Optimizing Book Delivery Routes Using Genetic Algorithms: Case Study of Erlangga Publisher Yogyakarta Branch

Damba Saputra Rosa[1], Asep Setiyono[2], Yohanes Renaldi Rio Santana[3], Paulus Mudjihartono[*4]
Program Studi Magister Informatika, Fakultas Teknologi Industri,
Universitas Universitas Atma Jaya Yogyakarta
Jl. Babarsari 44, Sleman 55281, DI Yogyakarta, Indonesia
Email: [1]235312663@students.uajy.ac.id, [2]235312541@students.uajy.ac.id,
[3]235312544@students.uajy.ac.id, [4]paulus.mudjihartono@uajy.ac.id

***Abstrak*. *Optimasi Rute Pengiriman Buku Menggunakan Algoritma Genetika: Studi Kasus Penerbit Erlangga Cabang Yogyakarta*.** Penelitian ini bertujuan untuk mencari rute terpendek pengiriman buku dengan pendekatan *Travelling Salesperson Problem* (TSP) yang diselesaikan dengan *Genetic Algorithm* (GA). Jarak antara pasangan lokasi (asal dan tujuan) akan diketahui dengan menggunakan garis bujur dan lintang sebagai koordinat lokasi (tempat dimana buku harus disampaikan dan perjalanan dilanjutkan). Jaringan koordinat lokasi ini kemudian dipandang sebagai TSP, yang membutuhkan GA untuk menyelesaikan jalur terpendeknya. Dengan menjalankan program sampai 100 iterasi, penelitian ini menghasilkan rute terpendek, 356 km secara keseluruhan. Di antara penelitian-penelitian sebelumnya, penelitian ini memiliki keunikan tersendiri, terutama ketika permasalahannya dipandang sebagai TSP dan ketika menyangkut mekanisme *crossover*, hal ini cukup jarang terjadi. Apalagi kasus penerbit Erlangga merupakan kasus pertama yang menggunakan GA.
Kata Kunci: *Traveling Salesperson Problem (TSP)*, algoritma Genetika, rute terpendek

Abstract. Optimizing Book Delivery Routes Using Genetic Algorithms: Case Study of Erlangga Publisher Yogyakarta Branch. This research aims to find the shortest route for book delivery using the Traveling Salesperson Problem (TSP) approach that is solved by a Genetic Algorithm (GA). The distance between the pair of locations will be known by using the longitude and latitude as the coordinates of the location (the place where books must be dropped and the trip continues). This network of the coordinates of locations is then viewed as TSP, which needs GA to solve the shortest path. Running the program for up to 100 iterations, this study resulted in the shortest route, 356 km in a whole route. Among the previous research, this research has its uniqueness, especially when the problem is viewed as a TSP, and when it comes to the crossover mechanism, it is quite rare. Moreover, the case of the Erlangga publisher is the first case that has used the GA.
Keywords: *Traveling Salesperson Problem (TSP), Genetic Algorithm, the best route*

1. Introduction

Erlangga Group, founded on April 30, 1952, has colored the history of book publishing in Indonesia. With a long journey of 71 years, this company first started its journey by focusing on publishing textbooks. However, Erlangga Group has become a main pillar of the Indonesian publishing industry. Book delivery is a crucial aspect of the book publishing industry. Efficiency in book delivery is important for reducing costs, increasing customer satisfaction, and maintaining competitiveness. Some of the variables involved in the delivery problem involve the distance between locations, the number of orders, the type of vehicle, delivery time, and the customer's immediate needs. Delays in one customer can impact the service time of the next customer if the vehicle passes through more than one customer in the delivery route. This delay has the potential to make consumers dissatisfied and feel disadvantaged [1]. The Traveling Salesperson Problem

(TSP) is an optimization problem that aims to find the shortest route to visit a certain number of location points. In this context, a salesman travels and finds an optimal solution to cover the distance from the Erlangga publishing office to each school and bookstore location only once and returns to the starting point. The main goal of TSP is to find the most efficient and shortest route so that the total distance traveled by the salesperson is minimal [2]. To reduce distance and travel time, it is important to determine the optimal distribution route [3]. Planning optimal delivery routes and allocating resources appropriately is a complex task. Genetic Algorithms (GA) are proven to be an effective tool for solving optimization problems, including delivery problems [4], [5]. GA can help companies reduce shipping costs, improve route efficiency, reduce carbon emissions, and provide better customer service.

The primary aim of this research is to develop and implement a solution to optimize the delivery routes for Erlangga Group, thereby minimizing both distance traveled and delivery time. By applying GA to the TSP in the context of Erlangga Group's book distribution network, this research seeks to identify the most efficient routes for book deliveries. The study will evaluate various factors such as the number of orders, delivery time, vehicle types, and customer needs. By leveraging GA, the research aims to reduce operational costs, enhance customer satisfaction, and contribute to environmental sustainability by lowering carbon emissions. Through careful analysis and implementation of this approach, the research intends to offer a robust solution that can be applied to real-world scenarios, thereby improving the overall efficiency and effectiveness of Erlangga Group's delivery system.

2. Previous Works

Several previous studies have been conducted by Ihsani et al. with the title "Implementation of Genetic Algorithms in Determining Optimal Routes for Web-Based Post Office Couriers (Wates Post Office Case Study)." This research produces a web-based application that aims to determine the best route for Wates Post Office couriers [6]. Other research was also conducted by Zahro and Wahyuni with the title “Optimizing Package Delivery Routes Using the Genetic Algorithm Method with the Shortest Route Output for the TSP” [7]. Hidayat et al. conducted research entitled “Optimizing Coconut Delivery Routes in Traditional Markets in Mojokerto Regency using Genetic Algorithms”. The research results in the optimization of coconut distribution routes to traditional markets throughout the Mojokerto district [8]. Priandani and Mahmudy conducted research entitled “Optimizing the Traveling Salesperson Problem with Time Windows (TSP-TW) on Scheduling Tourist Route Packages on the island of Bali using a Genetic Algorithm” with the results of the most optimal travel schedule with the shortest and most timely routes for tourism on the island of Bali. By implementing Genetic Algorithms in book shipping, this research provides practical insights that can be applied in various shipping industries.

In the studies mentioned, Genetic Algorithms (GA) have been utilized to effectively solve the problem of route optimization by simulating the process of natural evolution. The GA starts by generating an initial population of potential solutions, representing different routes. These solutions are evaluated based on a fitness function, often considering factors such as the total distance, time, or cost. Through iterative processes of selection, crossover, and mutation, the algorithm refines these solutions, gradually improving their fitness. For instance, in the study by Ihsani et al., the GA was employed to optimize courier routes for the Wates Post Office, resulting in minimized travel distances. Similarly, Zahro and Wahyuni used GA to generate the shortest delivery routes in a TSP context. The research by Hidayat et al. successfully optimized coconut delivery routes, while Priandani and Mahmudy applied GA to create the most efficient tourist routes in Bali, considering both time and distance constraints. These studies demonstrate GA's flexibility and effectiveness in tackling complex routing problems across various industries.

3. Methodology

The methodological steps within this research involved identifying problems, collecting data, applying the Traveling Salesperson Problem (TSP) method and Genetic Algorithms (GA), and drawing conclusions.

3.1. Identify the Problem

Erlangga Group, as a leading book publisher in Indonesia, faces significant challenges in the distribution of its books, particularly in ensuring timely and efficient delivery to schools across the country. The vast network of schools spread across various regions, coupled with varying order quantities and delivery schedules, complicates the logistics involved. One of the major issues is the inefficient routing of delivery vehicles, which often results in longer travel distances, higher fuel costs, and delays in getting books to their destinations. These inefficiencies not only increase operational costs but also negatively impact customer satisfaction, as schools depend on receiving educational materials on time.

Moreover, the delivery process must account for multiple variables, such as the distance between schools, the number of books ordered, the type of vehicles used, and the specific time windows for delivery. Failing to optimize these factors can lead to bottlenecks, where delays in one delivery can cascade into subsequent deliveries, further exacerbating the problem. As Erlangga Group continues to expand its reach, the complexity of coordinating deliveries efficiently becomes even more pronounced, necessitating a more sophisticated approach to route optimization. This research seeks to address these challenges by implementing the Multiple Traveling Salesperson Problem (MTSP) method and Genetic Algorithm (GA) to find the most efficient delivery routes, ultimately reducing costs and improving delivery times.

3.2. Data Collection

In the context of this research, the author chose to carry out semi-structured interviews with informants. This approach was chosen to facilitate the collection of answers that might arise from unplanned questions, thereby enabling the processing of initial samples as primary data. The decision to use semi-structured interviews was taken to gain a deeper understanding. To obtain samples, the author interviewed the head of logistics of PT Publisher Erlangga, Yogyakarta branch. The data collected in this research includes comprehensive details about book delivery schedules to schools, specifying the dates, times, names of school, and routes involved in the distribution process. This information provides a clear overview of the logistics necessary for ensuring timely deliveries. Additionally, geocoded coordinates obtained from Google Maps were used to identify the exact geographic locations of the schools and possibly bookstores. These coordinates, represented as latitude and longitude points, allow for precise mapping and analysis of the delivery routes. Together, the schedules and geocoded data offer a thorough understanding of both the timing and spatial aspects of book distribution, essential for effective logistical planning.

3.3. Travelling Salesperson Problem (TSP)

The Traveling Salesman Problem (TSP) is one of the most extensively studied problems in combinatorial optimization, largely because of its theoretical significance, complexity challenges, and practical applications in numerous real-world scenarios. One of them is an optimization problem that aims to determine the shortest route starting from the starting point, involving visiting a group of points exactly once and ending by returning to the starting point [9]. TSP can be interpreted as an optimization problem that can be applied to various activities, including package delivery [7], can also be used to handle production scheduling problems [10]. In other words, the TSP is broadly relevant in various contexts where travel optimization or route management is critical.

3.4. Genetic Algorithms

Genetic Algorithms are a search and optimization approach inspired by Darwin's theory of evolution. The basic principle of this algorithm is to simulate the evolution of a population of individuals representing a particular search problem by prioritizing the survival and reproduction of the most suitable individuals [11]. Genetic Algorithms are often used to handle complex problems, especially when the objective function does not have friendly properties, such as discontinuity or non-differentiability, or when knowledge about the domain is very limited or even nonexistent. The advantages of Genetic Algorithms become more apparent in such conditions, where design and analysis become difficult . In the Genetic Algorithm, one potential optimal solution is referred to as an entity. A collection of entities forms a population consisting of several possible solutions [12]. In this situation, the process of selection, combination, and mutation occurs to create the best-quality chromosomes. This concept can be used to find solutions to real-world problems. Genetic algorithms are suitable for handling the characteristics of problems that have many possible solutions and require "real-time" solutions, which means finding solutions quickly so that they can be implemented immediately for rapidly changing problems [3]. The details of the Genetic Algorithm are shown in Fig 1.

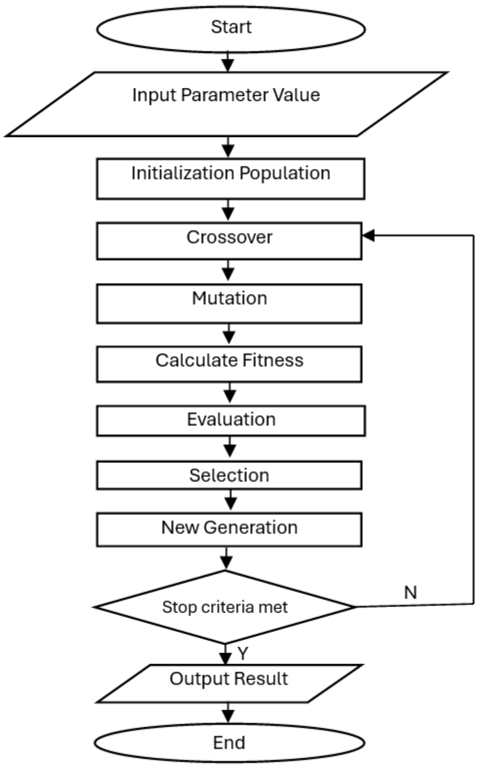

Figure 1. The Steps in The Genetic Algorithm Method

The steps for solving Genetic Algorithm problems are as follows. Selection is the stage where individuals are sorted based on fitness values, and this sorting process utilizes a selection method . The fitness function is evaluated by measuring an individual's performance, which is the fitness of a chromosome, which, in turn, may be retained or eliminated [13]. The steps in GA are as follows: (1) Set the required parameter values: the population's initial size, the problem's dimension size, and the stop criteria. (2) Randomly generate the initial population of chromosomes. Each chromosome forms a band of genes that represent the dimension. (3) Crossover is the stage where cross-gene from two different chromosomes occurs in each randomly selected chromosome, with the aim of the reproduction process. (4) Mutation is the process of swapping genes on a chromosome at a certain point. This stage involves randomly selecting some genes on a chromosome and swapping them according to a certain pattern, resulting in a new chromosome. (5) Evaluation is a step to calculate the fitness value of each chromosome. (6) Selection is the stage where individuals are sorted based on fitness values. This process determines which chromosome should be living any longer or die.

4. Result and Discussion

Depending on the type of data used in the chromosome representation, these genes can be binary, float, integer, or character values [14]. In this study, the genetic elements employed serve as markers akin to waypoints, denoted by geocoded coordinates expressed in longitude and latitude. Establishing the starting gene and the population's initial point, signifying the delivery location, requires geocoded coordinates obtainable through Google Maps. It was chosen 25 location coordinate points as samples. Table 1 details the geocodes designated for initial experimental sampling.

Table 1. **Location Geocode**

No	Location	Latitude	Longitude
1	SD JETIS HARJO	-7.7708123	110.3661234
2	SD BLUNYAH REJO	-7.7765877	110.3653571
3	SD VIDYASANA QASANA	-7.7839437	110.3580491
4	SD TEGALREJO 2	-7.7914841	110.3504981
5	SMA MUH. 2 YOGYAKARTA	-7.7993182	110.3803458
6	SD MUH. PURWOREJO	-7.7089102	110.0204728
7	MI MUH. SEMAMPIR	-7.4041347	109.6713651
8	SDN 01 MANDIRAJA WETAN	-7.4526246	109.5222317
9	SDN 01 JALATUNDA	-7.4841318	109.5079421
10	MI KEC. MANDIRAJA	-7.4649297	109.4711792
11	SMP MA'ARIF GAMPING	-7.7671151	110.3318020
12	SMP MUH. 2 GAMPING	-7.7756049	110.3360212
13	MAN 3 SLEMAN	-7.7681543	110.3629060
14	MTS PAMULANGAN	-7.7609841	110.3237986
15	SMP IT BAKTI INSANI	-7.7131962	110.3290734
16	MTSN 04 SLEMAN	-7.7135607	110.3392451
17	MAN 5 SLEMAN	-7.6659195	110.3297973
18	MTSN 8 SLEMAN	-7.7753409	110.4752326
19	TB. SB	-7.8035076	110.3647648
20	TB. TOGA MAS AFANDI	-7.7591050	110.3956180
21	TB. SOCIAL AGENCY JAKAL	-7.7327029	110.3938886
22	TB. SOCIAL AGENCY GODEAN	-7.7806939	110.3446693
23	TB. SOCIAL AGENCY AMBARUKMO	-7.7835516	110.3974609
24	TB. TOGA MAS SUROTO	-7.7846244	110.3389385
25	TB. SOCIAL AGENCY SAGAN	-7.7809186	110.3768411

Utilizing Google Maps' route functionality facilitates determining distances between distinct location points. The distances between these points are cumulatively computed for each individual, and the outcomes are then tallied to generate a set of fitness values, as illustrated in Table 2.

Table 2. Distance Table Between Points

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**
0	0	8.9	9.7	9	8.9	4.1	54.	127	147	142	148	13.	11.	9.8	14.	19.	18.	23.	13.	5.3	9.3	13.	9.4	5.9	8.5	8.1
1	8.9	0	0.9	2	2.3	4.4	47.	118	138	139	136	9.2	4.7	1.8	10.	11.	9.4	15.	14.	3.8	5	6.9	3	4.1	1.9	2.3
2	9.7	0.9	0	3.1	4	6.2	49.	119	136	139	136	7.2	5.4	0.9	8.4	10.	8.8	13.	15.	4.6	5.6	6.8	3.5	5.2	2.6	3.3
3	9	2	3.1	0	2.2	5.4	46.	119	137	138	137	4.8	3.4	2.8	6.1	11.	9.5	14.	15.	3.9	1.9	7.8	1.6	5.7	1.9	3.9
4	8.9	2.3	4	2.2	0	5.1	46.	116	141	136	142	5	3.5	4.8	6.2	11.	11.	16.	16.	3.3	8.3	9.8	1.8	6.4	3.7	4.6
5	4.1	4.4	6.2	5.4	5.1	0	55.	124	145	139	145	9.1	7.6	6.2	10.	15.	14.	18.	14.	2.7	5.3	9.1	5.9	3.3	3.1	2.7
6	54.	47.	49.	46.	46.	55.	0	80.	98.	82.	98.	43.	44	51.	43.	47.	50	49.	62	49.	53.	54.	44.	51.	48	50
7	127	118	119	119	116	124	80.	0	18.	24.	18.	113	116	116	112	108	110	105	132	122	118	116	116	122	120	121
8	147	138	136	137	141	145	98.	18.	0	6	1	131	142	134	130	125	128	123	150	143	136	134	143	143	137	138
9	142	139	139	138	136	139	82.	24.	6	0	7.1	136	136	140	136	131	133	128	152	137	145	146	137	142	139	140
1	148	136	136	137	142	145	98.	18.	1	7.1	0	131	142	134	130	125	128	123	150	143	136	134	143	148	137	138
1	13.	9.2	7.2	4.8	5	9.1	43.	113	131	136	131	0	2.6	7.5	1.3	7.4	7.6	14.	22.	9.3	9.4	10.	4.1	13.	7.4	9.4
1	11.	4.7	5.4	3.4	3.5	7.6	44	116	142	136	142	2.6	0	5.9	3.7	9.2	9.4	14.	19	7.9	10	11.	1.8	8.8	5	7
1	9.8	1.8	0.9	2.8	4.8	6.2	51.	116	134	140	134	7.5	5.9	0	8.3	9.7	7.9	12.	16.	5.8	5.7	6.9	3.5	6.1	3.5	4.2
1	14.	10.	8.4	6.1	6.2	10.	43.	112	130	136	130	1.3	3.7	8.3	0	6.6	6.8	13.	23.	9.8	10.	11.	4.7	11.	7.9	9.9
1	19.	11.	10.	11.	11.	15.	47.	108	125	131	125	7.4	9.2	9.7	6.6	0	1.8	6.4	23.	16.	10.	9.7	10.	14.	12.	13.
1	18.	9.4	8.8	9.5	11.	14.	50	110	128	133	128	7.6	9.4	7.9	6.8	1.8	0	6.5	22.	13.	9.7	8	9.7	13.	11.	11.
1	23.	15.	13.	14.	16.	18.	49.	105	123	128	123	14.	14.	12.	13.	6.4	6.5	0	29.	19.	16.	14.	18.	20	17.	18.
1	13.	14.	15.	15.	16.	14.	62	132	150	152	150	22.	19	16.	23.	23.	22.	29.	0	15.	13.	16.	15.	10.	12.	12.
1	5.3	3.8	4.6	3.9	3.3	2.7	49.	122	143	137	143	9.3	7.9	5.8	9.8	16.	13.	19.	15.	0	13.	16.	15.	10.	12.	12.
2	9.3	5	5.6	1.9	8.3	5.3	53.	118	136	145	136	9.4	10	5.7	10.	10.	9.7	16.	13.	13.	0	4.8	7.3	3.7	4.4	4
2	13.	6.9	6.8	7.8	9.8	9.1	54.	116	134	146	134	10.	11.	6.9	11.	9.7	8	14.	16.	16.	4.8	0	11.	8.3	6.6	6.7
2	9.4	3	3.5	1.6	1.8	5.9	44.	116	143	137	143	4.1	1.8	3.5	4.7	10.	9.7	18.	15.	15.	7.3	11.	0	7	3.3	5.2
2	5.9	4.1	5.2	5.7	6.4	3.3	51.	122	143	142	148	13.	8.8	6.1	11.	14.	13.	20	10.	10.	3.7	8.3	7	0	2.6	2.2
2	8.5	1.9	2.6	1.9	3.7	3.1	48	120	137	139	137	7.4	5	3.5	7.9	12.	11.	17.	12.	12.	4.4	6.6	3.3	2.6	0	0.8
2	8.1	2.3	3.3	3.9	4.6	2.7	50	121	138	140	138	9.4	7	4.2	9.9	13.	11.	18.	12.	12.	4	6.7	5.2	2.2	0.8	0

The basic concept involves constructing a matrix to depict the distance between points i and j, where Aij signifies the distance between the two cities. When i equals to j, Aii is naturally zero since the distance from point i to itself is inherently zero. The input for Table 2 is implemented using an array in code, denoted as [0, 8.9, 9.7, ..., 8.5, 8.1], accommodating up to 26 individuals.

4.1. Population

A population is an ensemble of individuals or chromosomes with a population size (number of individuals) that is constant for each generation [15]. First, initialize the variables by determining the class to represent the population. We decided to use a class-based implementation to attach pieces of information about generations of a particular population to objects of that class. Specifically, we can have things like a bag to represent the entire population, parents to represent the selected ones, selected superior few, score to store the scores of the best chromosomes in the population, best to store the best chromosomes themselves, and adjacency mat, an adjacency matrix that will we make used to calculate distances in the context of TSP.

4.2. Crossover

Crossover or the cross-breeding process aims to produce new values by combining genetic information from two different individuals [16]. In crossover, we take two parents. Then, we cut out part of one parent's chromosomes and fill in the rest with the other parent's chromosomes. When filling in the rest, we need to ensure that there are no duplicates in the chromosomes. Suppose one parent has [a, b, c, d, e] and the other has [b, a, e, c, d]. Suppose a cut of a random portion from the first parent is made and get [None, b, c, None, None]. Then, we fill the remaining empty indexes with the other unplaced parent genes in an orderly. In this case, we will get [a, b, c, e, d]. The vice versa thing happens to other offspring. The illustration of crossover is shown in Fig. 2.

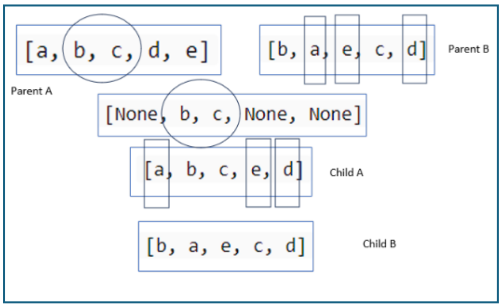

Figure 2. Crossover

4.3. Mutation

Mutation is the process of changing genetic material that occurs [17]. The mutation used is swap, which has child solutions represented as a sequence of points, for example [A, B, C, D, E], where each letter represents a point. This swap mutation operation randomly selects two positions in the sequence and swaps the points at those positions. Fig. 3 illustrates the mutation.

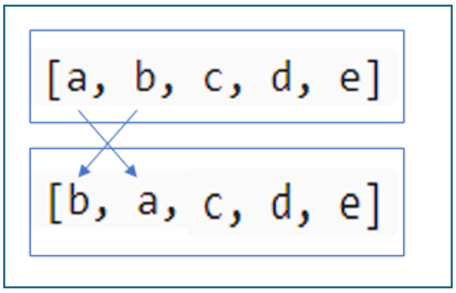

Figure 3. Mutations

Original Child Solution: [A, B, C, D, E]
Mutations occur at positions one and two:
Child Solution After Swap: [B, A, C, D, E]

In this way, the algorithm introduces some randomness in the solution to explore the search space more effectively. The image will essentially be a visual representation of the swap operation between two positions in the sequence.

Fig. 4 shows the iterations from generation 0 to 99, with the most optimal distance in each generation. It also provides a sequence of the best distance solutions.

```
Generation 0: 813.0000000000002 : None
Generation 1: 800.6000000000003 : [ 7  5 25  1 13 14  4 15 23  0 16 17 12 20  3 21  2  8 22 19 24 18  6 11
  9 10]
Generation 2: 470.0 : [7, 5, 25, 1, 13, 14, 4, 15, 23, 0, 16, 2, 12, 20, 3, 21, 17, 8, 22, 19, 24, 18, 6, 11, 9, 10]
Generation 3: 455.2000000000005 : [7, 5, 25, 1, 13, 14, 4, 15, 23, 0, 16, 17, 12, 20, 3, 21, 2, 11, 22, 19, 24, 18, 6, 8, 9, 10]
Generation 4: 435.3000000000007 : [7, 5, 25, 1, 13, 14, 19, 15, 23, 0, 16, 17, 12, 20, 3, 21, 2, 11, 22, 4, 24, 18, 6, 8, 9, 10]
Generation 5: 435.3000000000007 : [7, 5, 25, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 20, 3, 21, 2, 11, 22, 4, 24, 18, 6, 8, 9, 10]
Generation 6: 432.1 : [7, 5, 25, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 20, 3, 21, 2, 11, 22, 4, 24, 18, 6, 8, 9, 10]
Generation 7: 431.6500000000001 : [7, 5, 25, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 18, 6, 8, 9, 10]
Generation 8: 431.6500000000001 : [7, 5, 18, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 25, 6, 8, 9, 10]
Generation 9: 409.25 : [7, 5, 18, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 25, 6, 8, 9, 10]
Generation 10: 409.25 : [7, 5, 18, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 25, 6, 9, 8, 10]
Generation 11: 409.25 : [7, 5, 18, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 25, 6, 9, 8, 10]
Generation 12: 408.5 : [7, 5, 18, 1, 13, 14, 15, 23, 0, 19, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 25, 6, 9, 8, 10]
Generation 13: 407.0 : [7, 5, 18, 25, 13, 14, 15, 23, 0, 19, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 1, 6, 9, 8, 10]
Generation 14: 405.5000000000001 : [7, 5, 18, 25, 13, 14, 15, 23, 0, 19, 1, 16, 17, 12, 2, 3, 21, 20, 11, 22, 4, 24, 6, 9, 8, 10]
Generation 15: 403.9000000000001 : [7, 5, 18, 25, 13, 14, 15, 23, 0, 19, 1, 16, 17, 2, 3, 21, 20, 11, 22, 4, 12, 24, 6, 9, 8, 10]
```

Figure 4. Generation and Solution Iteration

After 100 iterations, the algorithm has converged to a minimum value of around 356.0. It turns out the best way to visit schools is in the order [7, 17, 21, 20, 3, 23, 18, 25, 0, 19, 11, 16, 15, 13, 5, 24, 2, 1, 22, 12, 4, 14, 6, 9, 8, 10] with a distance of 356 km. However, because the initial delivery started from PT. The Erlangga publisher is represented by zero, then the order of visiting schools is in the order [0, 19, 11, 16, 15, 13, 5, 24, 2, 1, 22, 12, 4, 14, 6, 9, 8, 10, 7, 17, 21, 20, 3, 23, 18, 25] with the same distance of 356 km. Fig. 5 shows the optimization with the axis of the total distance (km) and the number of iterations. The transition from the sequence [7, 17, 21, 20, 3, 23, 18, 25, 0, 19, 11, 16, 15, 13, 5, 24, 2, 1, 22, 12, 4, 14, 6, 9, 8, 10] to [0, 19, 11, 16, 15, 13, 5, 24, 2, 1, 22, 12, 4, 14, 6, 9, 8, 10, 7, 17, 21, 20, 3, 23, 18, 25] occurs because the delivery route must start and end at a specific point—in this case, the Erlangga Group's warehouse, represented by node zero. The Genetic Algorithm (GA), after 100 iterations, identified the initial sequence as the most efficient route to minimize the total distance traveled, calculated to be 356 km. However, this sequence did not account for the practical requirement that the delivery route must begin and end at the warehouse.

To meet this requirement, the sequence needed to be reordered. The solution involves shifting the sequence so that node zero becomes both the starting and ending point of the route. The portion of the sequence before node zero, specifically [7, 17, 21, 20, 3, 23, 18, 25], is moved to the end of the list, effectively creating a new sequence that starts with node zero and continues through the remaining nodes [19, 11, 16, 15, 13, 5, 24, 2, 1, 22, 12, 4, 14, 6, 9, 8, 10]. The result is a route that starts at the warehouse, visits each school in the optimal order identified by the GA, and returns to the warehouse, maintaining a total distance of 356 km.

The distance remains unchanged because the relative order of the schools in the sequence is preserved, ensuring that the optimal path identified by the GA is still followed. The only modification is the reordering to start and end at the warehouse, which does not affect the total distance but fulfills the practical constraint of the problem. This reordering process illustrates how the GA solution is adapted to meet real-world operational requirements while preserving the optimization of the route.

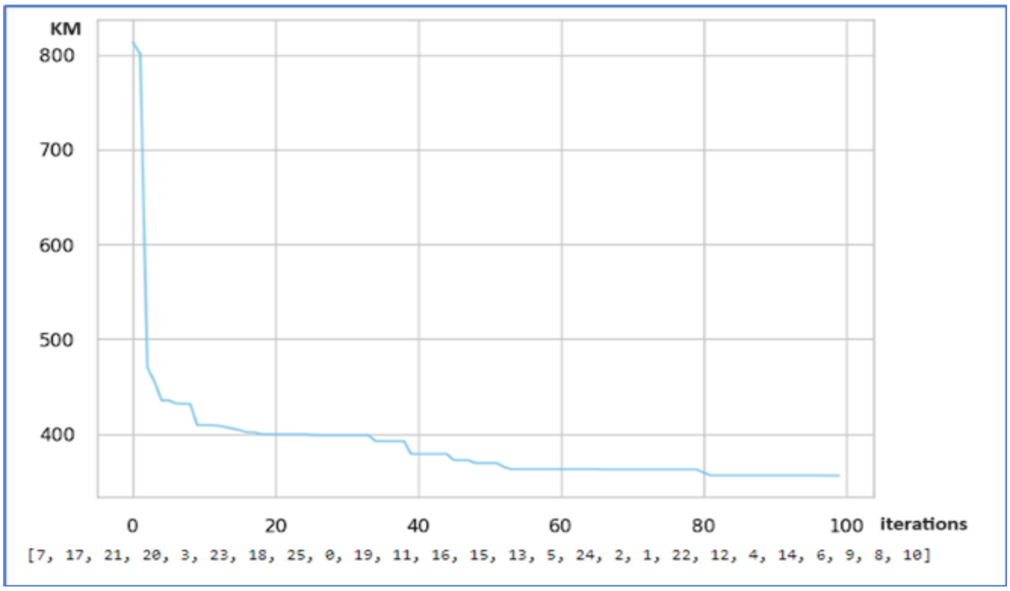

Figure 5. Representation of Route Optimization

5. Conclusion

Based on the research results regarding determining the shortest route for book distribution at the Yogyakarta branch of the Erlangga Publisher using a Genetic Algorithm, it can be concluded that a courier successfully distributed books with a route of 356 kilometers. The results of the shortest route for book distribution at the Yogyakarta branch of Erlangga Publishers were obtained by implementing the steps described above. The shortest route for distribution of this book starts from the Erlangga Publishing Office – Erlangga Publisher – TB SG – SMP Maarif Gamping – MTSN 04 Sleman – MAN 03 Sleman – SMA MUH 2 Yogyakarta – TB Togamas Suroto – SD Bluyahrejo – SD Jetis Harjo – TB Social Agency – SMP MUH 2 Gamping – SD Tegalrejo 02 – MTS Pamulangan – SD Muh Purworejo – SDN 01 Jalatunda – SDN Mandiraja Wetan – MI Kec Mandiraja – MI Muh Semampir – MAN 5 Sleman – TB Social Agency Jakal – TB Togamas Afandi – SD Vidiasana Qasana – TB Social Agency Ambarokmo – MTSN 8 Sleman – TB Social Agency Sagan - Erlangga Publisher. The distance of the route covered is 356 km. In the future, an interface needs to be created so that it can be used easily.

References

[1] A. Utamima, K. R. Pradina, N. S. Dini, and H. Studiawan, “Distribution Route Optimization of Gallon Water Using Genetic Algorithm and Tabu Search,” *Procedia Computer Science*, vol. 72, pp. 503–510, 2015, doi: 10.1016/j.procs.2015.12.132.

[2] T. Kolo, A. Fanggidae, and Y. N. Nabuasa, “Analisis Metode Cycle Crossover (Cx) Dan Metode Partial- Mapped Crossover (Pmx) Pada Penyelesaian Kasus Traveling Salesman Problem (TSP),” *Jurnal Komputer dan Informatika*, vol. 7, no. 1, 2019.

[3] S. R. Andani, “Optimasi Rute Menggunakan Vehicle Routing Problem (VRP) Dengan Algoritma Genetika,” *Jurnal Penerapan Sistem Informasi (Komputer dan Manajemen)*, vol. 4, no. 1, 2023.

[4] F. Ramadhani, F. A. Fathurrachman, R. Fitriawanti, A. C. Rongre, and V. N. Wijayaningrum, “Optimasi Pendistribusian Barang Farmasi Menggunakan Algoritma Genetika,” *Kumpulan Jurnal Ilmu Komputer*, vol. 5, no. 2, p. 159, Sep. 2018, doi: 10.20527/klik.v5i2.151.

[5] V. N. Wijayaningrum and W. F. Mahmudy, “Optimization of Ship’s Route Scheduling Using Genetic Algorithm,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 2, no. 1, p. 180, Apr. 2016, doi: 10.11591/ijeecs.v2.i1.pp180-186.

[6] I. Ihsani, A. Pramuntadi, D. H. Gutama, and D. P. Wijaya, “Implementasi Algoritma Genetika Dalam Penentuan Rute Optimal Untuk Kurir Kantor Pos Berbasis Web (Studi Kasus: Kantor Pos Wates),” *Indonesian Journal of Business Intelligence*, vol. 5, no. 2, p. 76, Dec. 2022, doi: 10.21927/ijubi.v5i2.2662.

[7] H. Z. Zahro' and F. S. Wahyuni, "Optimasi Rute Pengantaran Paket Menggunakan Metode Genetic Algorithm (GA)," *Inovatif : Jurnal Teknik Industri*, vol. 10, no. 1, pp. 41–44, Jun. 2020, doi: 10.36040/industri.v10i1.2527.
[8] Y. A. Hidayat, M. A. Arendra, and Y. D. Rosita, "Optimasi Rute Pengiriman Buah Kelapa Di Pasar Tradisional Kabupaten Mojokerto Menggunakan Algoritma Genetika," *Jurnal Informatika Teknologi dan Sains*, vol. 5, no. 2, pp. 289–293, May 2023, doi: 10.51401/jinteks.v5i2.2494.
[9] E. Sanggala, T. T. Dimyati, and Y. Yogaswara, "Genetic Algorithm Untuk Memperbaiki Rute Travelling Salesman Problem Yang Dihasilkan Dari Nearest Neighbour," *Jurnal Logistik Bisnis*, vol. 11, no. 2, pp 14-18, 2021.
[10] D. Prasisti and Y. A. Nugroho, "Optimasi Penjadwalan Produksi untuk Meminimalkan Makespan dengan Pendekatan Particle Swarm Optimization dan Genetic Algorithm," *Jurnal Teknologi dan Manajemen Industri Terapan*, vol. 2, no. 2, pp. 111–118, May 2023, doi: 10.55826/tmit.v2i2.134.
[11] A. Irianti, S. Cokrowibowo, and Aswandi, "Optimasi Multiple Traveling Salesman Problem dengan Algoritma Genetika pada Kasus Model Rute Terpendek Penjemputan Sampah di Kabupaten Majene", *Proceeding KONIK (Konferensi Nasional Ilmu Komputer)*, vol. 5, pp. 86–89, Aug. 2021.
[12] F. Puspitasari and P. Moengin, "Penerapan Metode Hybrid Genetic Algorithm (GA) dan Pattern Search (PS) untuk Penjadwalan Mata Kuliah Universitas," *Jurnal Rekayasa Sistem Industri*, vol. 9, no. 3, pp. 201–212, Oct. 2020, doi: 10.26593/jrsi.v9i3.4093.201-212.
[13] I. Permadi and Subanar, "Penerapan Algoritma Genetika untuk Optimasi Penjadwalan Tebangan Hutan", *JUITA: Jurnal Informatika*, vol. 1, no. 1, 2010.
[14] J. Massalesse, "Penerapan Algoritma Genetika Pada Penentuan Lintasan Terpendek Jalur Bus Rapid Transit Makassar," *Jurnal Matematika, Statistika dan Komputasi*, vol. 16, no. 2, p. 114, Dec. 2019, doi: 10.20956/jmsk.v16i2.7016.
[15] E. T. Setyoadi, "Penerapan Algoritma Genetik Untuk Optimasi Pola Penyusunan Barang Dalam Ruang Tiga Dimensi," *Teknika*, vol. 2, no. 1, pp. 10–19, Jul. 2013, doi: 10.34148/teknika.v2i1.9.
[16] S. F. Pane, R. M. Awangga, E. V. Rahcmadani, and S. Permana, "Implementasi Algoritma Genetika Untuk Optimalisasi Pelayanan Kependudukan," *Jurnal Tekno Insentif*, vol. 13, no. 2, pp. 36–43, Oct. 2019, doi: 10.36787/jti.v13i2.130.
[17] F. Mone and J. E. Simarmata, "Aplikasi Algoritma Genetika Dalam Penjadwalan Mata Kuliah," *Barekeng: Jurnal Ilmu Matematika & Terapan*, vol. 15, no. 4, pp. 615–628, Dec. 2021, doi: 10.30598/barekengvol15iss4pp615-628.