Compatibility Themed Solution of the Vehicle Routing Problem on the Heterogeneous Fleet

Metin Bilgin
Department of Computer Engineering, Bursa Uludağ University, Turkey
metinbilgin@uludag.edu.tr

Nisanur Bulut
Department of Computer Engineering, Bursa Uludağ University, Turkey
501731014@ogr.uludag.edu.tr

Abstract: *In this study, we discuss the solution to the vehicle routing problem for a heterogeneous fleet with a depot and a time window satisfied by meeting customer demands with various constraints. A 3-stage hierarchical method consisting of transportation, routing, and linear correction steps is proposed for the solution. In the first stage, customer demands have the shortest routing. They were clustered using the annealing simulation algorithm and assigned vehicles of appropriate type and equipment. In the second stage, a genetic algorithm was used to find the optimal solution that satisfies both the requirements of the transported goods and the customer requirements. In the third stage, an attempt was made to increase the optimality by linear correction of the optimal solution found in the second stage. The unique feature of the application is the variety of constraints addressed by the problem and the close proximity to real logistics practice*.

Keywords: *Time window, vehicle routing problem, multiple traveling salesmen problem, heterogeneous fleet, simulated annealing algorithm, genetic algorithm, optimization*.

Received May 19, 2020; accepted September 22, 2021
https://doi.org/10.34028/iajit/19/5/9

1. Introduction

Logistics is generally a collection of methods that enable the flow of customer requests to destination points. This flow is by land, sea, air, or rail. It attempts to minimize resource costs such as fuel, distance traveled, and labor, and to deliver customer requests efficiently and on time.

The traveling salesman problem is the circulation of destination points, assuming you get to each destination point only once [17]. In logistic applications, the basic unit is the traveling salesman problem and the shortest route is a critical value. This is because as the distance traveled increases, the environmental damage increases, as well as the time, lost to the company, and the material damage increases. For this reason, companies strive to protect the environment in their logistics activities by considering the consequences such as air pollution, greenhouse effect, natural disasters, and depletion of natural resources [4].

With the diversification of transported goods and services, the Vehicle Routing Problem (VRP) has been diversified in the added constraints. The vehicle fleet, which is used according to the specific needs of the transported goods, was also diversified, and an assignment problem between warehouse (vehicle) and transported (good) was created. Even if the vehicle fleet is homogeneous, the classical VRP belongs to the Non-deterministic Polynomial time (NP) to be solved. Solving the problem becomes more difficult when a heterogeneous fleet is used. However, in real logistics applications, it is not common to satisfy all customer requests with homogeneous vehicles, i.e., vehicles of the same type and capacity [2].

The VRP problem in this study exhibits the constraints that occur in real logistics applications.

Some requirements must be met during the transportation of customer demands. These requirements arise due to the special demands of the customers, the special equipment required by the transported goods, or the reconciliations between the customer and the trader that have penal consequences.

Depending on the type of assets being transported, different types of removal requirements arise. For example, vehicles with tankers should be used to transport liquid goods such as milk, oil, and fuel, while goods such as biscuits that need to be kept in boxes should use vehicles that allow pallets to be transported. For goods containing raw materials or goods that need to be transported at a certain temperature, such as milk and chocolate, vehicles must have special heaters. When transporting hazardous and flammable materials, this can lead to unacceptable results. For this reason, the transport vehicles must have the necessary protective equipment [21]. There is a process of unloading the goods loaded onto the vehicle from the vehicle. Some customers want to use vehicles that enable transportation, especially with lifts to reduce the unloading time of the vehicle when delivering goods. When the vehicle reaches the delivery point, acceptance tests are performed on the goods being transported. Goods that are not at the specified temperature, crushed, worn, or damaged cannot pass the acceptance tests. Therefore, it is inevitable to use a heterogeneous fleet in the logistics process, which includes vehicles with different capacities, types, and characteristics.

The resolution of VRP with a fleet of vehicles of different capacities is called a Heterogeneous Vehicle Routing Problem (HVRP). It was brought to the literature by HVRP Golden *et al*. [7] HVRP has been varied over time and has taken a large place in the research field with the presence or absence of time window, limited or unlimited fleet size, and being added to constraints such as type and capacity. With their work, Baldacci and his colleagues discussed HVRP and offered intuitive solutions. Hoff and his colleagues have dealt with HVRP in different transportation sectors such as sea and road [10]. The vehicle fleet in this problem consists of a limited number of vehicles of different types, capacities, and properties.

Delivery points that accept goods usually operate on an appointment system. Customers can accept goods on certain days of the week, as well as on certain days of the week. In joint arrangements between the distribution company and the customers, the acceptance times of the goods can be further adjusted with payables. In this case, the vehicle is expected to arrive at the delivery point or have its load unloaded at the destination point on the date specified for the acceptance of goods. This appointment time is called a time window.

When it is necessary to reach the destination points within predetermined time intervals, the classical VRP, Time Window Vehicle Transforms İnto A Routing Problem (TWVRP) [13]. When the vehicles have to arrive at the delivery point and unload their goods within the predetermined time window, this time window is called a narrow time window. If the vehicle cannot reach or unload the delivery point within the time window, it may incur losses such as fines or the vehicle may not be allowed to unload its goods. In this case, vehicles may have to wait until the next appointment to unload at the delivery point. This is reflected in time and service costs.

In this study, the concept of the time window is customized for each customer and it is not enough that the vehicle arrives at the customer point within the time window. The vehicle is expected to unload and deliver the goods to the customer within the time window. However, for some customers, this is not enough. This is because there is an additional time cost associated with unloading the goods from the vehicle. Therefore, there is also an unloading time for unloading the vehicles. During this time, the vehicle is supposed to unload and deliver its goods to the customer. If the vehicle unloads its goods at the delivery point within the specified unloading time, it is considered as on-time delivery. In the study, the arrival of the vehicle at the customer point within the targeted time frame is referred to as "on-time arrival", while the situation where the vehicle unloads the goods within the targeted time frame is referred to as "on-time delivery".

Delivery points that accept goods are areas that store goods. Vehicles use the ramp gates shown in Figure 1 to unload their goods at customer points. The number of ramps is not infinite and due to the volumetric characteristics of the vehicles, such as length, height, width, or one or two stories, not every vehicle can approach every ramp door. Therefore, the vehicles must be appropriately equipped with the ramp doors at the delivery points when the distribution plan is prepared.

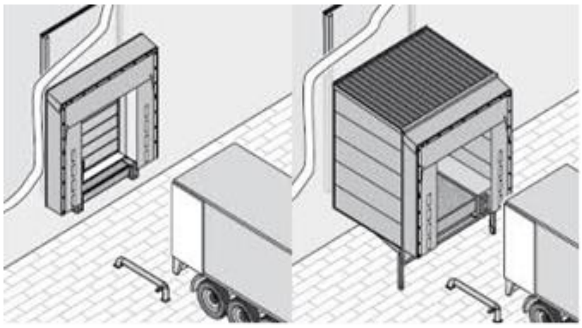

Figure 1. Ramp gate identification.

In addition, the vehicle must have a ramp door suitable for its volumetric characteristics to unload its goods at the delivery point. This ramp door must be suitable for use. Otherwise, when the vehicle reaches the delivery point, it will not be able to unload its goods. Even if the vehicle arrives on time and reaches the delivery point, it cannot unload its goods and deliver on time if it cannot find a ramp door. Therefore, the compatibility of the vehicle with the ramp doors at the delivery point directly affects the quality of on-time delivery.

The remainder of this paper is organized as follows. Section 2 presents a review of the related works and section 3 provides current definitions. Section 4 presents the mathematical statement of the problem and section 5 presents the application. Section 6 presents the shipping phase. Section 7 presents the linear correction phase, section 8 presents results and section 9 provides the discussion.

2. Related Works

The VRP, first discussed by Dantzig and Ramser, refers to a vehicle's shortest route from origin to destination [3]. In their studies, Clarce and Wright proposed the savings algorithm for the vehicle routing problem that starts and ends in a warehouse and uses equally qualified tools to distribute a given set of customer requests. The savings algorithm aims to reduce the cost, such as the total travel distance and the number of vehicles used, without exceeding the vehicle capacity while satisfying the customer requests. [14].

The quality of the shortest path is the most important constraint in VRP. This is because the shorter the total distance, the lower the wear and tear of the vehicles, the number of gasses released into the environment, the risk of accidents, and the travel time. Eryavuz and Gencer [5] addressed the VRP problem in a comparative study for a service application where the vehicles are fully utilized while the total distance is shortened. The routes created were determined using the savings algorithm and the random savings algorithm, and the route developer was improved using the 2-opt and the Or-opt algorithms, which are intuitive. In the obtained primary solutions, the total path distance was a more optimal result with the random savings algorithm, while the savings algorithm results became more optimal after the tour developer algorithm was used.

Although HVRP is a common situation in real logistics applications, the number of studies addressing this issue in the literature is very small, and studies addressing HVRP mostly focus on the physical characteristics of vehicles. To the best of our knowledge, there is no study that addresses HVRP by considering the compatibility of vehicles at the ramp door at the destination point. In addition, there are very few studies that consider equipment such as air curtains or air coolers instead of the volumetric characteristics of width and length, or that measure the usability of auxiliary vehicles such as forklifts during loading and unloading.

In the heterogeneous fleet, vehicles vary in size due to differences such as length and width. These dimensions directly affect the loading and unloading operations. Considering this situation, Leung *et al*. [12] proposed a solution to the HVRP with an unlimited number of vehicles by using an annealing simulation algorithm. They confirmed the effectiveness of their proposed method using 360 benchmark examples.

The total road distance directly affects the working and break times of drivers. Drivers' working and break times are protected by legal constraints. Duygu and her friends have developed a model based on the Tabu algorithm that considers the drivers' break times when processing the HVRP. When the time window is added to the HVRP, the time window turns into a Heterogeneous Fleet Vehicle Routing Problem (HTWVRP). With this model, they aimed to minimize the waiting cost due to early arrival and service cost due to late delivery for deliveries that do not match the time window. Later, they developed their solutions from optimization with a post-optimization process. They expressed that their results were successful [19].

One of the most important values in many VRP related problems is timely delivery to the customer. This delivery time can be a fixed time based on the customer's demand, but it can also be a time window with a start and end interval. In cases with soft time constraints, a tolerance can be achieved in criminal cases if customer demand cannot be met within the appropriate time frame, while in cases with tight time window constraints, there are consequences such as penalties or failure to deliver.

Taillard *et al*. [18] and his colleagues studied the soft time window constraint for a single-depot vehicle routing problem on the homogeneous fleet using a tabu search algorithm. If the vehicle arrives at the destination point before the lower bound of the time window, it has to wait in vain, and if it arrives after the upper bound of the time window, it is considered late. For this reason, a penalty is set for each customer. For customers with large penalty coefficients, the time window becomes tight, while for customers with small penalty coefficients, the time window becomes soft. The Lagrange method is used to functionalize the objective function in this context.

As customer demand has increased and transportation methods have diversified, so have the variations of the VRP. The single-depot VRP problem has evolved into a multi-depot VRP problem. Baea and Moonb [1] proposed an intuitive genetic algorithm to solve the problem of routing service tools used for the maintenance and repair of electronic devices. They found that they can find suitable solutions to relatively large problems with this method, which uses a complex integer model. With the increasing customer demands and diversified transportation methods, the variations of VRP have also increased. The single-depot VRP problem has evolved into a multi-depot VRP problem. Baea and Moonb proposed an intuitive genetic algorithm to solve the problem of routing service tools for electronic equipment maintenance and repair. They stated that with this method, which uses a complex integer model, they can find suitable solutions for relatively large problems.

For food that is not stored under proper conditions and will spoil if not delivered on time, the distribution plan in logistics should be optimal. Tirkolaee *et al*. [20] have proposed a method for VRPTW which contains many warehouses. This method is based on ant colony heuristic algorithm and uses mixed-integer linear programming model. With their proposed model, they aimed to satisfy customer demands by minimizing the cost incurred due to early or late delivery, total travel time, and vehicle usage cost. They conducted sensitivity experiments on three types of random problems in small, medium, and large size and proved the success of the model on the problems.

Together with Seixas and Mendes [16] HVRP, they presented a mathematical model based on the source flow variables, taking into account the total driving time of the drivers and the constraints on the average speed of the vehicles. Their problems involve the following rules: Each vehicle can travel multiple routes during the day, and each vehicle has a single driver. The initial solution was created using a structural heuristic algorithm and then scanned using the Tabu algorithm. They used the column evolution method to evaluate the results of the study. As a result of the study, they obtained relatively successful solutions.

When the studies on the time window were examined, it was found that the focus was on the criminal situations or the situation that the vehicle reaches the destination point within the time window. In this study, the time window is divided into two concepts: timely delivery and timely arrival. Thus, the time window interval also indicates the situation of unloading the goods of the vehicle. Moreover, while the vehicle is unloading its goods at the destination point, the compatibility between the volumetric characteristics of the vehicle and the doors of the destination point is studied. These two approaches are innovations that the study brings to the literature. In addition, the vehicle must find an empty ramp door to unload its goods at the delivery point. This study also investigated the possibility of finding an empty ramp door within the vehicle's time window. The study examples dealing with this situation were not found during the literature search.

When examining the studies conducted in the literature, it is found that VRP problems using a heterogeneous fleet based on the shortest path constraint and containing a narrow time window are very rarely addressed. However, the constraints to be added can be extended. The obligations imposed by the time window, the need for equipment such as air curtains specific to the cargo carried, the duration of the break based on the number of drivers, and many other such constraints can be added to the VRP. In reviewing the literature, it was noted that these restrictions were addressed piecemeal and not evaluated as a whole. This is because the VRP is inherently difficult to solve, and with the added constraints, solving the problem becomes even more difficult. However, even if all constraints are fully satisfied and all possible situations are considered, the problem with all constraints cannot always be expected to reach a final optimal solution. This is because real customer requirements are stochastic requirements. A vehicle malfunction, a customer canceling his order for various reasons, a traffic accident that may occur, a road closure due to natural disasters are all possible situations. For this reason, in solving the problem in our study, we considered the constraints as a whole and evaluated them with the concept of compliance. Thus, a decision support system was developed that can quickly adapt to possible surprising changes. This study aims to enrich the literature and solve a problem that is suitable for real logistics applications.

3. Definition

The problem addressed in this study deals with the delivery of goods of different types of vehicles from the central warehouse to the distribution points using the shortest route. As can be seen from this basic definition, the problem addressed is the time window problem of vehicle routing with a heterogeneous fleet of m vehicles from a centrally accepted warehouse, as shown in Figure 2.

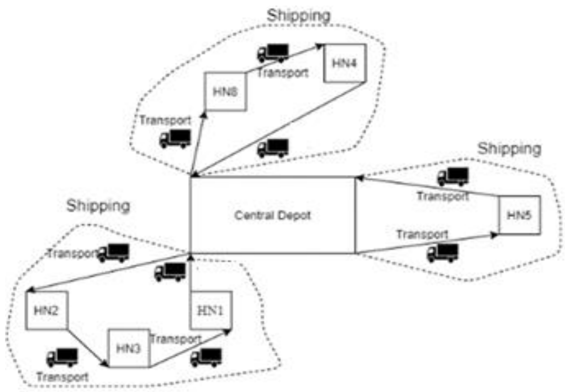

Figure 2. Identification of problem entries.

The problem consists of rules and concepts of harmony. This section presents information on the constraints and concepts of the problem.

3.1. Constraints in the Problem

- Each customer request must be met with appropriate equipment and tools.
- Each customer request shall be fulfilled with only one vehicle.
- Each vehicle must take the shortest route to the delivery location.
- Vehicles will depart from the central depot and stop at each delivery point, then return to the central depot.
- The vehicles must deliver the customer requests within the specified time window.

3.2. Concepts in the Problem

- Distance Balance (DB): refers to the balance of total distance traveled by vehicles during delivery.
- Efficiency Balance (ED): it refers to the total time spent by the vehicle without transporting goods.
- Ramp Balance (RB): if you send more than one vehicle to the delivery point at the same time, the loading process will be prevented. It refers to the number of ramp doors that are used at the same time at the delivery point, i.e., overlap.
- Timely Arrival Balance (TAB): specifies that the vehicle must drop off its load at the delivery point within the time window.

These concepts used for balance research in the solution phase of the problem provide information about the quality of the solution devised.

4. The Mathematical Statement of the Problem

The depot point is taken as 1.

- Clusters:

R: the set of shipments r=1,…, R
$i,$: set of customers $i, = 1, \dots, n$
t : set of different types of vehicles $t = 1, \dots, T$
v : number of vehicles of different types $v = 1, \dots, A_t$

- Parameters:

Q_t: t. type of vehicle capacity
c_{ij}: distance from customer i to customer j
A_t: t. number of vehicles of the genre
di: i customer request
M_{tv}: t. type v. cost of using the vehicle
p_t: route cost per kilometer of type t
LB_k: k timeframe starts $\forall_k \in K$
UB_k: k timeframe ends $\forall_k \in K$
h_{kij}: travel speed from the customer I to customer j in k timeframe $\forall_i, j \in N; \quad \forall_k \in K$
S_i: i. customer service time $\forall_i \in N_c$
T_i^+: time to enter the vehicle i. customer $\forall_i \in N_c$
T_i^-: time to leave the vehicle i. customer $\forall_i \in N_c$

- Decision Variables

$$X_{ijtv} = \begin{cases} 1, & \text{If the v. vehicle of type t goes from i to j} \\ 0, & \text{In another case} \end{cases}$$

$$f_{tv} = \begin{cases} 1, & \text{If the v. vehicle of type t is used} \\ 0, & \text{In another case} \end{cases}$$

$$T0_i^+ = \begin{cases} 1, & If\ the\ vehicle\ turns\ from\ the\ i\ point \\ & the\ depot\ the\ return\ time \\ 0, & \text{In another case } \forall_i \in N_c \end{cases}$$

TT_{ij}= Transition time from point i to point j $\forall_{i,j} \in N$
u_i and u_j defined as a positive variable used in the sub tour.

- Purpose function: total distance of each transport

SmallestR(r)
$=\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{t=1}^{T}\sum_{v=1}^{A_t} c_{ij} p_t x_{ijtv} + \sum_{t=1}^{T}\sum_{v=1}^{A_t} M_{tv} f_{tv}$

Constraints

$$\sum_{i=1}^{n}\sum_{j\neq i}^{n}\sum_{v=1}^{A_t} d_i x_{ijtv} \leq Q_t \qquad (1)$$

$$\sum_{j=1}^{n}\sum_{v=1}^{A_t} v_{1jtv} \leq A_t \qquad (2)$$

The sum of customer demands on the route of vehicle V of type T must not exceed the capacity of that vehicle. Equation (1) expresses this situation. Equation (2) controls the number of vehicle types. For each vehicle type, no more than the total number of vehicles of that type can leave the depot. Because if all vehicles leave the depot, the allocation is meaningless.

$$\sum_{i\neq k}^{n} x_{iktv} - \sum_{j\neq k}^{n} x_{kjtv} = 0 \quad \forall_k = 1, \dots, n, \forall_t, \forall_v \qquad (3)$$

Equation (3) refers to the flow constraint. If each tool v starts from customer i and travels to customer k, it travels from customer k to another customer ($j \neq i$). Equation (4) states that each customer j must be reached from another customer i. In Equation (5), Miller et al. express the partial tour required in the literature. [14] In Equation (6), v. If the vehicle is used, f_tv=1 is considered and must return to the depot. Equation (7) is the condition that establishes the link between the decision variable x and the decision variable f.M is a (comparatively) large number

$$\sum_{i=1}^{n}\sum_{t=1}^{T}\sum_{v=1}^{A_t} \forall_{ijtv} = 1 \quad \forall_t = 1, \dots, n, j \neq i \qquad (4)$$

$$u_i - u_j + n\sum_{t=1}^{T}\sum_{v=1}^{A_t} x_{ijtv} \leq n-1 \quad \forall_{i,j} = 2, \dots, n, i \neq j \qquad (5)$$

$$\sum_{i=1}^{n} v_{i1tv} = f_v \qquad (6)$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ijtv} \leq M * f_{tv} \qquad (7)$$

$$x_{ijtv}, f_v \in i, j = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, \forall_t \qquad (8)$$

$$\sum_{k\in K}\sum_{l\in K} y_{ijkl} = x_{ij} \quad \forall_i, j \in N \qquad (9)$$

$$TT_{ij} \leq T_j^+ - T_i^- + M_{ij}(1 - x_{ij}) \quad \forall_i \in N, \ \forall_j \in N \qquad (10)$$

$$TT_{ij} \geq T_j^+ - T_i^- + M_{ij}(x_{ij} - 1) \quad \forall_i \in N, \ \forall_j \in N \qquad (11)$$

$$TT_{j0} \leq T0_j^+ - T_j^- + M_{j0}(1 - x_{j0}) \quad \forall_j \in N \qquad (12)$$

$$TT_{j0} \geq T0_j^+ - T_j^- + M_{j0}(x_{j0-1}) \quad \forall_j \in N \qquad (13)$$

According to the Equation (9), the vehicle's journey must start within a certain time period and the vehicle must finish its journey within the same or a different time period. The Equations (10, 11, and 13) ensure that the entry and exit times to the customer points are associated with the transition times between the points. The Equations (10, and 11) determine the access times from all points to other customer points, and the Equations (12, and 13) are the constraints that determine the return times from the customer points to the depot.

$$\forall_{i,} \forall_j, \ \forall_k, \forall_l \in K \quad d_{ij} - M(2 - x_{ij} - y_{ijkk}) \leq (T_j^+ - T_i^-) h_{kij} \qquad (14)$$

$$d_{ij} - M(2 - x_{ij} - y_{ijkl}) \leq (UB_k - T_i^-) h_{kij} + \sum_{k<m<l} (UB_m - LB_m) h_{mij} + (T_j^+ - LB_l) h_{lii} \quad l > k \qquad (15)$$

$$d_{jo} - M(2 - x_{j0} - y_{j0kk}) \leq (T0_j^+ - T0_i^-) h_{kj0} \qquad (16)$$

$$d_{j0} - M(2 - x_{j0} - y_{j0kl}) \leq (UB_k - T_i^-) h_{kj0} + \sum_{k<m<l} (UB_m - LB_m) h_{mj0} + (T_j^+ - LB_l) h_{li0} \quad l > k \qquad (17)$$

Equations (14-17) ensure that the transition between the speed of the vehicle and the distance traveled between time windows is related. Equation (15) refers to the return of the vehicle from a customer within time window k and the return to the central depot point within the same time window. Equation (16) expresses the exit of the vehicle from the customer point within time window k and to the central depot point within a larger time window.

- Equations:

$$\sum_{j\in N}\sum_{k,l\in K} y_{ijkl}^{UB_k} >= T_i^- \ \forall_i \in N \quad (18)$$

$$\sum_{j\in N}\sum_{k,l\in K} y_{ijkl}^{LB_k} <= T_i^- \ \forall_i \in N \quad (19)$$

$$\sum_{j\in N}\sum_{k,l\in K} y_{ijkl}^{UB_l} >= T_i^+ \ \forall_i \in N \quad (20)$$

$$\sum_{j\in N}\sum_{k,l\in K} y_{ijkl}^{LB_l} <= T_i^+ \ \forall_i \in N \quad (21)$$

$$\sum_{j\in N}\sum_{k,l\in K} y_{i0kl}^{UB_l} >= T0_i^+ \ \forall_i \in N \quad (22)$$

$$\sum_{j\in N}\sum_{k,l\in K} y_{i0kl}^{LB_l} <= T0_i^+ \ \forall_i \in N \quad (23)$$

$$T_i^- = T_i^+ + s_i \ \forall_i \in N \quad (24)$$

(18-24) define the ratio of the lower and upper bounds of the time windows to the entry and exit times of the vehicle from one customer point to all other customer points. The Equations (21, and 22) define the ratio of the time window boundaries to the vehicle return time from the customer point to the central depot point. Equation (23) states that the time the vehicle leaves the customer point is equal to the sum of the entry time and the service time spent at the customer point.

5. Application

The application consists of three successive phases. In the first phase, the customer requests are clustered to have the shortest total distance based on the delivery locations. In the second phase, the clustered customer demands are distributed among the vehicles using a genetic algorithm to find the most suitable solution and the solution that is most compatible with the given constraints is searched. The solution generated by the genetic algorithm is processed linearly.

In order to make problem-solving more understandable and easier, some models were created for the problem inputs. These models are shown in Figure 3.

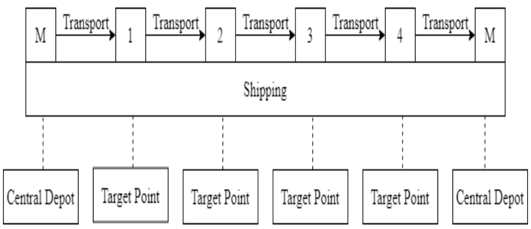

Figure 3. Transport and shipping relation definition.

- Source Point (SP): denotes the starting point for the transport of goods between two points.
- Destination Point (TP): refers to the point at which goods are delivered as part of the transport between two points.
- Depot Point (DP): this is the starting and ending point of transportation. This refers to the central point.
- Transport: this is the transport of goods requested from a source point to a destination point. Each customer request is a transport.
- Shipping: this is the bundling of shipments by arranging them to have the smallest total distance by road. The first source point of each shipment is the central depot point.
- Real vehicle: it refers to the available vehicles.
- Virtual vehicle: when the number of available vehicles is insufficient for customer demand, it is assumed that there are vehicles of the type and characteristics suitable for customer demand, and they are identified as virtual vehicles.

6. Shipping Phase

The first phase of problem-solving is the shipping phase. Shipping means bundling shipments. The goal is for the total road distance in the cluster to be as short as possible. Each cluster is designated as a shipment and identified with vehicles of the appropriate type and property. The annealing simulation algorithm is used to determine the shipments. To ensure that all shipments have the shortest total distance, the two methods were compared. The shipping phase was performed using the method that was found to be advantageous.

In the annealing simulation proposed by Kirkpatrick *et al*. [9] the process of slow cooling of the solids after heating is mimicked. In annealing, the material is softened by heating and it is ensured that the molecules of the material become regular by gradual heating. In the optimization application, this behavior is mimicked. The search process starts with a high-temperature value so that the search space is large. Each time the temperature is lowered, adjacent solutions are generated in a certain number of cycles. In this way, possible solutions are circulated. The annealing simulation, based on the random principle, allows accepting also bad solutions that do not contribute to the objective function, preventing them from getting stuck in the locally best places.

Table 1. Annealing simulation parameter values.

Temperature	Temperature Reduction Rate	Loop Numbers
5,10,15,20,25,50,75	0.99	250,500,750,1000

In practice, the simulation of annealing was started with a randomly generated starting solution. The temperature value was reduced at the temperature reduction rate given in Table 1. At each temperature reduction, neighboring solutions were examined by performing the same number of cycles as the number of cycles given in Table 1. The temperature parameter also directly affects the acceptance of the neighboring solution as an existing solution. If the temperature parameter is too high, it ensures that the searched space is very large. Keeping the temperature parameter too low reduces the jumps in the search space, allowing potential good solutions to be overlooked. For this reason, the temperature was lowered gradually, starting from a high-temperature value. Lowering the temperature quickly reduces the search time, but may also prevent an adequate search for the best solution. For this reason, the temperature value was gradually lowered by a factor of 0.99, as shown in Table 1. Annealing simulation complexity is O(m * n * k)) with m the temperature, n the temperature reduction rate and k the size of the loop numbers.

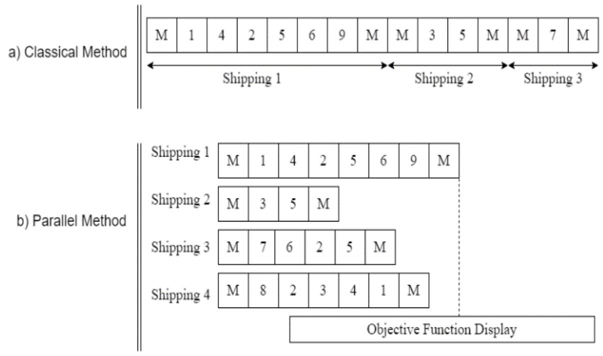

Figure 4. Objective function calculation method.

In the annealing simulation algorithm, the objective function computes the total path distance. To decide which solution is the best, a different approach was proposed. The classical method is described in Figure 4-a. The total distance of the N-generated transports was accepted as the score value of the objective function. In comparison to this method, a method that mimics the scheduling of parallel machines was proposed. In this method, each vehicle is modeled as a parallel machine, as shown in Figure 4-b, and is accepted as sequential work at the transport destination points. The maximum value of completion of all jobs in parallel machines is the CMAX value [11]. Here, the CMAX value is the total road distance of the transport that has the largest total distance among the existing transports.

6.1. Using Simulation Algorithm with the Parallel Approach In Problem Solution

14 Plans at various scales were used in the transportation phase. These plans were used to compare the classical method and the parallel approach method. While there are 48 transports in the smallest plan, there are 140 transports in the largest plan.

In the proposed parallel approach, the destination change is made simultaneously for all shipments. The total road distance of the shipment that has the largest total distance in the solution set is considered as the score value. The parallel approach and the results of the classical method are shown in Figure 5.

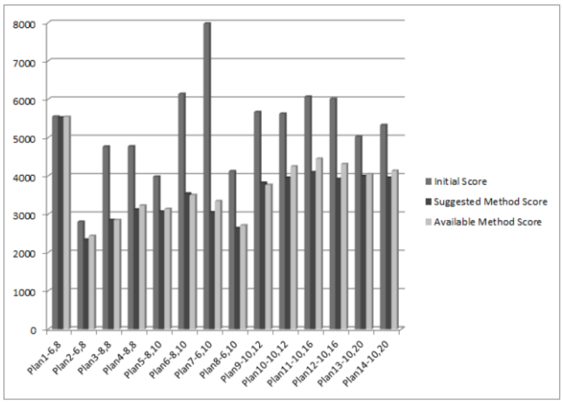

Figure 5. Result display of annealing simulation algorithm.

When comparing the classical method and the parallel method, it is noticed that both methods have improved the initial solution, but the proposed parallel solution approach has given a greater improvement over the initial solution than the classical method. For this reason, when the parallel solution approach was used in the dispatch phase, the score values generated by the objective function were compared. Algorithm (1) contains the pseudo code that describes the use of the annealing simulation algorithm in solving the problem.

Algorithm 1: Pseudocode of annealing simulation algorithm

1. It starts with the initial temperature value.
2. Neighboring solutions are produced as much as the specified number of cycles. The production of neighbor solutions for each cycle is as follows:
2.1. Shipments are produced by making random selections among all transports.
2.2.Two randomly determined target points are displaced for each shipping.
2.3. The total path distance is calculated using the Euclidean formula for the ordered destination points in the shipping.
3. The total distance is calculated using the Euclidean formula for sequential arrival points in transport.
4. The difference (Δ) between the score value of the current solution set and the score value of the neighborhood solution set is calculated.
4.1. If Δ≥0 neighbor solution set is accepted.
4.2. If Δ <0 and r is a value in the range of 0 <r <1, $r < exp(\frac{-\Delta}{temperature})$*, the adjacent solution set is accepted.*
4.3. If the conditions for Δ are not met, the neighbor solution set is not accepted.
5. The loop control parameter is increased by 1.
6. The temperature control parameter is reduced by multiplying 0.99.
7. If the temperature value is> 1, steps 1,2,3,4,5,6 are repeated. If not, the algorithm will stop running.

6.2. Compliance Search Optimization Stage

This phase is the second phase of the solution process. In this phase, the consignments kept in the previous phase are matched with the vehicles. Conformity scores are computed to show how compatible this matching is with the constraints in the problem definition. A genetic algorithm was used to find the most compatible solution among the computational results.

The genetic algorithm introduced by Menger in the 1930s was the most commonly used algorithm for combinatorial optimization problems. The search space of the genetic algorithm, whose foundations were laid by Goldberg, is used to study the optimal solution for very broad problems. With the evolutionary approach, the entire search space is searched by searching a portion of the solution space for bad solutions. In this way, the solution time is reduced [6]. There are many types of genetic algorithms. The genetic algorithm used in practice is a simple genetic algorithm.

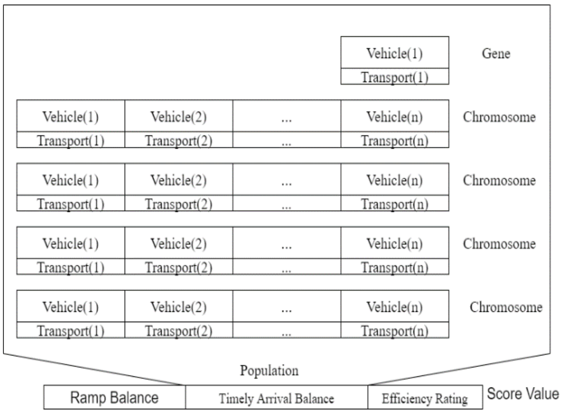

Figure 6. Definition display of genetic algorithm units.

The basic operations of using genetic algorithm for problem-solving are as follows: Modeling of the problem as a chromosome unit, determination of the objective function. These processes directly affect finding the optimal solution [8]. In the literature, there are many methods for representing chromosomes. Examples of these coding methods are binary coding, permutation representation, value coding, tree representation and encoding. In this study, the chromosome units used in the genetic algorithm were created using permutation coding technique. The chromosome units are shown in detail in Figure 6.

Setting the mutation rate to a high value prevents the GA from reaching a stable structure. As stated in clause 7.2, keeping the mutation rate dynamic prevents the GA from getting stuck (all chromosomes on the same plateau). The complexity of GA is $O(g(nm + nm + n))$, where g is the number of generations, n is the population size, and m is the individual size.

6.3.1. Using Genetic Algorithm in Compliance Search Optimization

In a genetic algorithm, the basic unit is a gene. Genes come together to form chromosomes. Each transport used in modeling a problem is included as a gene in the solution of a genetic algorithm. The constraints used to evaluate the problem are hidden in the passwords used by the gene. The transports are included as chromosomes in the genetic algorithm. All the shipments included in the problem were modeled as populations. The pseudocode of the general steps for using the genetic algorithm in solving the problem is shown in Algorithm (2).

Algorithm 2: Genetic algorithm pseudo-code

1. *The parameters are read. (Mutation rate, number of cycles)*
2. *The transitions involved in optimization are read.*
3. *Vehicles involved in optimization are read.*
4. *The number of vehicles suitable for transportation is checked. Because every transport cannot be done on every vehicle. If there are not enough vehicles, vehicles suitable for transportation types are produced. These vehicles are virtual.*
5. *Dictionary is created. The dictionary is a set of possibilities. It is created by assuming that every vehicle involved in optimization will carry out each transportation.*
6. *The following steps are performed for the maximum number of iterations or until the program duration has passed.*
    a. *The starting population is created.*
    b. *Compatibility scores are calculated for each individual in the starting population.*
    c. *Individuals are ranked according to their compliance scores and N individuals are determined as elite individuals.*
7. *The new population is produced.*
    a. *The crossing is applied to elite individuals and the new population is produced.*
    b. *The mutation rate is changed dynamically.*
    c. *A random number is generated in the range of 10-100, and if the mutation rate is less than this number, the mutation process is applied.*
8. *The resulting cluster is determined*

6.3.2. Creating New İndividuals

The creation of new individuals in the genetic algorithm is achieved by mutation and interbreeding. Both operations are based on the random principle. Prior to interbreeding, elite individuals are identified in the current population. Crossover is applied to the elite individuals as shown in Figure 7. Random genes are selected from these chromosomes and exchanged between these genes, resulting in new chromosomes.

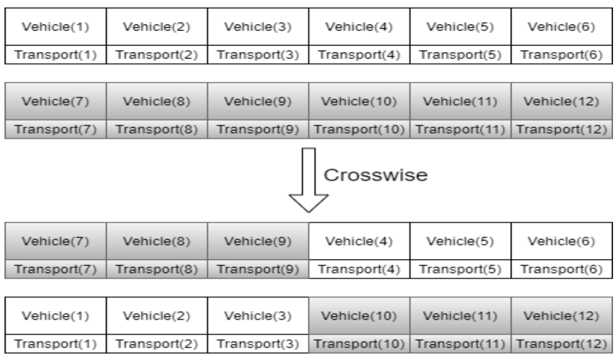

Figure 7. Crosswise process display.

The mutation rate is determined by comparing the score values of the population obtained by cross-operation and the population generated in the previous cycle with the current cycle. The mutation rate is dynamic throughout the working time of the genetic algorithm. The covariance value is calculated as shown in the pseudocode in Algorithm (3). This covariance value is used when the mutation rate is changed.

Algorithm 3. Determination of mutation rate pseudo code

Start:
A is the average of the set of score values of the previous population
B average of the set of score values of the current population
s Number of set A
i=0
While(i<s){
Covariance = Covariance +[$(A_i - A)$***$(B_i - B)$*]*
}
*MutationRate =(Covariance /S)*100*
if(MutationRate >50)
    MutationRate =50
else if(MutationRate <20)
    MutationRate =20
Return MutationRate

Two genes randomly selected from the chromosomes are shifted among themselves, as shown in Figure 8. This completes the mutation process.

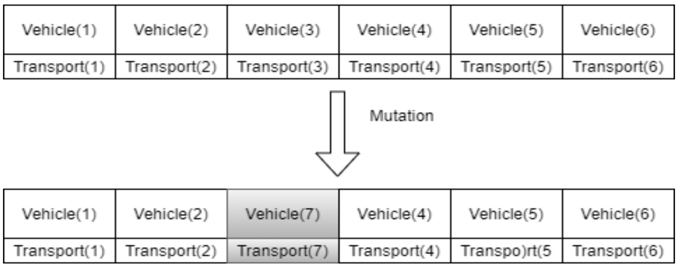

Figure 8. Mutation process representation.

6.3.3. Calculating Score Value

The score value is used to compare the results of the genetic algorithm. Each individual produced in the population has a score value and the individuals are ranked according to their score values. The most efficient score value in the population is considered as the score value of the population. The pseudo-code of the score calculation method is shown in Algorithm (4).

Algorithm 4: Calculation of score value

A is the total number of vehicles simultaneously located at a target point
K is the total number of doors at a target point
İf(A>K){
    There is a conflict situation.
    RU=K/A
}
DB= It is the score value of the algorithm that is run during the shipping process.
Z is the total number of vehicles reaching the target point in time.
TAB=Z/A
ED= Free time total of all vehicles/total number of transports
Score Value =DB+TAB+RB+ED

7. Linear Correction Phase

The linear correction process is the process of improving the solution of the result produced by the genetic algorithm. This improvement process aims to increase the number of vehicles arriving on time and decrease the ramp condition compliance value. The linear correction is not applied for every vehicle. Linear processing is performed only for non-virtual vehicles. Algorithm (5) provides the pseudo-code for the linear correction. The complexity of linear correction is $O(n^2)$.

Algorithm 5: Linear editing core code

The dictionary is a set containing the possibility of each vehicle performing every move.
As Vehicle set
Ts Transportation set
Foreach x in As
  Foreach t in Ts
    Add the status of carrying t in the x vehicle to the dictionary, Calculate score value
Foreach x in As
  if x is not a virtual tool
    skoreVehicle= Score value of vehicle x
    dictionaryTransportation = Transport of the x vehicle in the dictionary with the skoreVehicle
    vehicleTransportation = Transportation of x vehicle
    scoreDictionary = Find the best score of x tool in the dictionary
    if(scoreDictionary > vehicleTransportation)
      vehicleTransportation = dictionaryTransportation
For i to Ts
  For j to Ts
    İf(Score of Ts[i]<Score of Ts[j])
      Temporary= Ts[i]
      Ts[i]= Ts[j]
      T[j]= Temporary

8. Result

To observe the effect of the application, tests were performed with the same set of tools for many different transport numbers. The linear correction was applied to the original and final solutions. The scores before and after the linear correction are shown cumulatively in Table 2. The first row of data for each result in Tables 2 and 3 is the initial data. The algorithm has not worked yet. The data in the second row were obtained after the algorithm was run.

Table 2. Solution score values.

Solution Type	Timely Arrival Balance	Efficiency Rating	Distance Balance	Score Degree	Ramp Balance
Before Linear Correction					
Initial solution	2779.51	3592.41	3202.05	59832.18	512.51
	4059.83	3923.97	4009.51	80476.20	42.03
After Linear Correction					
Result solution	4038.072	3897.071	3989.496	78560.120	13.872
	4097.428	3941.782	4056.744	80595.603	2.03

The initial solution shown in Table 2 is the randomly generated solution before the genetic algorithm is executed. The resulting solution is the optimal solution generated by the genetic algorithm. The linear correction was applied to both solutions. Examining the results, it can be seen that the point values of the original solution have undergone positive changes before and after the linear correction. The ramp compliance value decreases as expected, while the other values increase.

Table 3. Total number of transports with timely arrival compliance.

Solution Type	Timely Arrival Balance	Efficiency Rating	Distance Balance	Score Degree	Ramp Balance
Before Linear Correction					
Initial solution	2899.86	3690.60	3164.42	420.91	61476.21
	4251.84	4067.03	3990.91	39.39	84262.25
After Linear Correction					
Result solution	4229.34	4042.17	3972.13	12.55	82376.58
	4229.34	4042.17	3972.13	12.55	82376.58

Table 3 shows the cumulative shipments that arrive on time based on the score values. The arrival times of shipments based on the compliance scores are shown in Table 3. When the final values of the initial solution are analyzed, it can be seen that the number of shipments that arrive on time increases due to the effect of linear correction. Comparing the initial solution with the final solution, it is found that the number of transports that arrive on time increases in the final solution. However, the linear correction applied in the final solution had no effect on the number of transports that arrived at the right time.

9. Discussion

This study developed a decision support system that provides fast answers to the problem of routing with heterogeneous fleets in tight time windows, whose solution time is directly proportional to the problem size. For stochastic customer demands, results are to be generated that allow evaluation appropriate to the instantaneous changes. Examining the application results, one concludes that the scores are consistent with these goals.

Annealing simulation and genetic algorithms are heuristic algorithms. Therefore, the initial solution directly affects the optimal result. In practice, the initial solutions are randomly generated depending on the constraints.

Positive changes in scores are observed as the size of the problem increases. However, the rate of increase in these positive changes decreases. This is normal for a problem that NP has already solved.

In the final solution, the linear correction process had a positive effect on the provision of constraints, but not on the number of transports with timely arrival compatibility. This situation can be addressed in the ongoing process of the study. Thus, in the final solution, an increase in the number of transports with on-time compatibility can be achieved. In this study, each condition has the same level of importance. In this study, each constraint has the same level of importance. Therefore, the constraints in the objective function are not weighted with fixed or variable factors in the second phase of the solution. The study can be reconsidered by weighting these constraints.

References

[1] Bae H. and Moon I., "Multi-Depot Vehicle Routing Problem with Time Windows Considering Delivery and Installation Vehicles," *Applied Mathematical Modelling*, vol. 40, no. 13-14, pp. 6536-6549, 2016.

[2] Çetin S., Özkütük E., and Gencer C., "Heterojen Araç Filolu Eş Zamanlı Dağıtım-Toplamalı Araç Rotalama Problemi İçin Bir Karar Destek Sistemi," *International Journal of Research and Development*, vol. 3, no. 1, pp. 11-18, 2011.

[3] Dantzig G. and Ramser J., "The Truck Dispatching Problem," *Management Science*, vol. 6, no. 1, pp. 80-91, 1959.

[4] Dinçer S. and Dişkaya F., "Yeşil Lojistik Zincirinde Araç Rotalama Problemi İçin Bir Model Önerisi," *Beykoz Akademi Dergisi*, vol. 6, no. 1, pp. 29-49, 2018.

[5] Eryavuz M. and Gencer C., "Araç Rotalama Problemine Ait Bir Uygulama," *Süleyman Demirel Üniversitesi BF Dergisi*, vol. 6, no. 1, pp. 139-155, 2001.

[6] Goldberg D., *Genetic Algorithms in Search Optimization and Machine Learning*, Addison Wesley Publishing Company,1989.

[7] Golden B., Assad A., Levy L., and Gheysens G., "The fleet size and mix vehicle routing problem," *Computers and Operations Research*, vol. 11, no.1, pp. 49-66, 1984.

[8] Grefenstette J., "Optimization of Control Parameters for Genetic Algorithms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 16, no. 1, pp. 122-128, 1986.

[9] Kirkpatrick S., Gelatt C., and Vecchi M., "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp. 671-680, 1983.

[10] Koç Ç., Bektaş T., Jabali O., and Laporte G., "Thirty Years of Heterogeneous Vehicle," *European Journal of Operational Research*, vol. 249, no. 1, pp. 1-21, 2016.

[11] Lei D. and Liu M., "An Artificial Bee Colony with Division for Distributed Unrelated Parallel Machine Scheduling with Preventive Maintenance," *Computers and Industrial Engineering*, vol. 141, no. 1-2, pp. 21-32, 2020.

[12] Leung S., Zhang Z., Zhang D., Hua X., and Lim M., "A Meta-Heuristic Algorithm for Heterogeneous Fleet Vehicle Routing Problems With Two-Dimensional Loading Constraints," *European Journal of Operational Research*, vol. 225, no. 2, pp. 199-210, 2013.

[13] Liong C., Wan I., and Omar K., "Vehicle Routing Problem: Models and Solutions," *Journal of Quality Measurement and Analysis*, vol. 4, no. 1, pp. 205-218, 2019.

[14] Miller C., Tucker A., and Zemlin R., "Integer Programming Formulation of Traveling Salesman Problems," *Journal of the ACM*, vol. 7, no. 4, pp. 326- 329, 1960.

[15] Ruey-Maw C., Yin-Mou S., and Wei-Zhi H., "Neural-Like Encoding Particle Swarm Optimization for Periodic Vehicle Routing Problems," *Expert Systems with Applications*, vol. 138, pp. 112833, 2019.

[16] Seixas M. and Mendes A., "Column Generation for a Multi-trip Vehicle Routing Problem with Time Windows Driver Work Hours and Heterogeneous Fleet," *Mathematical Problems in Engineering*, vol. 2013, no. 8, pp. 1-13, 2013.

[17] Shalaby M., Mohammed A., and Kassem S., "Supervised Fuzzy C-Means Techniques to Solve the Capacitated Vehicle Routing Problem," *The International Arab Journal of Information Technology*, vol. 18, no. 3A, pp. 452-463, 2021.

[18] Taillard E., Badeau P., Gendreau M., Guertin F., and Yves-Potvin J., "A Tabu Search Heuristic For The Vehicle Routing Problem With Soft Time Windows," *Transportation Science*, vol. 31, no. 2, pp. 170-186, 1997.

[19] Taş D., Dellaert N., Van-Woensel T., and De-Kok T., "Vehicle Routing Problem With Stochastic Travel Times İncluding Soft Time Windows and Service Costs," *Computers and Operations Research*, vol. 40, no. 1, pp. 214-224, 2013.

[20] Tirkolaee E., Goli A., Bakhsi M., and Mahdavi I., "A Robust Multi-Trip Vehicle Routing Problem of Perishable Products with İntermediate Depots and Time Windows," *Numerical Algebra Control and Optimization*, vol. 7, no. 4, pp. 417-433, 2017.

[21] Yılmaz Z., Erol S., and Aplak H., "Tehlikeli Maddelerin Taşınması-Bir Literatür Taraması," *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 22, no. 1, pp. 39-53, 2016.

Metin Bilgin received the Ph.D. degree in Computer Engineering from Yıldız Technical University in 2015. Also, he did research post-doc in the Computational Linguistic department at Uppsala University for about 10 months. He is currently an assistant professor in the Department of Computer Engineering, Bursa Uludağ University, Turkey. His current research interests include machine learning, natural language processing, and text classification.

Nisanur Bulut received MSc Degree in Computer Engineering from Bursa Uludağ University in 2021. She is computer engineer at Siemens. She is specialization in scheduling and machine planning problems.