is approximate. One type of heuristic algorithm is a genetic algorithm, where this algorithm will generate random numbers as candidate solutions and after going through several selection, evolution and evaluation processes the most appropriate value for the solution will be found. This research discusses the implementation of a genetic algorithm to label graphs (all types of graphs) in a computerized manner based on non-inclusive irregular labeling. It is hoped that this program will help researchers find non-inclusive irregular labeling formulas, without having to label graphs manually.

**2020 Mathematics Subject Classifications:** 05C78

**Key Words and Phrases:** Genetic Algorithm, Graph Theory, Non-inclusive Vertex Irregular Labeling

---

### 1. Introduction

Graph labeling is the assignment of labels, usually represented by integers, to elements of graph (vertices and edges). Marr dan Wallis [2] explained that if the domain of the function is the set of vertices, then it called vertex labeling. Slamin [3] introduced a concept of new labeling which is vertex irregular d-distance vertex labeling. Bong et al. [4] then generalized the concept to inclusive and non-inclusive vertex irregular d-distance vertex labeling. We will limit our research from distance d to distance 1 in this paper. Let k be positive integer. A non-inclusive vertex irregular labeling of graph  $G$  with vertex set  $V$  is an assignment :  $V \rightarrow \{1, 2, \dots, k\}$  such a way that the weight calculated are distinct.

---

\*Corresponding author.

DOI: <https://doi.org/10.29020/nybg.ejpam.v17i4.5538>

*Email addresses:* [kiswara.fmipa@unej.ac.id](mailto:kiswara.fmipa@unej.ac.id) (K. A. Santoso), [febyangilang@gmail.com](mailto:febyangilang@gmail.com) (F. G. Cristyanto), [ikhsan.fmipa@unej.ac.id](mailto:ikhsan.fmipa@unej.ac.id) (I. Halikin), [alfarisi.fkip@unej.ac.id](mailto:alfarisi.fkip@unej.ac.id) (R. Alfarisi)

The weight  $wt(x)$  of vertex  $x$  in  $G$  is defined as the sum of the labels of all vertices adjacent to  $x$  (distance 1 from  $x$ ),

$$wt(x) = \sum_{y \in N(x)} l(y)$$

with  $N(x)$  is the set  $x$  neighbors. The smallest integer  $k$  in this labeling is the distance irregularity strength of  $G$  and denoted as  $dis(G)$ . The difference between inclusive and non-inclusive vertex irregular labeling depend on the way to calculate the vertex weight whether the label of vertex we calculate its weight is included or not. Consequently, the first concept that Slamir introduced is categorized as non-inclusive vertex irregular labeling.

Representation of graph normally modeled by circles and lines, but this model is not suited for big sized graph. Another way to represent graph is with listing the adjacent vertices for every vertex in the graph, called adjacency matrix and denoted as  $A(G)$ . Elements of  $A(i, j) = 1$  if there exist edge that connect vertex  $i$  and vertex  $j$ , otherwise  $A(i, j) = 0$ . Li [5] explained that there are several ways to increase storage space efficiency in graph data. For example adjacency matrix of undirected simple graph always symmetric in its main diagonal. So, we can cut off the upper or lower triangular matrix to reduce its storage space.

Genetic algorithm is a searching method inspired from Darwin's theory of evolution, which states that the survival of a creature is influenced by the strong rule to produce offspring of the next generation [6]. Based on theory of evolution, genetic algorithm can be used to find a solution to optimization problems [7]. Genetic algorithm has been used widely in the fields of science and technology. Its famous applications are finding optimal solutions to multi-objective optimization problems [8]. Genetic algorithm consists of generate initial population chromosome, evaluate fitness function, and genetic operations which are selection, crossover, and mutation [9]. Genetic operation will produce new chromosome that will be the next generation [10]. This method already used in various fields, especially in optimization problems [11, 12, 13]. In [14], genetic algorithm also used in covid-19 problems with graph as the representation of the problem.

In previous research, genetic algorithms were used to solve the labeling of inclusive irregular vertex. In this research, genetic algorithms are used to solve the problem of labeling non-inclusive irregular vertex.

## 2. The Method

The following are steps to solve the problem of labeling non-inclusive irregular points using a genetic algorithm.

1. Change the graph to be labeled into an adjacency matrix ( $A$ ) as shown in Figure 1 as the example.
2. Generate chromosomes with  $n$  genes as random numbers  $(0-n)$ , with  $n$  being the number of points on the labeled label. Chromosomes here can be interpreted as possible solutions. Figure 2 shows possible solutions/chromosomes

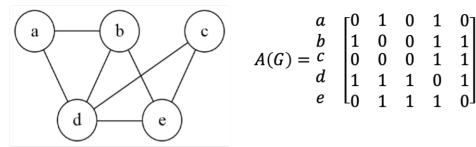


Figure 1: Graph and Adjacency matrix

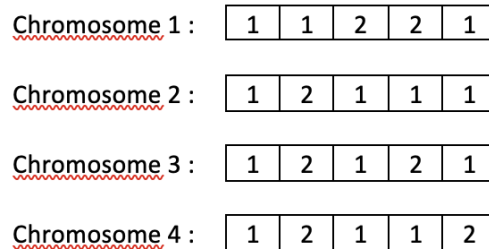


Figure 2: chromosome/possible solution

3. Find the weight of each point by multiplying the adjacency matrix by a random number (chromosome). Figure 3 is the process of calculating the weight value of each vertex.
4. Multiply each row of the adjacency matrix by the chromosome weight (A.\*Wi). Figure 4 shows the result of multiplying the adjacency matrix by chromosome W1.
5. Evaluate the results of the matrix multiplication in step 3. If in one row of the matrix, there are the same numbers > 0 then the chromosome is not a non-inclusive vertex labeling and the process continues to step 5. If in each row of the matrix, there are no same numbers > 0 then the chromosome is a non-inclusive vertex labeling solution and the process ends. In Figure 4, column 4 equal to column 5 shows that this chromosome is not a solution column 1 equal to column 3,so this chromosome is not a solution Based on the explanation above, it can be concluded that this chromosome is not a non-inclusive vertex labeling solution. Because the chromosome does not meet the labeling rules, the process continues to step 5.
6. Cross chromosomes 1 and 2 by exchanging the third gene with the first gene and the fourth gene with the third gene. The same thing is done for the cross chromosomes 3 and 4. Figure 5 shows the crossover process and results.

$$W(1) = (1 \ 1 \ 2 \ 2 \ 1) * \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} = 3 \ 4 \ 3 \ 5 \ 5$$

Figure 3: Weight of vertex

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} * (3 \ 4 \ 3 \ 5 \ 5) = \begin{bmatrix} 0 & 4 & 0 & 5 & 0 \\ 3 & 0 & 0 & 5 & 5 \\ 0 & 0 & 0 & 5 & 5 \\ 3 & 4 & 3 & 0 & 5 \\ 0 & 4 & 3 & 5 & 0 \end{bmatrix}$$

Figure 4: Evaluation Process



Figure 5: The Crossover Process and Result

7. We analyze chromosome 6 by calculating the weight matrix  $W(6)$  as in step 4.

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} * (3 \ 4 \ 2 \ 7 \ 5) = \begin{bmatrix} 0 & 4 & 0 & 7 & 0 \\ 3 & 0 & 0 & 7 & 5 \\ 0 & 0 & 0 & 7 & 5 \\ 3 & 4 & 2 & 0 & 5 \\ 0 & 4 & 2 & 7 & 0 \end{bmatrix}$$

From the results of the multiplication between the adjacency matrix and the weight matrix, it can be seen in Figure 6 that there are no identical numbers in each row, so it can be said that chromosome 6 is a solution to non-inclusive labeling.

If the analysis results show that the chromosome is not a non-inclusive labeling, repeat steps 1 to 6 until the iteration is fulfilled. If the iteration has been fulfilled but the chromosome that satisfies the labeling rule has not been found, then it can be said that the graph cannot be labeled according to the irregular non-inclusive rule.

### 3. Results

This section discusses implementation of genetic algorithm on non-inclusive vertex irregular labelings for graphs in several existing researches. We modeled the graph in the form of an adjacency matrix and stored the data using Microsoft Excel as an intermediary. Figure 6 shows the lower triangle of the adjacency matrix of the cycle graph  $C_{12}$ ,

With the help of MATLAB, the data is processed and genetic algorithm is applied to find the solution of non-inclusive irregular vertex labeling problems.

Figure 7 is the output of the genetic algorithm program for vertex irregular labeling using MATLAB. We set the number of chromosomes by 4 and mutation rate of 0.1. Figure

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	0												
2	1	0											
3	0	1	0										
4	0	0	1	0									
5	0	0	0	1	0								
6	0	0	0	0	1	0							
7	0	0	0	0	0	1	0						
8	0	0	0	0	0	0	1	0					
9	0	0	0	0	0	0	0	1	0				
10	0	0	0	0	0	0	0	0	1	0			
11	0	0	0	0	0	0	0	0	0	1	0		
12	1	0	0	0	0	0	0	0	0	0	1	0	
13													

Figure 6: Cycle graph  $C_{12}$  data

7 shows that there are 4 chromosomes in the last generation population which are potential solutions to the labeling problem. Of all chromosomes, the one that has the best fitness value and has no weight error is selected as the final solution. The following table are labelings using genetic algorithm application with the same rules as Figure 7 labeling in several previous researches [3, 15].

This labeling is not unique, meaning that a graph can have more than one possible non-inclusive irregular labeling. If in step 6 of the methodology above the process is not stopped even though the labeling has been found (the process stops only if the iteration is fulfilled) then there is a possibility that more than one labeling is found.

#### 4. Discussion

This research using the genetic algorithm to solve non-inclusive vertex irregular labeling of graph. There are some process for determine the non-inclusive vertex irregular labeling using the genetic algorithm namely the first step a graph  $G$  represented to be adjacency matrix, second step using genetic algorithm procedure. The final step, we get the non-inclusive vertex irregular labeling from genetic algorithm procedure but we need show using the lower bound of non-inclusive vertex irregular.

The selection process is carried out on each generation to selected the best chromosomes. This research is using tournament selection method as the selection process. The crossover process aim is to produce new offspring chromosomes. Chromosomes that have gone through selection process will be paired in a crossover process. The crossover process is conducted at a predetermined point. The mutation process conducted on each genes in offspring chromosomes based on the predetermined mutation rate, which is 0.1. A random number will be taken and if the number is less than or equal to the mutation rate, the gene will mutated. After the mutation process is complete, 2 offering chromosomes are obtained which will enter the new population. The previous population will erase the 2 worst chromosomes based on their fitness value and replace them with 2 offspring chromosomes after the mutation process. The initial generation population or generation 0 will

```

Labeling Result: [ 2  5  2  5  4  4  7  1  7  1  6  2 ]
Vertices Weight: [ 7  4 10  6  9 11  5 14  2 13  3  8 ]
-----
Chromosome      : 1
Graph labeling  : [ 2  5  2  4  4  4  7  1  7  1  6  2 ]
Vertices Weight: [ 7  4  9  6  8 11  5 14  2 13  3  8 ]

Chromosome      : 2
Graph labeling  : [ 2  5  2  4  4  4  7  1  7  1  6  2 ]
Vertices Weight: [ 7  4  9  6  8 11  5 14  2 13  3  8 ]

Chromosome      : 3
Graph labeling  : [ 2  5  2  5  4  4  7  1  7  1  6  2 ]
Vertices Weight: [ 7  4 10  6  9 11  5 14  2 13  3  8 ]

Chromosome      : 4
Graph labeling  : [ 2  5  2  4  4  4  7  1  7  1  6  2 ]
Vertices Weight: [ 7  4  9  6  8 11  5 14  2 13  3  8 ]

```

Figure 7: The vertex weight of  $C_{12}$  in MATLAB

become the population of generation 1 after the genetic algorithm process.

Each generation will choose one chromosome that has the highest fitness value as the solution labeling. The first generation result of genetic algorithm was not producing good results. All chromosomes still have labeling errors that cause repeated weights. The evolution process will be repeated until it gets the result that close or equal to optimal. If the fitness value of solution of one generation is equal to the previous generation, then the fitness value of solution is said to be convergent. If the convergence has reached a certain value but the labeling errors still exist, then it will be assumed that the graph  $G$  has no labeling with  $dis(G) = k$ . If this happens, then the algorithm needs to increase its  $dis(G)$  to  $dis(G) = k + 1$ . Changes in the value will occur in the next generation mutation process. This process will always repeated until the optimal solution is found. In this case the process of evolution stops at the 9th generation.

The application of genetic algorithms to non-inclusive vertex irregular labeling of graph resulting produce a solution to the problem. The pursuit for labeling solutions using human assistance can take a relatively long time, especially in graphs with a large number of vertices and without patterns. Genetic algorithms can be applied with the help of computer as one option to find solutions to problems.

Table 1. Results of genetic algorithms implementation in previous researchs

Previous Researches		Genetic Algorithm Solution	Processing Time Estimation
Graph Types	dis		
$K_3$	3	3 1 2	10 s
$K_4$	4	2 1 3 4	33 s
$K_5$	5	5 1 3 4 2	1 min 24 s
$K_6$	6	2 5 4 1 6 3	3 min 29 s
$P_3$	$\infty$	-	3 s
$P_4$	2	2 2 1 2	8 s
$P_5$	3	1 1 3 2 3	24 s
$P_6$	3	3 1 3 2 2 2	39 s
$P_7$	4	4 1 2 2 3 2 4	1 min 36 s
$C_4$	$\infty$	-	5 s
$C_5$	3	1 3 1 3 2	20 s
$C_6$	4	1 4 2 5 3 2	1 min 30 s
$C_7$	5	4 1 5 1 5 4 2	2 min 2 s
$C_8$	5	1 5 2 1 3 1 3 4	2 min 18 s
$C_9$	5	3 1 1 1 5 2 5 5 4	3 min
$C_{10}$	6	1 4 1 4 3 6 3 6 2 3	10 min 47 s
$C_{11}$	7	2 5 3 7 6 4 7 6 7 1 1	9 min 44 s
$C_{12}$	7	2 7 3 7 4 5 7 5 6 1 2 2	10 min 51 s
$C_{13}$	7	2 1 1 1 6 5 7 5 7 6 5 2 4	13 min 39 s
$W_5$	3	1 3 1 3 2 1	35 s
$W_6$	4	4 3 2 4 1 5 3	2 min 4 s
$W_7$	5	4 3 3 2 4 1 5 3	2 min 37 s
$W_8$	5	4 3 5 2 5 1 3 1 4	3 min 17 s
$W_9$	5	5 2 1 1 1 5 4 5 5 3	4 min 7 s
$W_{10}$	6	2 6 3 5 3 4 2 4 1 6 1	9 min 37 s
$W_{11}$	7	1 2 7 3 7 3 1 4 1 6 2 5	12 min 45 s
$W_{12}$	7	4 4 6 3 6 7 5 2 1 2 7 1 7	14 min 31 s
$W_{13}$	7	7 1 7 4 6 7 1 7 5 2 7 1 3 1	17 min 12 s

### 5. Conclusion

The Based on the explanation above, it can be concluded that; Genetic algorithms can be used to label the vertex of graphs according to non-inclusive irregular rules. If conventional methods take a very long time to label graphs, then with this algorithm the time needed is only a matter of seconds. If in conventional methods the labeling pattern can only be used for specific graphs, then with this method the labeling can be applied to all types of graphs or is universal.

### Acknowledgment

We gratefully acknowledge the support from University of Jember of year 2024.

### References

- [1] B. H. Gwee, M. H. Lim, and J. S. Ho, Solving four-colouring map problem using genetic algorithm, in proceedings of first new zealand international two-stream conference on artificial neural networks and expert systems, New Zealand. (1993), 332-333.
- [2] M. A. Marr and W. D. Wallis, Magic graph, Basel: Birkhäuser. (2001).
- [3] Slamin. On distance irregular labelling of graphs. Far East Journal of Mathematical Sciences. 102 (2017), 919-932.
- [4] N. H. Bong, Y. Lin, and Slamin, On inclusive and non-inclusive vertex irregular d-distance vertex labelings, J. Comb. (2020).
- [5] Santoso K, Setiawan B, Kusbudiono K, Application of Genetic Algorithm on Inclusive 5 of a Graph, Indonesian Journal of Pure and Applied Mathematics. (2022).
- [6] H. Li. A visual canonical adjacency matrix for graphs, Lowell: Department of Computer Science. University Of Massachusetts Lowell. (2009).
- [7] H. Tomori and K. Hiyoshi, Control of pneumatic artificial muscles using local cyclic inputs and genetic algorithm, Actuators. 7 (2018).
- [8] T. S. Widodo, Evolutionary computation genetic algorithm, genetic programming, and evolutionary programming, Yogyakarta: Graha. (2012).
- [9] T. M. Hung, Optimal selection for an air suspension system on buses through a unique high level parameter in genetic algorithms, Heliyon. 8 (2022).
- [10] O. A. Alimi, K. Ouahada, A. M. Abu-Mahfouz, S. Rimer, Power system events classification using genetic algorithm based feature weighting technique for support vector machine, Heliyon. 7 (2020).
- [11] M. Putnins and J. P. Androulakis, Self-selection of evolutionary strategies: adaptive versus non-adaptive forces, Heliyon. 7 (2021).
- [12] X. Shi, W. Long, Y. Li, and D. Deng, Multi-population genetic algorithm with er network for solving flexible job shop scheduling problems, PLOS ONE. 15, (2020).
- [13] K. T. Hidayat, R. Arifudin, and Alamsyah, Genetic algorithm for relational database optimization in reducing query execution time, Scientific Journal of Informatics. 5 (2018).
- [14] L. T. Utomoa, F. Wahyudi, Implementation of genetic algorithm in optimization of subject scheduling, JEEMECS. 3 (2020), 125-134.
- [15] T. Schlick, Q. Zhu, S. Jain, and S. Yan, Structure-altering mutations of the sars-cov-2 frameshifting rna element, Biophysical Journal. 119 (2021), 1040–1053.



- [16] N. H. Bong, Y. Lin, and Slammin, On distance-irregular labelings of cycles and wheels, *Australasian Journal Of Combinatorics*. 69 (2017), 315-322.
- [17] L. A. Anggraini, I. Rosyida, and T. S. N. Asih, Graph coloring problem solving using genetic algorithm, *UNNES Journal of Mathematics*. 8 (2019), 30-39.