

PAPER • OPEN ACCESS

Optimised Genetic Algorithm Crossover and Mutation Stage for Vehicle Routing Problem Pick-Up and Delivery with Time Windows

To cite this article: M F Ibrahim *et al* 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* **1071** 012025

View the [article online](#) for updates and enhancements.



EEG/ECOG AMPLIFIERS
& ELECTRODES
ELECTRICAL/CORTICAL
STIMULATORS
REAL-TIME PROCESSING

g.tec
gtec.at/shop
SHOP NOW

Optimised Genetic Algorithm Crossover and Mutation Stage for Vehicle Routing Problem Pick-Up and Delivery with Time Windows

M F Ibrahim^{1*}, F R Nurhakiki¹, D M Utama² and A A Rizaki¹

¹Department of Logistics Engineering, Universitas Internasional Semen Indonesia, Gresik, Indonesia.

²Department of Industrial Engineering, University of Muhammadiyah Malang, Malang, Indonesia.

E-mail: faisalibrahim.ie@gmail.com

Abstract. Some problems happen in transportation and distribution. Vehicle Routing Problem (VRP) can be applied in some systems above. Deciding the optimal route for every vehicle will impact increasing economic interests and expected logistical planning results. This research will raise the problem of a shipping and logistics company. There are 45 branch offices with one main depot to serve a certain area that will be a transit point before all packages will be sent to the destination. The vehicle will depart from the depot to all branch offices to deliver and pick the package up at certain hours. In every route, planning should be considered to the amount of load when loading and unloading. Every vehicle has carrying capacity, and every branch office has various loading and unloading service time windows. Based on the problem's description, this research was conducted to find the optimal solution in the Vehicle Routing Problem Pick-up and Delivery with Time Windows (VRPPDTW). An optimised genetic algorithm was developed to solve these problems by adjusting the crossover and mutation stages. The result informs that the route proposed from optimised genetic algorithms is better than the company's existing route in all aspects. On the other hand, we carried out an analysis effect of the number of iterations on distance traveled, the number of penalties, and the fitness value. This algorithm can be applied in VRPPDTW and produces an optimal solution.

1. Introduction

The vehicle routing problem (VRP) attempts to find a route to deliver goods from a central depot to a geographically dispersed destination set. Many companies face this problem, which is very important in distribution and logistics because of the need to find a cost-effective route that provides high customer satisfaction [1]. During the product distribution process, it is often a problem if a product's delivery takes longer to exceed the specified delivery working hours. It is influenced by several factors, including travel time at each destination and vehicle capacity. So the solution that needs to be done and considered is calculating the route at each destination starting from the Distribution Warehouse. This route calculation states that the vehicle will visit the customer exactly once according to the shortest route's order until the vehicle returns to the depot again. The issues raised in this article will lead to Vehicle Routing Problem Pick-up and Delivery with Time Windows (VRPPDTW).

VRPPDTW has the primary goal of determining the needs of resources and routes with minimum costs, but in pick-up and delivery, consider the capacity and time windows [1]. VRPPDTW

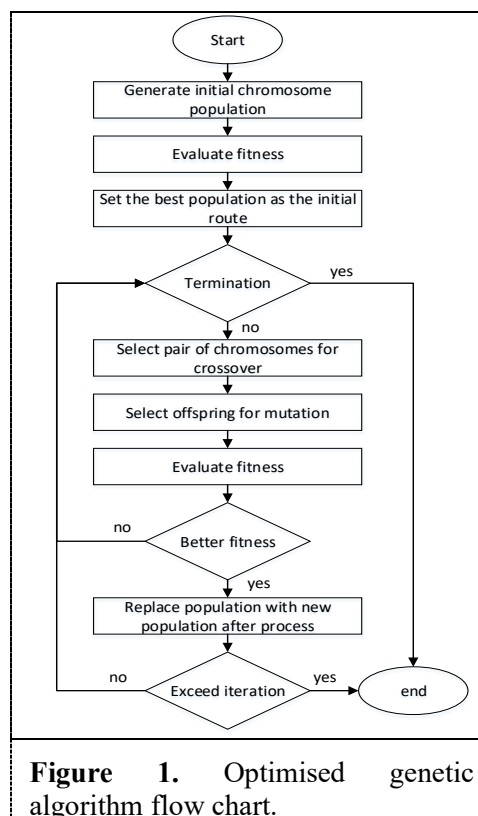


based on the integration of vehicles that deliver and pick-up goods in the transport network that is affected by time, thus enabling the optimization of the combined assignment of goods into vehicles and routing of congested transportation networks [2]. Pick-up and delivery systems refer to how the fleet fulfills the destination node's request, and the time window refers to the time when the fleet reaches the customer's node. There are two types of time windows, and it imposes that each vehicle delivers the goods to the customers within a specific time interval, and there are two types of time windows [3]. The vehicle may not arrive more than the customer's node closing time or before the customer's node open time. The best route must be on time and disciplined capacity as well as minimum distance and cost.

2. Optimised Genetic Algorithm

Holland [4] developed Genetic Algorithms (GA) and continued to use it to solve complex problems. When the optimal solution fails to find another search, GA is often tried to solve the problem. Several previous studies, including Whitley, Starkweather [5], Sponser [6], and Marin, Gonzalez [7], apply GA to solve complex problems when optimal solutions fail to be found by another search. GA can solve complex problems with relatively shorter computation time to reach near-optimal solutions. Recent research that uses genetic algorithms in complex Vehicle Routing Problems (VRP) problems have been carried out by Mohammed, Abd Ghani [8], de Oliveira da Costa, Mauceri [9], Liu and Jiang [10], and Sethanan and Jamrus [11]. Complex multi-constraint problems were successfully solved in their study, and came up with a suitable solution. According to Ibrahim, Masudin [12], Mohammed, Abd Ghani [8], Ho, Ho [13] and de Oliveira da Costa, Mauceri [9], GA is very efficient for solving problems with high complexity such as VRP.

An optimised genetic algorithm is applied in solving the problem of determining the vehicle route of a shipping and logistics company. This problem deserves to be a real example of the VRPPDTW application. The flow chart optimised genetic algorithm used can be seen in Figure 1.



The implementation of an optimised genetic algorithm in VRPTW problems can be seen in the following algorithm:

Algorithm 1: The CVRPPDTW model

1. start;
 2. population (a set of the route) initialization, use randomized method and nearest neighbor algorithm to form chromosomes (routes) structure;
 3. evaluate the 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, will 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;
 - 5.4. generate random numbers (R_m) in the range 0.00 to 1.00, will be 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;
 - 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.

2.1. Establishment of Initial Population

Generally, the initial population is formed randomly, but the algorithm developed in this study uses two random and heuristic methods. According to Ibrahim, Masudin [12], Nearest Neighbor is a heuristic method that can be used quite simply and produces a good initial population. Each point is represented by one number, and then it can be called a gene. A valid single route will be 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. Fitness

The fitness value in VRPPDTW cannot be directly equated with the basic VRP model. In general, the fitness value can be calculated using the total distance traveled or transportation costs. However, in this developed algorithm, the fitness value will depend not only on distance but also on capacity penalties and time windows. Both types of penalties must be considered to maintain the quality of the route in terms of pick-up and delivery cases and time windows. The fitness value will be influenced by arrival time's suitability with branch office time windows and vehicle load discipline with its capacity. Therefore, these three components must be considered in determining the fitness value. The fitness value is formulated as follows:

$$fitness = \frac{1}{1 + total\ penalty} + \frac{1}{length\ of\ route}$$

2.3. Selection

The selection process is carried out at this stage. Gene selection is carried out on pre-formed chromosomes. The Roulette Wheel Selection (RWS) selection method is the most commonly used method. In this method, the parents obtained based on the proportion of the chromosome's fitness value, meaning that the chromosome with the best fitness value has elected a higher likelihood than other chromosomes. Chromosome parents elected at the next selection process enter the crossover process to produce offspring chromosomes.

2.4. Optimised Crossover

Not all chromosomes from parents are chosen for crossover. The crossover probability will determine whether the crossover is done or not. We developed an optimized crossover for the operator to produce better offspring and match expectations. In this section, we will explain the optimized crossover steps that are developed to produce new offspring. The distance between genes considered in the crossover stage to create a better new population, loop the following steps as much as the number of chromosomes:

- Step 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;
- Step 2: Find the distance between the crossover point node from the previous node $C_d = \text{distance gen } C_p - 1 \text{ to gen } C_p$
- Step 3: Determine the chromosomes that will 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;
- Step 4: Find the distance between the crossover point node to all nodes on the chromosome selected for a crossover;
- Step 5: Crossover the C_p genes with genes that have the closest distance;

2.5. Optimised Mutation

Similar to crossover, the mutation probability will influence whether the mutation process is carried out or not. In this study, the mutation process will consider the distance between genes. The hope is that considering the distance will produce a new population that has better fitness value. Apart from that, it can also shorten the computation time. Loop the following steps as much as the number of chromosomes:

- Step 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;
- Step 2: Find the distance between the mutation point node from the previous node $M_d = \text{distance gen } M_p - 1 \text{ to gen } M_p$
- Step 3: Find the distance between the node $M_p - 1$ to all other nodes on the same chromosome;
- Step 4: Mutation the M_p genes with genes that have the closest distance;

2.6. Establishment of a New Population

The new population, which was formed using a genetic algorithm, has previously calculated the fitness value. The new population will replace the old population if it has better fitness value. If the new population replaces the old population, all chromosomes begin in a generation replaced by chromosome results obtained after the crossover and mutation stages. On the other hand, if the resulting new one has a lower fitness value, it will not be used, and the algorithm will be repeated in the old population.

2.7. Repetition Number of Iterations

The selection process, evaluation of fitness, crossover, mutation, and new population formation is repeated until the number of iterations is reached to get the best results. The only chromosomes that

can survive are those with the best fitness value among other chromosomes. This process follows the principle of natural selection, "who is strong (has a better fitness value) will survive."

2.8. Determining the Best Solution

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

3. Result and Discussion

We compare the total cost of the existing route and the proposed route obtained with an optimized genetic algorithm through numerical experiments. The cost component used as a reference for calculation in this case study is the cost of fuel related to the total distance traveled from the vehicle. Table 1 presents data related to the total cost of each vehicle on the existing and proposed routes. The results of numerical calculations show that the total cost generated from the proposed route is less than the existing route. The proposed route's total cost is Rp. 196,619.09, and the total cost of the existing route is Rp. 247,649.55. It can be concluded that the route formed by an optimized genetic algorithm has succeeded in producing a route that is closer to optimal. In calculating the total cost, we have not considered the costs incurred because the vehicle arrives outside the destination time windows node and costs due to overload. However, we will compare the total penalties incurred where this penalty is related to the violation of time windows for each destination node and vehicle capacity.

Table 1. Total cost comparison Actual Route & Improved Hybrid Genetic Algorithm

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

The VRPPDTW problem in this paper uses the type of soft time windows. Penalties apply if the vehicle arrives outside the available time window (penalty time windows). Likewise, the capacity penalty applies if vehicles carry more than their capacity. The more penalties given on a route illustrate, the worse the route. We continue to analyze in-depth the penalties that occur on the proposed and existing routes.

Table 2. Fitness value Comparison of Actual Route & Improved Hybrid Genetic Algorithm

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

Table 2 shows the comparison of the total penalty value of the proposed route and the existing route. The numerical calculation results show that there are two penalty capacity on the existing route. That means that there are two times the excess capacity of the vehicles on the existing route. This excess capacity occurs because the route is not optimal. For example, a fully-loaded vehicle visits a destination node that has fewer delivery requests than pick-up requests. If this happens in the real system, the vehicle will not carry the pick-up request because the capacity is insufficient. So it is

crucial to pay attention to the number of pick-ups and deliveries, not just the order of delivery with a minimum distance.

On the other hand, there are four penalty time windows (closing) on the existing route. This can be interpreted as an event that the vehicle arrives at the destination node when the destination node has closed or the service time has ended. In VRPPDTW problems, the vehicle must arrive between the time windows (opening) and time windows (closing). The destination node cannot serve vehicles that arrive too fast or are late, so they have to wait or come back afterward. This resulted in economic losses and led to longer travel times. From the results of these numerical calculations, it is also seen that no penalties have occurred on the proposed routes generated by the improved genetic algorithm. Thus, it can be concluded that all vehicles always arrive at each destination node according to the time windows and never overcapacity.

4. Conclusion

In this paper, we study the vehicle routing problem pick-up and delivery with time windows. Where is related to the constraints of vehicle capacity and time windows of the destination node. Besides, vehicles not only deliver goods but also collect goods at the same time. We developed genetic algorithms with optimised crossover and mutation operators for solving VRPPDTW cases. Numerical experiments show that the substrate route is closer to optimal than the existing route. For future work, the researcher can modify the crossover and mutation stages to solve other VRP problems.

Acknowledgments

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

References

- [1] Lin CKY. A cooperative strategy for a vehicle routing problem with pick-up and delivery time windows. *Computers & Industrial Engineering*. 2008;55(4):766-82.
- [2] Mahmoudi M, Zhou X. 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*. 2016;89:19-42.
- [3] Elshaer R, Awad H. A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants. *Computers & Industrial Engineering*. 2020;140:106242.
- [4] Holland JH. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*: University of Michigan Press; 1975.
- [5] Whitley D, Starkweather T, Bogart C. Genetic algorithms and neural networks: optimizing connections and connectivity. *Parallel Computing*. 1990;14(3):347-61.
- [6] Sponsler JL. Genetic algorithms applied to the scheduling of the hubble space telescope. *Telematics and Informatics*. 1989;6(3):181-90.
- [7] Marin FJ, Gonzalez FJ, Sandoval F. The Routing Problem in Traffic Control Using Genetic Algorithms. *IFAC Proceedings Volumes*. 1994;27(3):187-91.
- [8] Mohammed MA, Abd Ghani MK, Hamed RI, Mostafa SA, Ahmad MS, Ibrahim DA. Solving vehicle routing problem by using improved genetic algorithm for optimal solution. *Journal of Computational Science*. 2017;21:255-62.
- [9] de Oliveira da Costa PR, Mauceri S, Carroll P, Pallonetto F. A Genetic Algorithm for a Green Vehicle Routing Problem. *Electronic Notes in Discrete Mathematics*. 2018;64:65-74.
- [10] Liu R, Jiang Z. A hybrid large-neighborhood search algorithm for the cumulative capacitated vehicle routing problem with time-window constraints. *Applied Soft Computing*. 2019;80:18-30.

- [11] Sethanan K, Jamrus T. 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*. 2020;146:106571.
- [12] Ibrahim MF, Masudin I, Saputro T. A Hybrid Genetic Algorithm Implementation For Vehicle Routing Problem With Time Windows. *Jurnal Ilmiah Teknik Industri*. 2015;14:196-204.
- [13] Ho W, Ho GTS, Ji P, Lau HCW. A hybrid genetic algorithm for the multi-depot vehicle routing problem. *Engineering Applications of Artificial Intelligence*. 2008;21(4):548-57.