

# An Improved Genetic Algorithm for Vehicle Routing Problem Pick-up and Delivery with Time Windows

M. F. Ibrahim <sup>a\*</sup>, M. M. Putri <sup>a</sup>, D. Farista <sup>a</sup>, D. M. Utama <sup>b</sup>

<sup>a</sup> Logistics Engineering Department, Universitas Internasional Semen Indonesia, Indonesia  
Kompleks PT. Semen Indonesia (Persero) Tbk Jl. Veteran Gresik, Indonesia

<sup>b</sup> Industrial Engineering Department, Universitas Muhammadiyah Malang, Indonesia  
Jl. Raya Tlogomas No. 246, Malang 65144, Indonesia

\* Corresponding author: [faisalibrahim.ie@gmail.com](mailto:faisalibrahim.ie@gmail.com)

## ARTICLE INFO

### Article history

Received July 16, 2020

Revised February 26, 2021

Accepted February 28, 2021

Available Online February 28, 2021

### Keywords

VRPPDTW

Routing

Pick-up and Delivery

Time Windows

Genetic Algorithm

## ABSTRACT

Vehicle Routing Problem (VRP) has many applications in real systems, especially in distribution and transportation. The optimal determination of vehicle routes impacts increasing economic interests. This research aims to find the optimal solution in Vehicle Routing Problem Pick-up and Delivery with Time Windows (VRPPDTW). Targets of this problem included reducing distance travel and penalties. Three penalties that were considered are a capacity penalty, opening time capacity, and closing time capacity. An improved genetic algorithm was developed and used to determine the vehicle route. There were one main depot and 42 customers. This research raised the problem of a shipping and logistics company. Analysis of the results showed that the proposed route obtained from improved genetic algorithms (GA) was better than the existing route and previous algorithm. Besides, this research was carried out to analyze the effect of the number of iterations on distance traveled, the number of penalties, and the fitness value. This algorithm could be applied in VRPPDTW and produces an optimal solution.



This is an open-access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



## 1. Introduction

Vehicle Routing Problems (VRP) is a problem that attempts to find routes to deliver goods from a central depot to various destinations. Many companies face these problems, which are crucial in distribution and logistics. The correct route can find cost-effective routes, which contributed to high customer satisfaction [1]. Moreover, transportation plays a vital role in the success of the distribution process [2]. In this case, it is necessary to determine the path of visits to a set of customers by several vehicles [3]. The vehicles start and end at one depot. The depot's capacity and transport capacity affect the route designed to serve all customer requests [4]. Many more complex VRP problems have been developed from the classic VRP combination problems [5] [6]. One of some problem is Vehicle Routing



<https://doi.org/10.22219/JTIUMM.Vol22.No1.1-17>



<http://ejournal.umm.ac.id/index.php/industri>



[ti.jurnal@umm.ac.id](mailto:ti.jurnal@umm.ac.id)

Please cite this article as: Ibrahim, et al. (2021). An Improved Genetic Algorithm for Vehicle Routing Problem Pick-up and Delivery with Time Windows. Jurnal Teknik Industri, 22(1), 1-17.  
doi:<https://doi.org/10.22219/JTIUMM.Vol22.No1.1-17>

Problem Pick-up and Delivery with Time Windows (VRPPDTW). VRPPDTW has the primary goal of determining vehicles and routes' allocation to minimize costs by considering pick-up and delivery, capacity, and time windows [7]. VRPPDTW is based on integrated vehicles that deliver and pick-up goods in the transport network affected by the time [8]. Pick-up and delivery systems refer to how the vehicle fulfills the destination node's demand. The time window refers to the time when the fleet reaches the customer's node [9]. There are two types of time windows are imposed on each vehicle delivering the goods to the customers within a specific time interval [10]. The vehicle may not arrive at the customer's node closing time and before node open time. The best route must be on time and suit the disciplined capacity and minimum distance and cost.

The first Genetic Algorithm (GA) was developed by Holland and his colleagues at the University of Michigan in the 1960s. The main objective of developing GA is to design algorithms that can explore natural adaptation mechanisms' phenomenon into computer systems. GA adheres to biological evolution to transfer chromosome populations to other chromosomes through genetic crossover and mutation. GA necessarily is searching algorithms based on the selection and the genetics mechanisms. The use of GA in solving engineering and industrial problems reduces the amount and time of calculations, simplifies the modeling of functions, and reduces modeling errors [11]. Genetic algorithm applications have been widely used to support decision-making in routing problems [12].

VRP has numerous applications in real life, especially in transportation and distribution. VRP improves the economic interests and appropriateness of logistics planning [13]. Mohammed, et al. [14], Ho, et al. [15], and de Oliveira da Costa, et al. [16] have investigated VRP that was solved by an improved GA. The algorithm succeeded in solving the problem quickly and found a shorter distance for some routes. Good routing and scheduling can reduce delivery distance and travel that improve efficiency and decrease delivery cost. Minimization of travel time and distance is essential to optimize costs. Several studies that utilized GA include Nazif and Lee [17], Saxena, et al. [18], and Visutarrom and Chiang [19] that solved capacitated vehicle routing problems. The proposed algorithm used an optimized crossover operator. Computational results showed that the proposed algorithm is competitive in terms of the quality of the solutions found. Liu, et al. [20] and Escobar-Falcón, et al. [21] have utilized the GA to solve fleet size and mix vehicle routing problems. Even, Marinakis and Marinaki [22] combined GA and particle swarm optimization. Their research shows that combined GA and particle swarm optimization can increase the algorithm's effectiveness in solving a problem VRP.

Cheng and Wang [23] proposed solving the Vehicle Routing Problem with Time Windows (VRPTW) by a decomposition technique and GA. The results explained that GA provides a better solution if combined with the decomposition technique. Ghoseiri and Ghannadpour [24] solved VRPTW by goal programming and GA. Two famous local heuristics, namely one-interchange shortest path heuristic and elitism, keep a few right individuals. Vidal, et al. [25] developed an efficient hybrid GA for problem VRPTW. The proposed algorithm outperforms all current state-of-the-art algorithms on classical benchmark instances. Furthermore, Ursani, et al. [26] introduced the localized GA. The result from this research is a proposed algorithm on VRPTW for can minimize the distance. The results showed that the new algorithm provides the superior performance of the previous algorithm. The genetic algorithm helps solve challenging and complex issues for more substantial problems. Although it is used to solve various kinds of VRP, GA should possibly solve the VRPPDTW. Almost all paper describes a GA is developed for the VRP which can be competitive with the other modern heuristic techniques in terms of

solution time and quality. In the previous research, the researcher solved VRPPDTW using Genetic Algorithms under some of the facts that have been explained previously and considered that GA has never been applied in the case of VRPPDTW. This paper improves the standard genetic algorithm for solving VRPPDTW.

This paper proposes an improved GA for solving VRPPDTW problems based. Genetic algorithms through previous research have indeed been proven to solve various VRP problems. VRPPDTW has a higher complexity level, and the GA must be improved to solve the VRPPDTW problem effectively. The goal is to find the solution that most closely approaches the optimal solution with several targets: (1) to reduce the total distance and (2) to reduce the penalty. The improved GA for solving VRPPDTW was adapted from the original genetic algorithm developed by Holland [27]. Nearest neighbors heuristic algorithms are used in the initial population formation to create a better parent than just random generation. Calculation of fitness value (value of how good the route) based on distance traveled and several penalties related to VRPPDTW, including time windows open/close penalties and capacity penalties.

This paper's organization is presented as follows: The notations, assumptions, and descriptions of VRPPDTW problems, improved genetic algorithm, and case studies are presented in section 2. Comparison of the improved GA and existing route and comparison of the improved GA with the standard GA are presented in the results and discussion section 3. Section 4 presents conclusions and further work.

## 2. Methods

### 2.1 The Notations, Assumptions, and Descriptions of VRPPDTW Problems

The notations used in this problem are presented as follows:

$G_m$	: vehicle capacity
$d_{ij}$	: the distance between $i$ customer to $j$ customer
$d_i$	: total demand on $i$ customer
$p_i$	: quantity of pick-ups to $i$ customer
$K$	: set the number of vehicles
$k$	: a set of a vehicle used for distribution
$N$	: set the number of customers
$i, j$	: customer index
$n_k$	: the number of vehicles
$N$	: customer index (0 indicates the depot)
$C_i$	: customer $i$ , $C_0$ show depot
$V_k$	: vehicle route $k$
$q_{ik}$	: total demand for vehicle $k$ to $i$ customer
$x_{ijk}$	: has a value of 1, meaning that vehicle $k$ from $i$ customer going straight to $j$ customer, and if it is worth 0, then vice versa.
$R_k$	: total route time for vehicles $k$
$t_{ij}$	: travel time between customer $i$ and $j$ (proportional to distance)
$t_i$	: time of arrival at the $i$ customer
$f_i$	: $i$ customer service time
$w_i$	: waiting time before $i$ customer service
$e_i$	: earliest time to serve $i$ customer
$l_i$	: due date for $i$ customer service
$L_i$	: lateness of service in $i$ customer

$PO_i$  : opening penalty on  $i$  customer  
 $PC_i$  : closing penalty on  $i$  customer  
 $PCap_i$  : capacity penalty for the  $i$ th customer

VRPPDTW is an extension of Capacitated Vehicle Routing Problem. Every customer  $i$  (in this case, the branch office) is associated with demand and pick-up, representing the demand for homogeneous commodities to be sent and taken at each customer [28]. Besides, each branch office  $i$  is also associated with a time interval which is called as time windows.

The assumptions used in this problem are:

- (i) Vehicles must arrive and carry out loading and unloading at each customer  $i$  within the available time windows span, and the vehicle stop at the customer  $i$  during service time.
- (ii) If the vehicle arrives earlier than the customer time windows  $i$ , vehicles can wait until service can be started.
- (iii) Each route visits the depot vertex.
- (iv) Each customer vertex is visited precisely on one route.
- (v) Current vehicle load along the route must not be negative and exceed the capacity of the vehicle  $C$ .
- (vi) Demand is deterministic.
- (vii) The time window (open and close) for the customer is fixed.
- (viii) Vehicle speed is considered constant.

The mathematical model of the VRPPDTW problem is as follows:

Objective Function

$$Min = \sum_{i=1}^N (PO_i + PC_i + PCap_i) + \sum_{k=1}^K \sum_{i=0}^N \sum_{j=0}^N x_{ijk} \cdot d_{ij} \tag{1}$$

Constraint

$$\sum_{k \in K} \sum_{i \in V} x_{ijk} = 1, \forall j \in V \tag{2}$$

$$\sum_{i \in V} x_{ijk} - \sum_{i \in V} x_{jik} = 0, \forall j \in V, \forall k \in K \tag{3}$$

$$\sum_{i \in V} x_{ik} \leq 1, k \in K \tag{4}$$

$$\sum_{i \in V} q_{ijk} = \sum_{i \in V} \sum_{j \in V} x_{ijk} d_i, \forall k \in K \tag{5}$$

$$\sum_{k \in K} \sum_{i \in V} q_{ijk} - d_j = \sum_{k \in K} \sum_{i \in V} q_{ijk} - p_j, \forall j \in V \tag{6}$$

$$\sum_{i \in V} q_{ijk} = \sum_{i \in V} \sum_{j \in V} x_{ijk} p_j, \forall k \in K \quad (7)$$

$$q_{ijk} + p_{jk} - d_{jk} \leq G_m \quad (8)$$

$$\sum_{i=0}^N \sum_{i=0}^N x_{ik} (t_{ij} + f_i + w_i) \leq R_k, .k = 1 \dots K \quad (9)$$

$$t_j \geq t_i + w_i + f_i + t_{ij} - M(1 - x_{ijk}), ; i, j = 1, \dots, N; k = 1, \dots, K \quad (10)$$

$$e_i \leq t_i < l_i ; i = 1, \dots, N \quad (11)$$

$$t_i \geq 0 ; i = 1, N \quad (12)$$

$$w_i = \max\{0, (e_i - t_i)\} \quad (13)$$

$$PO_i = \begin{cases} 1, & w_i > 0 \\ 0, & \text{Otherwise} \end{cases} \quad (14)$$

$$L_i = \max\{0, (t_i - l_i)\} \quad (15)$$

$$PC_i = \begin{cases} 1, & L_i > 0 \\ 0, & \text{Otherwise} \end{cases} \quad (16)$$

$$PCap_i = \begin{cases} 0, & q_{ijk} + p_{jk} - d_{jk} \leq G_m \\ 1, & \text{Otherwise} \end{cases} \quad (17)$$

$$x_{ijk} = \begin{cases} 1, & \text{Vehicle access to } j \text{ after access } i \\ 0, & \text{Otherwise} \end{cases} \quad (18)$$

The optimization model in equation (1) indicates the minimization of the opening penalty, closing penalty, capacity penalty, and vehicle distance. Equation (2) is to ensure that each customer must be visited by 1 vehicle. Equation (3) is used to guarantee that the same vehicle comes and goes from every customer who visits it. Equation (4) is used to define that each vehicle only serves once. Equation (5) is a formula to ensure that the depot load is equal to all customers' demands to be served. Equation (6) is a formula that indicates the change in load after the vehicle serves customer i before going to customer j.

Meanwhile, equation (7) ensures that the vehicle's load is equal to the quantity of pick-up from all that is served by the vehicle k. Equation (8) ensures that the load carried by each vehicle does not exceed the maximum load capacity. Constraint (9) states that each vehicle serves all customers scheduled for that vehicle without exceeding its travel time. Constraint (10) is used to ensure that the two consumers' arrival times are compatible, where M is a huge real number. Constraint (11) ensures that the vehicle arrives at each consumer during the consumer's time window. Constraint (12) ensures that the arrival time of the vehicle to each customer is always positive. Equation (13) shows the waiting time for the vehicle to the customer i. Equation (14) PO<sub>i</sub> has a value of 1 when the vehicle experiences waiting time for the i-th customer. Constraint (15) shows the time of delay in service in the ith customer. Equation (16) has a value of 1 when the vehicle is delayed, and vice versa is 0. Equation (17) is 0 when the transported capacity does not exceed the delay load capacity. Equation (18) represents the characteristics and decision variables.

## 2.2. Improved GeneticAlgorithm

This section outlines the proposed improved GA. The GA is one of the most powerful metaheuristic methods in solving very complex problems. According to [Mohammed et al. \[14\]](#), GA is an adaptive heuristic search system based on natural and genetic selection's evolutionary ideas. The basic concept of GA is following the natural evolutionary system. Natural selection occurs, which makes the best individuals survive.

In this paper, the GA is utilized in solving the VRPPDTW problem. The main problem is delivering and picking up goods at each customer with different time windows. The use of genetic algorithms aims to explore the solution space. With GA, it is expected that this complex problem can be solved with near-optimal results.

GA was developed initially by [Holland \[27\]](#). So far, GA has been developed as a method of solving complex problems. GA is starting to be used to find suitable solutions to complex problems like in [Whitley, et al. \[29\]](#), [Sponsler \[30\]](#), and [Marin, et al. \[31\]](#), [Liu and Jiang \[32\]](#), and [Sethanan and Jamrus \[33\]](#), especially when the optimal solution fails to be found with other searches. GA can be a complex problem-solving solution that requires big-time in achieving optimal solutions. GA is efficient for solving problems with high complexity, such as VRP.

An improved genetic algorithm is proposed in solving the problem of determining the vehicle route in the VRPPDTW problem. The flow chart improved genetic algorithm used can be seen in [Fig. 1](#).

The implementation of an improved genetic algorithm in VRPTW problems can be seen in [Algorithm 1](#). The proposed GA improves stages such as establishing initial population, fitness calculations, selection, crossover, mutation, updating new population, repetition number of iterations, and determining the best solution. Several stages of improved GA are presented as follows:

### 2.2.1 Establishment of Initial Population

The initial population formed using two methods, namely random and heuristic. One heuristic method that is used is the Nearest Neighbor. Each point is represented by one number, and then it can be called a gene. A valid single route is represented by each chromosome that contains a collection of numbers or genes. The use of heuristic methods in genetic algorithms to form initial populations is expected to minimize the time to find the best solution.

### 2.2.2 Fitness

In VRPPDTW, Route optimization is influenced by the suitability of arrival time with the customer time windows and the vehicle load discipline with its capacity. This research adopted the VRPPDTW conceptual model and represented it in the form of a penalty. In this study, the fitness value considered the penalties that occur. The penalty is worth one every time a violation occurs and can be accumulated with other abuses. Thus, the three components must be considered in determining the fitness value. Fitness values were formulated in equation (19).

$$fitness = \frac{1}{1+total\ penalty} + \frac{1}{length\ of\ route} \quad (19)$$

### 2.2.3 selection

In this stage, the selection process was carried out to select genes in the chromosomes formed before. One of the most commonly used selection methods was the Roulette Wheel Selection (RWS). In this method, the parents obtained the chromosome's fitness value proportion. The chromosome with the best fitness value has elected a higher likelihood than other chromosomes. Chromosome parents elected at the next selection process entered the crossover process to produce offspring chromosomes.

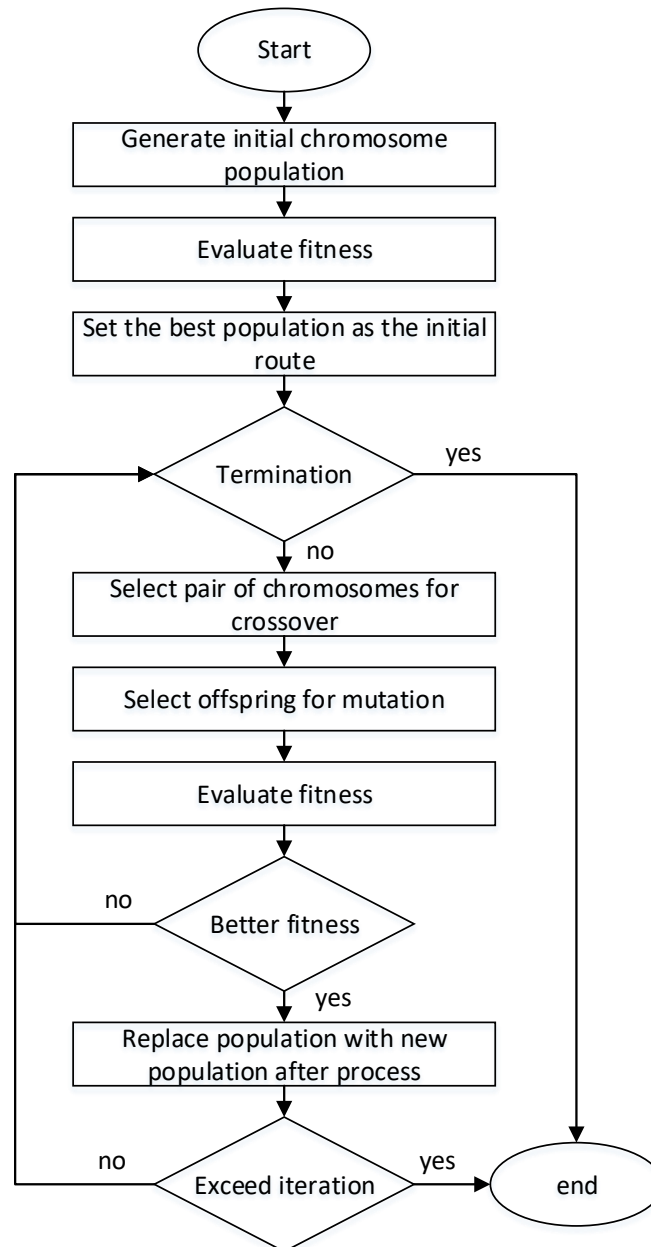


Fig. 1. The flow chart of the improved GA

#### *2.2.4 crossover*

In the crossover stage, not all chromosomes from parents were chosen for crossover. The crossover method used in this study was the heuristic crossover. The crossover step detail is explained in Algorithm 1. The distance between genes considered in the crossover stage to create a better new population.

#### *2.2.5 Mutation*

Like a crossover, whether or not a mutation process depends on the probability of mutation, in detail, the mutation process is explained in Algorithm 1. In this study's mutation process, the distance between genes was considered to create a better new population.

#### *2.2.6 Establishment of a New Population*

In this stage, a new population formed to replace the old population. Population replacement means that all chromosomes beginning in a generation were replaced by chromosome results obtained after the crossover and mutation stages. The population became a new population and was repeated before selecting crossover and mutation back at the new population.

#### *2.2.7 Repetition Number of Iterations*

At this stage, the selection process, evaluation of fitness, crossover, mutation, and new population formation were repeated under the number of iterations determined to achieve optimal results. That way, the chromosomes that survive were the chromosomes with the best fitness value, among other chromosomes. This process followed the principle of natural selection, "who is strong (has a better fitness value), it survives.

#### *2.2.8 Determining the Best Solution*

By comparing the fitness values obtained for each iteration, the best solution was the chromosome with the best fitness value that survives at the end of the iteration.

### **2.3. Case Study**

The case study was conducted in a shipping and logistics company serving shipping services throughout Surabaya, East Java, Indonesia. One main depot and 42 branch offices (customer) are scattered to help certain areas as a transit point for all packages before being sent to their destination. Each vehicle has a carrying capacity. Therefore, in the route's preparation, one must consider the load amount when loading and unloading. In this case, the branch office has various loading and unloading service time windows. The vehicle must wait if it arrives earlier than the branch office open time windows. If the vehicle arrives more than close time windows, it disrupts the branch office's package delivery schedule. Couriers at the branch depart for the last mile delivery at specified times. Packages brought by vehicles arriving at times exceeding close time windows cannot be carried by the courier at that time and sent on the next shipment. Packages that are delivered late reduce customer satisfaction and build up in the branch office warehouse, which is harmful. The case study data used are data on one day of delivery.

The company has four transport vehicles with a maximum capacity of 700 kilograms. The vehicles depart from the depot at eight in the morning to several branch offices. Each transport vehicle is assumed to have an average fuel consumption of eleven kilometers per liter of fuel. Data for each branch office related to the quantity of pick-up



and delivery and time windows can be seen in [Table 1](#). The distance matrix data, which then be converted into the travel time matrix, can be seen in [Appendix A](#).

Algorithm 1: Proposed Improve GA for CVRPPDTW Problem

1. start;
  2. population (a set of the route) initialization, use randomized method and nearest neighbor algorithm to form chromosomes (routes) structure;
  3. evaluate total length, total penalty, and fitness value of the initial population,  $fitness = (1/(1 + total\ penalty)) + (1/length\ of\ route)$ ;
  4. choose the population with the best fitness value as the initial route;
  5. loop the following steps for the specified number of iterations (the more iterations, the better):
    - 5.1. copy chromosome structure of the best population to new population (route);
    - 5.2. generate random numbers ( $R_c$ ) in the range 0.00 to 1.00, be compared to crossover probability ( $P_c$ );
    - 5.3. if  $R_c > P_c$  proceed to stage 5.4; if  $R_c \leq P_c$  do chromosome crossover; loop the following steps as much as the number of chromosomes:
      - 5.5.1. specify a random crossover point  $C_p$  by generating random numbers in range 1 to the value of the number of genes (node) in a chromosome;
      - 5.5.2. find the distance between the crossover point node from the previous node  $C_d = distance\ gen\ C_p - 1\ to\ gen\ C_p$
      - 5.5.3. determine the chromosomes that be crossover by generating random numbers in the range of 1 to the value of the number of chromosomes except for the previously chosen chromosome;
      - 5.5.4. Find the distance between the crossover point node to all nodes on the chromosome selected for a crossover;
      - 5.5.5. Crossover the  $C_p$  genes with genes that have the closest distance;
    - 5.4. generate random numbers ( $R_m$ ) in the range 0.00 to 1.00, compared to mutation probability ( $P_m$ );
    - 5.5. if  $R_m > P_m$  proceed to stage 5.6; if  $R_m \leq P_m$  do chromosome mutation; loop the following steps as much as the number of chromosomes:
      - 5.5.1. specify a random mutation point  $M_p$  by generating random numbers in range 1 to the value of the number of genes (node) in a chromosome;
      - 5.5.2. find the distance between the mutation point node from the previous node  $M_d = distance\ gen\ M_p - 1\ to\ gen\ M_p$
      - 5.5.3. Find the distance between the node  $M_p - 1$  to all other nodes on the same chromosome;
      - 5.5.4. Mutation of the  $M_p$  genes with genes that have the closest distance;
    - 5.6. evaluate the fitness value of the result population;
      - 5.6.1 If the fitness value is better than the initial population;
        - 5.6.2.1 Replace population with population after the crossover and mutation process
      - 5.6.2 If the fitness value is not better than the initial population;
        - 5.6.2.2 Do not replace the population / Keep using the initial population
  6. end of the loop;
  7. write out the surviving population (best route);
- End.



Table 1. The quantity of pick-up, the quantity of delivery, and time windows data

No.	Time Windows		Quantity		No.	Time Windows		Quantity	
	Open	Close	Pick-up	Delivery		Open	Close	Pick-up	Delivery
0	00:00	23:59	0	0	22	08:00	10:30	38.5	17.5
1	08:00	10:30	21	52.5	23	08:00	10:30	17.5	49
2	08:00	10:30	14	52.5	24	08:00	10:30	14	52.5
3	08:00	10:30	14	56	25	08:00	10:30	31.5	35
4	08:00	10:30	17.5	63	26	08:00	10:30	38.5	45.5
5	08:00	10:30	10.5	56	27	08:00	10:30	10.5	49
6	08:00	10:30	35	171.5	28	08:00	10:30	42	52.5
7	08:00	10:30	24.5	73.5	29	08:00	10:30	35	38.5
8	08:00	10:30	17.5	59.5	30	08:00	10:30	3.5	7
9	08:00	10:30	3.5	31.5	31	08:00	10:30	17.5	14
10	08:00	10:30	7	56	32	08:00	10:30	31.5	17.5
11	08:00	10:30	31.5	52.5	33	08:00	10:30	14	21
12	08:00	10:30	17.5	0	34	08:00	10:30	3.5	28
13	08:00	10:30	28	24.5	35	08:00	10:30	7	35
14	08:00	10:30	14	17.5	36	08:00	10:30	17.5	24.5
15	08:00	10:30	3.5	42	37	08:00	10:30	38.5	17.5
16	08:00	10:30	24.5	28	38	08:00	10:30	24.5	10.5
17	08:00	10:30	24.5	21	39	08:00	10:30	21	17.5
18	08:00	10:30	21	42	40	08:00	10:30	7	3.5
19	08:00	10:30	7	0	41	08:00	10:30	10.5	3.5
20	08:00	10:30	28	49	42	08:00	10:30	7	3.5
21	08:00	10:30	14	56					

This study's experiments used the GA parameters as follows: crossover probability: 0.8, mutation probability: 0.8, chromosome: 4 (according to the number of vehicles), and 10,000 iterations. This experiment was performed using Matlab 14 running on an Intel (R) Core (TM) i5-3210M CPU @ 2.50GHz 8Gb RAM. This parameter was also used as a parameter for the comparison of the previous GA algorithm. This study also compared the Total cost comparison Actual route. The total mileage of the problem was multiplied by the cost per kilometer. The cost per kilometer of this problem was IDR. 595.45.

### 3. Results and Discussion

#### 3.1 Comparison Improve GA and Existing Route

The primary purpose of VRPPDTW was to minimize total penalty and distance. In VRPPDTW, the valid route was when it has a minimum distance and if the capacity and time windows constraints can be met. The VRPPDTW problem was very close to the actual system conditions where time windows and capacity also affect a distribution route's success. The improved genetic algorithm was proven to produce the best route for pick-up and delivery and time windows. This paper compared the actual route and the route that results from the improved genetic algorithm. The analysis focused on each route's

advantages and disadvantages, especially the results of the developed algorithm. The total distance of each vehicle was obtained from the total distance between the nodes traversed. Each vehicle's total journey was added to get the total distance traveled by all vehicles on that day. A penalty value of 1 was given every time there is excess capacity. The vehicle arrived before the time windows, and the vehicle arrived after the windows.

Table 2. Distance comparison of the existing route and improved GA

Vehicle	Existing Route		Improved GA Route	
	Driving time (h:mm)	Distance Travelled (km)	Driving time (h:mm)	Distance Travelled (km)
1	1:53	94.1	1:32	76.90
2	1:57	97.1	1:49	90.90
3	2:13	111.1	1:52	93.40
4	2:16	113.6	1:22	69.00
Total	8:19	415.9	6:35	330.20

Table 2 describes distance comparison of the existing route and improved GA. From the total mileage value of all vehicles, it was revealed that the route from the improved GA results was in shorter mileage compared to the company's existing routes. These results supported the earlier research statement by Ibrahim, et al. [13] that genetic algorithms can produce effective solutions to reduce total mileage in the case of VRP. On the proposed route, it was found that all vehicles have less mileage than the existing route. The proposed route resulted in a 20.6% cut in the distance. It can be concluded that the route resulting from the improved genetic algorithm succeeded in minimizing total mileage.

This research also compared the total cost of existing routes and proposed routes obtained with an improved genetic algorithm. Total cost comparison actual route and improved GA (In IDR) could be seen in Table 3. Results found that the costs of the improved GA are more minimum, which is IDR. 196,619.09 compared to the company's existing routes with a total cost of IDR. 247,649.55. Therefore, the route formed by an improved GA was better than the company's existing route. The costs calculation did not consider costs incurred due to vehicles arriving before or after the destination's time windows. The total cost was also not considered if there was an excess capacity.

Table 3. Total cost comparison actual route and improved GA (in IDR.)

Vehicle	Existing Route	Improved Genetic Algorithm Route
1	56,032.27	45,790.45
2	57,818.64	54,126.82
3	66,155.00	55,615.45
4	67,643.64	41,086.36
Total	247,649.55	196,619.09

Table 4 shows the total penalty value comparison of the actual route and improved GA. The VRPPDTW problem in this paper used the type of soft time windows. Penalties were applied if the vehicle arrived outside the available time window (penalty time windows). Likewise, the capacity penalty was applied if vehicles carried more than their capacity. The more penalties were given on a route illustrate, the worse the route was. We

continued to analyze in-depth the penalties that occurred on the proposed and existing routes.

Table 4. Penalty value comparison of actual route and improved GA

Item	Existing Route	Improved GA Route
Penalty capacity	2	0
Penalty time windows (opening)	0	0
Penalty time windows (closing)	4	0
Total Penalty	6	0

The penalty capacity was worth two from the computational results, which means twice the vehicle's excess capacity events. Because the VRP problem in this paper considered pick-up and delivery, routes that were not optimal can cause the vehicle's overcapacity. For example, a vehicle departed at full capacity and visited branch office A. Goods delivered were significantly less than those picked up from branch office A. There was overcapacity in the vehicle. Therefore, the route formed must also consider the number of pick-ups and deliveries, not just the delivery order with the minimum distance. There have been four-time windows (closing) penalties, which means four vehicles were coming after the time windows limit. In this company's case, the vehicle must arrive before the time windows limit because the delivered goods were immediately taken by the courier to be delivered to the customer's home. Delay in the vehicle's arrival would cause the package to be delivered late to the customers' home because they had to follow the next delivery time. Besides, the buildup of packages at the branch office also causes various new problems. On the route produced by an improved genetic algorithm, there were no penalties or violations committed. It can be concluded that all vehicles always arrived at the branch office time windows and all vehicles were never overcapacities.

### 3.2 Comparison improve GA with the standard GA

This article also analyzed the effect of the number of iterations on the value of fitness. GA was closely related to the concept of natural selection. Thus, the best generation survives while the weak be defeated and discarded. In this case, the intended generation was a set of routes. The higher the fitness value, the better the route (minimizing mileage and penalty). Logically, the value of fitness increased with the increasing number of iterations. Thus, the more the number of iterations, the better the resulting route.

Based on Fig. 2, the experiment results showed that the initial fitness value was 0.11 when the number of iterations increased to 1 when the number of iterations was 7700. The fitness value continued to grow when the iteration number was increased in proportion to the reduction in the value of the penalty and traveled distance. However, it stopped at a certain number of iterations. Even though growing fitness was valued, even though the number of iterations continued added, it illustrated no better solution. As seen in the results, the number of penalties during the iteration of 7700 has a value of 0. It means that no restrictions are violated.

The cessation and insignificant increased fitness value were among the stopping criteria in the Improved GA, besides the specified number of iterations. Hence, if the research set the number of iterations to 20,000 iterations, but there has been no significant increase in fitness value when the number of iterations is 10,000, then the computation could be stopped. This study compared the performance of the Improved GA developed

with the standard GA. From Fig. 2, it can be seen that the improved GA has succeeded in getting the best solution with a smaller number of iterations which certainly affected the computation time.

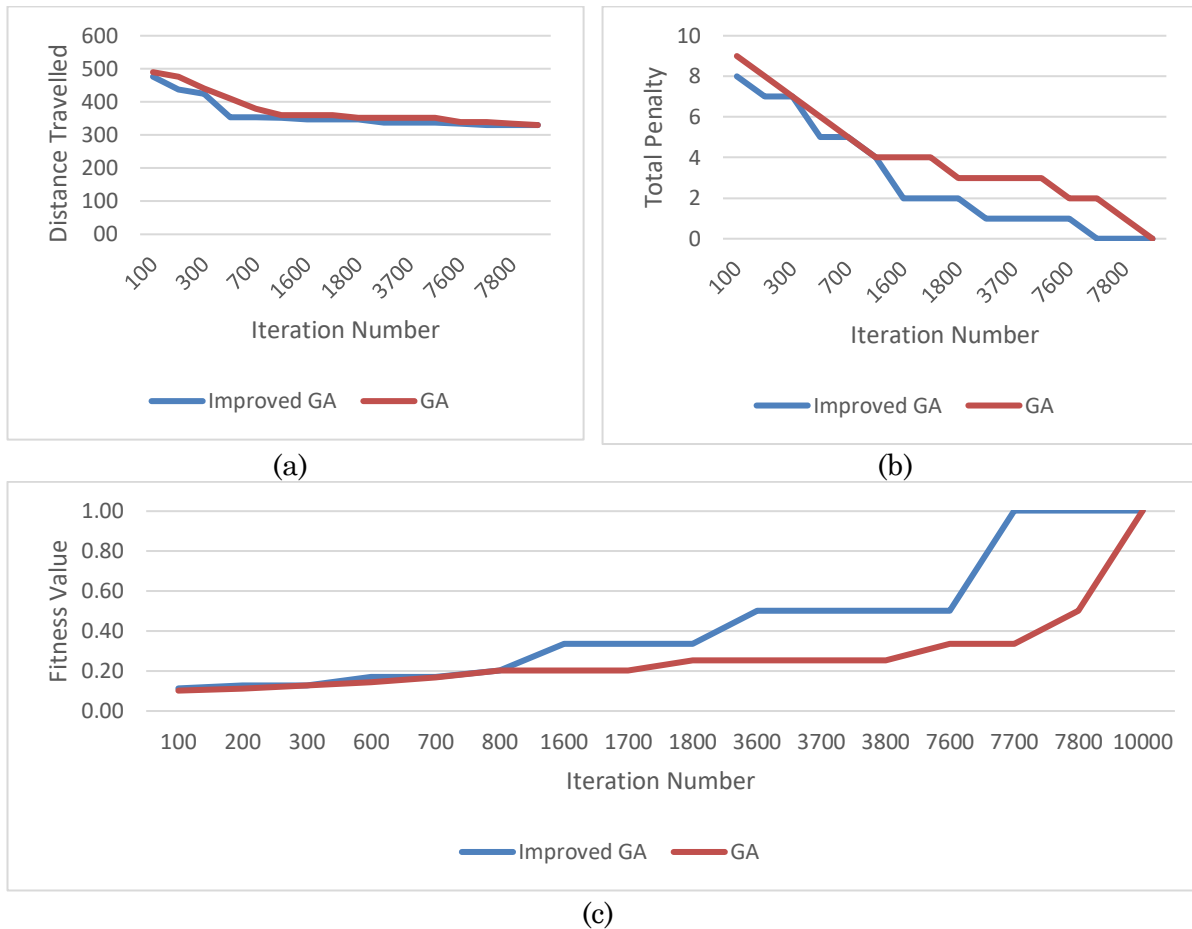


Fig. 2. (a) The effect of iterations toward the distance, (b) Effect of iterations toward Total Penalty, and (c) Effect of iterations toward Fitness value

#### 4. Conclusion

This study discussed the VRPPDTW model to optimize the route of shipping and logistics company's vehicle. An Improved GA was developed and applied to solve the VRPPDTW problem. From the case study, it can be concluded that the improved GA that developed can be applied in determining the route. The resulting route was better than the existing company route, both in costs and customer service satisfaction. Improved GA results compared with the company's existing routes and routes generated by standard GA. The analysis was carried out in various aspects, including vehicle distance, total penalty, and total transportation costs. The total distance traveled on the proposed route was shorter than the existing route, affecting transportation's total cost. There were six violations related to time windows and capacity on the actual route. In contrast, on the proposed route, there were no violations at all. The best solution resulted from an improved GA with a shorter computation time than standard GA. Therefore, it can be

concluded that the Improved GA produced a better route in all aspects and resolved the VRPPDTW problems faced with relatively short computing time. Future work exploring various crossover and mutation methods can be done to provide recommendations for the most suitable crossover and mutation methods.

### Acknowledgments

This research was supported and financed by the Ministry of Research, Technology, and Higher Education of the Republic of Indonesia under the PDP scheme.

### References

- [1] V. N. Coelho, A. Grasas, H. Ramalhinho, I. M. Coelho, M. J. F. Souza, and R. C. Cruz, "An ILS-based algorithm to solve a large-scale real heterogeneous fleet VRP with multi-trips and docking constraints," *European Journal of Operational Research*, vol. 250, pp. 367-376, 2016. <https://doi.org/10.1016/j.ejor.2015.09.047>.
- [2] D. M. Utama, D. S. Widodo, M. F. Ibrahim, and S. K. Dewi, "A New Hybrid Butterfly Optimization Algorithm for Green Vehicle Routing Problem," *Journal of Advanced Transportation*, vol. 2020, p. 8834502, 2020. <https://doi.org/10.1155/2020/8834502>.
- [3] D. M. Utama, S. K. Dewi, A. Wahid, and I. Santoso, "The vehicle routing problem for perishable goods: A systematic review," *Cogent Engineering*, vol. 7, p. 1816148, 2020. <https://doi.org/10.1080/23311916.2020.1816148>.
- [4] F. Arnold and K. Sörensen, "What makes a VRP solution good? The generation of problem-specific knowledge for heuristics," *Computers & Operations Research*, vol. 106, pp. 280-288, 2019. <https://doi.org/10.1016/j.cor.2018.02.007>.
- [5] P. Sitek, J. Wikarek, K. Rutczyńska-Wdowiak, G. Bocewicz, and Z. Banaszak, "Optimization of capacitated vehicle routing problem with alternative delivery, pick-up and time windows: A modified hybrid approach," *Neurocomputing*, vol. 423, pp. 670-678, 2021. <https://doi.org/10.1016/j.neucom.2020.02.126>.
- [6] S. K. Dewi and D. M. Utama, "A New Hybrid Whale Optimization Algorithm for Green Vehicle Routing Problem," *Systems Science & Control Engineering*, vol. 9, pp. 61-72, 2021. <https://doi.org/10.1080/21642583.2020.1863276>.
- [7] C. K. Y. Lin, "A cooperative strategy for a vehicle routing problem with pick-up and delivery time windows," *Computers & Industrial Engineering*, vol. 55, pp. 766-782, 2008. <https://doi.org/10.1016/j.cie.2008.03.001>.
- [8] M. Mahmoudi and X. Zhou, "Finding optimal solutions for vehicle routing problem with pick-up and delivery services with time windows: A dynamic programming approach based on state-space-time network representations," *Transportation Research Part B: Methodological*, vol. 89, pp. 19-42, 2016. <https://doi.org/10.1016/j.trb.2016.03.009>.
- [9] M. F. Ibrahim, F. R. Nurhakiki, D. M. Utama, and A. A. Rizaki, "Optimised Genetic Algorithm Crossover and Mutation Stage for Vehicle Routing Problem Pick-Up and Delivery with Time Windows," in *IOP Conference Series: Materials Science and Engineering*, 2021, p. 012025. <https://doi.org/10.1088/1757-899x/1071/1/012025>.
- [10] R. Elshaer and H. Awad, "A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants," *Computers & Industrial Engineering*, vol. 140, p. 106242, 2020. <https://doi.org/10.1016/j.cie.2019.106242>.

- [11] J. Protopopova and S. Kulik, "Educational Intelligent System Using Genetic Algorithm," *Procedia Computer Science*, vol. 169, pp. 168-172, 2020. <https://doi.org/10.1016/j.procs.2020.02.130>.
- [12] S. Karakatić, "Optimizing nonlinear charging times of electric vehicle routing with genetic algorithm," *Expert Systems with Applications*, vol. 164, p. 114039, 2021. <https://doi.org/10.1016/j.eswa.2020.114039>.
- [13] M. F. Ibrahim, I. Masudin, and T. E. Saputro, "A hybrid genetic algorithm implementation for vehicle routing problem with time windows," *Jurnal Ilmiah Teknik Industri*, vol. 14, pp. 196-204, 2016. <https://doi.org/10.23917/jiti.v14i2.985>.
- [14] M. A. Mohammed, M. K. Abd Ghani, R. I. Hamed, S. A. Mostafa, M. S. Ahmad, and D. A. Ibrahim, "Solving vehicle routing problem by using improved genetic algorithm for optimal solution," *Journal of Computational Science*, vol. 21, pp. 255-262, 2017. <https://doi.org/10.1016/j.jocs.2017.04.003>.
- [15] W. Ho, G. T. S. Ho, P. Ji, and H. C. W. Lau, "A hybrid genetic algorithm for the multi-depot vehicle routing problem," *Engineering Applications of Artificial Intelligence*, vol. 21, pp. 548-557, 2008. <https://doi.org/10.1016/j.engappai.2007.06.001>.
- [16] P. R. de Oliveira da Costa, S. Mauceri, P. Carroll, and F. Pallonetto, "A Genetic Algorithm for a Green Vehicle Routing Problem," *Electronic Notes in Discrete Mathematics*, vol. 64, pp. 65-74, 2018. <https://doi.org/10.1016/j.endm.2018.01.008>.
- [17] H. Nazif and L. S. Lee, "Optimised crossover genetic algorithm for capacitated vehicle routing problem," *Applied Mathematical Modelling*, vol. 36, pp. 2110-2117, 2012. <https://doi.org/10.1016/j.apm.2011.08.010>.
- [18] R. Saxena, M. Jain, K. Malhotra, and K. D. Vasa, "An optimized openmp-based genetic algorithm solution to vehicle routing problem," presented at the Smart Computing Paradigms: New Progresses and Challenges, vol 767, pp 237-245. 2020. [https://doi.org/10.1007/978-981-13-9680-9\\_20](https://doi.org/10.1007/978-981-13-9680-9_20).
- [19] T. Visutarrom and T. Chiang, "An Evolutionary Algorithm with Heuristic Longest Cycle Crossover for Solving the Capacitated Vehicle Routing Problem," in *2019 IEEE Congress on Evolutionary Computation (CEC)*, 2019, pp. 673-680. <https://doi.org/10.1109/CEC.2019.8789946>.
- [20] S. Liu, W. Huang, and H. Ma, "An effective genetic algorithm for the fleet size and mix vehicle routing problems," *Transportation Research Part E: Logistics and Transportation Review*, vol. 45, pp. 434-445, 2009. <https://doi.org/10.1016/j.tre.2008.10.003>.
- [21] L. Escobar-Falcón, D. Álvarez-Martínez, J. Wilmer-Escobar, and M. Granada-Echeverri, "A specialized genetic algorithm for the fuel consumption heterogeneous fleet vehicle routing problem with bidimensional packing constraints," *International Journal of Industrial Engineering Computations*, vol. 12, pp. 191-204, 2021. <https://doi.org/10.5267/j.ijiec.2020.11.003>.
- [22] Y. Marinakis and M. Marinaki, "A hybrid genetic – Particle Swarm Optimization Algorithm for the vehicle routing problem," *Expert Systems with Applications*, vol. 37, pp. 1446-1455, 2010. <https://doi.org/10.1016/j.eswa.2009.06.085>.
- [23] C.-B. Cheng and K.-P. Wang, "Solving a vehicle routing problem with time windows by a decomposition technique and a genetic algorithm," *Expert Systems with Applications*, vol. 36, pp. 7758-7763, 2009. <https://doi.org/10.1016/j.eswa.2008.09.001>.



- [24] K. Ghoseiri and S. F. Ghannadpour, "Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm," *Applied Soft Computing*, vol. 10, pp. 1096-1107, 2010. <https://doi.org/10.1016/j.asoc.2010.04.001>.
- [25] T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins, "A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows," *Computers & Operations Research*, vol. 40, pp. 475-489, 2013. <https://doi.org/10.1016/j.cor.2012.07.018>.
- [26] Z. Ursani, D. Essam, D. Cornforth, and R. Stocker, "Localized genetic algorithm for vehicle routing problem with time windows," *Applied Soft Computing*, vol. 11, pp. 5375-5390, 2011. <https://doi.org/10.1016/j.asoc.2011.05.021>.
- [27] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*: MIT press, 1992.
- [28] P. Toth and D. Vigo, "The vehicle routing problem," ed: SIAM, 2002.
- [29] D. Whitley, T. Starkweather, and C. Bogart, "Genetic algorithms and neural networks: optimizing connections and connectivity," *Parallel Computing*, vol. 14, pp. 347-361, 1990. [https://doi.org/10.1016/0167-8191\(90\)90086-O](https://doi.org/10.1016/0167-8191(90)90086-O).
- [30] J. L. Sponser, "Genetic algorithms applied to the scheduling of the hubble space telescope," *Telematics and Informatics*, vol. 6, pp. 181-190, 1989. [https://doi.org/10.1016/S0736-5853\(89\)80015-2](https://doi.org/10.1016/S0736-5853(89)80015-2).
- [31] F. J. Marin, F. J. Gonzalez, and F. Sandoval, "The Routing Problem In Traffic Control Using Genetic Algorithms," in *Intelligent Components and Instruments for Control Applications 1994*, C. Bányász, Ed., ed Oxford: Pergamon, 1994, pp. 187-191. <https://doi.org/10.1016/B978-0-08-042234-3.50032-1>.
- [32] R. Liu and Z. Jiang, "A hybrid large-neighborhood search algorithm for the cumulative capacitated vehicle routing problem with time-window constraints," *Applied Soft Computing*, vol. 80, pp. 18-30, 2019. <https://doi.org/10.1016/j.asoc.2019.03.008>.
- [33] K. Sethanan and T. Jamrus, "Hybrid differential evolution algorithm and genetic operator for multi-trip vehicle routing problem with backhauls and heterogeneous fleet in the beverage logistics industry," *Computers & Industrial Engineering*, vol. 146, p. 106571, 2020. <https://doi.org/10.1016/j.cie.2020.106571>.



