Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

Multi-depot heterogeneous fleet vehicle routing problem with time windows: Airline and roadway integrated routing

Ömer Osman Dursun[a*] and Asuman Özger[b]

[a]*Firat University, School of Aviation, Department of Avionics, Elazig, Turkey*
[b]*Eskisehir Technical University, Faculty of Aeronautics and Astronautics, Department of Avionics, Eskisehir, Turkey*

C H R O N I C L E

Article history:
Received November 4 2021
Received in Revised Format December 21 2021
Accepted January 14 2022
Available online January, 14 2022

Keywords:
Multi-depot heterogeneous vehicle routing problem with time windows
Hybrid genetic algorithm with variable neighborhood search
Aircraft constraints
Airline cost
Air transportation

A B S T R A C T

In transportation, the multi-depot heterogeneous fleet vehicle routing problem with time windows (MDHFVRPTW) is one of the hard-to-solve real-life problems. In the study, a new node-based MDHFVRPTW has been developed. Unlike other studies in the literature, heterogeneous fleets including both airline and roadway vehicles are used for routing. In the model, real-life data of the airline and roadway are taken into consideration. In particular, important aviation constraints such as the range of the aircraft, landing and take-off cycle (LTO) cost according to the engine type, and the penalty cost are presented in the model. The problem is analysed by using narrow and wide time windows, which is the realization of fast and normal demand. A new hybrid genetic algorithm with variable neighborhood search (HGA-VNS) has been proposed for the solution of the MDHFVRPTW model. In the solution of the model, remarkable results have been obtained with the HGA-VNS algorithm compared to the genetic algorithm and off-the-shelf solvers. Also, the HGA-VNS algorithm has been tested with small and large-scale instances and compared with other studies in the literature. It is thought that the proposed MDHFVRPTW model and the developed HGA-VNS algorithm will bring a different perspective to transportation.

1. Introduction

In recent years, logistics and transportation have played a crucial key role in our lives, with increasing online shopping and e-commerce due to pandemic COVID 19. Although passenger transportation has come to a halt due to pandemics, freight transportation has continued for economic reasons. The decision-making process is very important in the logistics and cargo sectors as in every sector. Logistics companies make a short and long-term strategic plan that will increase their shares and profits in the sector and decrease their costs by making use of many factors. Common problems encountered in the decision-making in logistics are transportation type selection, fleet planning, vehicle scheduling, and vehicle routing (Pečený et al., 2020; Zhen et al., 2020). In the transportation type selection; the transportation mode is chosen to perform frequency of service, speed, cost, transport time, accessibility, and reliability. Air freighters are the safest and fastest type of freighters in the world. According to the long-term forecast of Airbus before the pandemic, air freighters will grow by about 3.6% per annum by 2038. Also, the aircraft freighter fleet is expected to increase by 55% by 2038 (Airbus Global Market Forecast, 2020). Considering the increasing trend in airline transportation, both aircraft, and road vehicles have been used in this paper as an integrated way. The purpose of fleet planning is to determine the type and number of vehicles that will meet the demand of the customers at the desired service level in a way to minimize the cost (Franceschetti et al., 2017; Karimi Dastjerd & Ertogral, 2019). The main problem to be solved in vehicle scheduling is time. It is a form of problem that investigates how many vehicles in which day and periods, from where to go (Carosi et al., 2019). Vehicle routing is a problem that determines routes with minimum cost considered customer demand and requests (Molina et al., 2020). All the above-mentioned problems

* Corresponding author
E-mail: oodursun@firat.edu.tr (Ö. O. Dursun)

2022 Growing Science Ltd.
doi: 10.5267/j.ijiec.2022.1.001

and shown in Fig. 1 such as transportation type selection, fleet planning, vehicle scheduling, and vehicle routing have been solved with the proposed model.

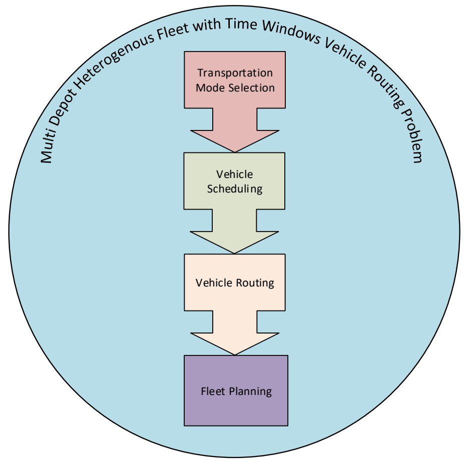

Fig. 1. Types of decision-making problems included in the model

The aim of this paper is twofold: firstly, Mixed Integer Linear Programming (MILP) formulation is developed for a node-based Multi-Depot Heterogeneous Fleet Vehicle Routing Problem with Time Windows (MDHFVRPTW).

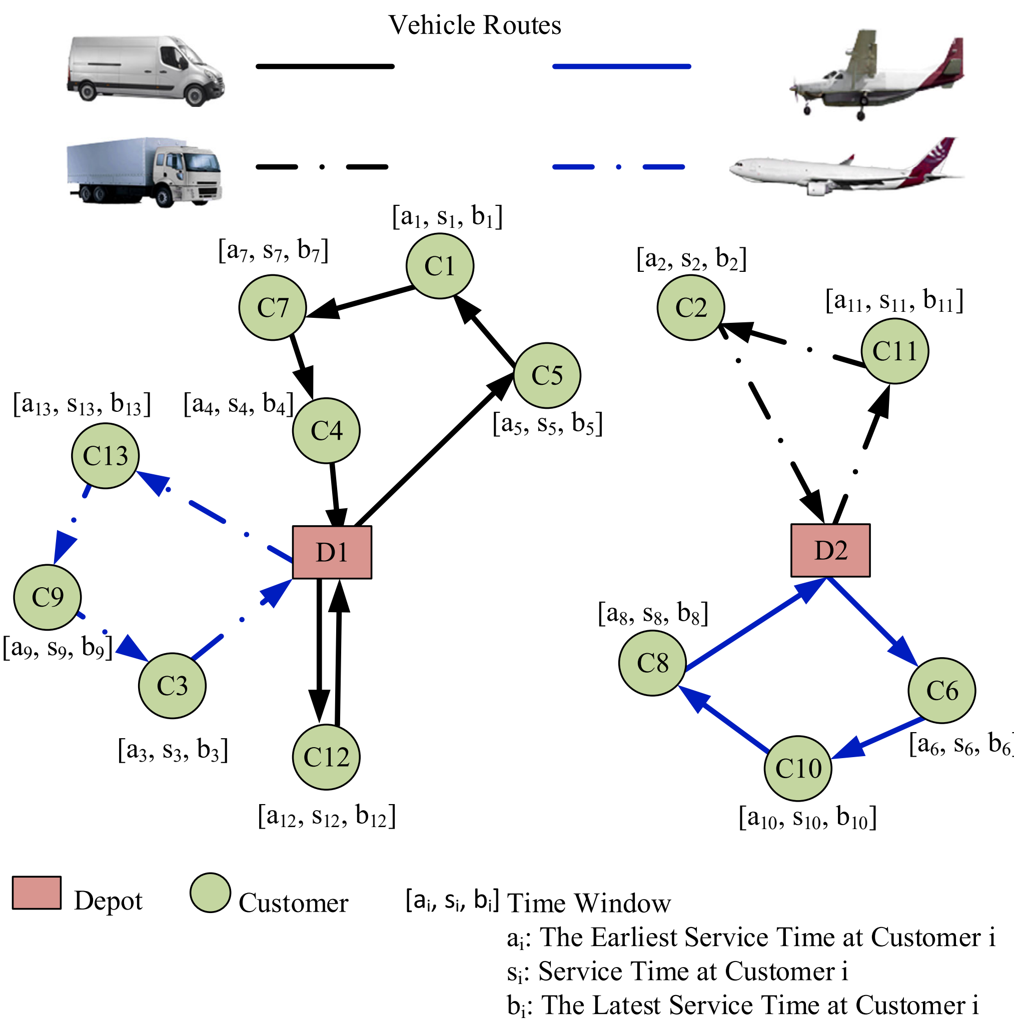

Fig. 2. The schematic representation of MDHFVRPTW

This model has not been taken into consideration in the past literature due to the heterogeneous fleet structure related to aviation. The developed model has been solved with some off-the-shelf solvers e.g., CPLEX, Intlinprog. Secondly, a hybrid genetic algorithm-based variable neighborhood search (HGA-VNS) has been developed to obtain high-quality solutions and to reduce search space. The nearest neighborhood search algorithm has been used to determine the initial population. The proposed algorithm has been tested with commonly known benchmark data. Also, the developed algorithm has been compared with other algorithms in the literature. As mentioned later, the proposed algorithm has produced an effective solution in MDHFVRPTW. A schematic representation of the problem is given in Fig. 2.

The remainder of this paper is structured as follows. In Section 2 related works in literature are reviewed. The mathematical model and its assumptions which we have developed are expressed in detail in Section 3. Also, detailed information about the parameters has been represented. Section 4 presents our HGA-VNS algorithm for solving MDHFVRPTW. Section 5 describes the results of computational experiments. Section 6 discusses the numerical results. Finally, Section 7 concludes the paper and briefly outlines our future research directions.

2. Literature review

Firstly, the vehicle routing problem (VRP) was suggested by Dantzig and Ramser to find the optimum routing of gasoline trucks (Dantzig & Ramser, 1959). After that, many developments and changes took place in the VRP. The development of the VRP was taken into account by Laporte (2009). Also, the innovations of VRP, different types and solution methods of VRP were tackled by Toth and Vigo (2014). Considering the different assumptions of the Capacitated Vehicle Routing Problem, also known as the classical, different types of vehicle routing problems were obtained. Generally, because real-life problems depend on multiple depots, different vehicle types, and time, the proposed paper is also based on MDHFVRPTW (Mancini, 2016; Montoya-Torres et al., 2015). In the MDHFVRPTW, different types of vehicles and multiple depots are used. Additionally, MDHFVRPTW is a different type of classic vehicle routing problem where vehicles leave their depots and return to the same depots after servicing customer demands within a certain time. Since this problem involves three types of vehicle routing problems based on multi-depot, heterogeneous fleet, and time windows, these vehicle routing problems are included in the literature review. The concept of heterogeneous fleet vehicle routing problem (HFVRP) was introduced for the first with Kirby's study (Kirby, 1959). HFVRP, which is used the fleet of vehicles with different capacities to service the customers for demands, is a routing problem with the minimum cost. For HFVRP, Gheysens et al. (1984) developed a mathematical method and a heuristic method. Azimi and Salari developed a mathematical method and proposed a heuristic algorithm (Naji-Azimi & Salari, 2013). Takan and Kasimbeyli developed three mathematical models for capacitated, open and split delivery heterogeneous fixed fleet vehicle routing problems (Takan & Kasimbeyli, 2021). Taillard (1999), Gendreau et al. (1999), Renaud and Boctor (2002) developed an algorithm for HFVRP. HFVRP types in the literature were examined by Baldacci, Batara, and Vigo. The proposed lower bound and heuristic algorithms for HFVRP were reviewed (Golden et al., 2008). Koç et al. conducted a study to classify examine HFVRP species. Besides this, the metaheuristic algorithms developed for HFVRP were examined and compared in the study (Koç et al., 2016). Knight and Hofer (1968) worked on the scheduling problem for a 40-vehicle company in London according to time and invocation status. Savelsbergh (1985) developed a local search algorithm for TWVRP. Solomon (1987) examined scheduling with time windows and heuristic algorithms for VRP. An algorithm based on column generation for VRPTW was developed by Desrochers et al., (1992). Taniguchi and Shimamoto (2004) developed a dynamic vehicle scheduling and routing model which carries real travel time information. Hsu et al. (2007) worked on probabilistic VRPTW for perishable food delivery. VRP with a fuzzy time windows was proposed and solved by Tang et al. (2009). VRPTW model for balanced cargo was developed by Kritikos and Ioannou (2010) and solved with a heuristic approach. A model for semi-flexible TWVRP was established by Qureshi et al. (2010) and the solution was obtained by using heuristics based on genetic algorithms in the study. A semi-heuristic tabu search algorithm was developed by Repoussis and Tarantilis (2010) for the heterogeneous and fleet-sized vehicle routing problem with time windows. Time Windows Vehicle Routing Problem (TWVRP) is an expanded type of CVRP. TWVRP, according to time windows, is divided into two as hard time windows and flexible time windows (Toth & Vigo, 2014; Zhang et al., 2019). Zhang et al. (2019) developed a hybrid ant colony optimization algorithm for the Multi-objective vehicle routing problem with flexible time windows. Laporte et al. (1988) examined multi-depot VRP and location routing problems. Renaud et al. (1996) developed a tabu search heuristic algorithm for MDVRP. Crevier et al. (2007) developed a multi-depot vehicle routing model where vehicles use intermediate storage points and carried out the solution of the model with a heuristic approach based on adaptive memory and tabu search algorithm. Ho et al. (2008) recommended a hybrid genetic algorithm for MDVRP. Within the scope of multi-depot, Aksoy and Kapanoglu (2012) carried out the solution of the problem of transporting personnel and parts to the requested bases with the existing cargo aircraft in the Turkish Air Force. Xu et al. (2012) developed a variable neighborhood search algorithm for MDHVRPTW. Salhi et al. (2014) proposed an algorithm based on the formulation and a variable neighborhood search algorithm for the multi-depot heterogeneous fleet vehicle routing problem (MDHFVRP). Bae and Moon (2016) developed MDVRPTW mixed-integer programming model for routing delivery and set up tools. For the solution of the model, they developed a heuristic method based on the nearest neighbor and genetic algorithm. Li et al. (2018) developed a mathematical model for MDVRP, where resources are not shared, and MDVRP, where resources are shared, and compared the fuel consumption of the two models. Bezerra et al. (2018) proposed a general neighborhood search metaheuristic algorithm. Wang et al. (2020) developed a model for collaborative multi-depot vehicle routing problems with time windows- (CMDVRPTW). They implemented the solution of the model with the hybrid heuristic algorithm including K-means clustering, saving algorithm, and Extended Non-dominated Sorting Genetic Algorithm-II. Zhen et al. (2020) developed a formulation (MIP formulation) for the problem of multi-depot and multi-turn time windows vehicle routing. They also proposed a hybrid particle swarm optimization algorithm (HPSO) and a hybrid genetic algorithm.

3. Problem Formulation

In this section, the MILP formulation for MDHFVRPTW developed in this study is presented. Also, detailed information is given about the parameters of the proposed model.

3.1. A mixed-integer linear formulation for MDHFVRPTW

Let $G(N, A)$ be a completed directed network where $N= 1,\dots,m+n$ refers to all nodes including potential depots and costumers, and A stands for paths between these nodes. $N= 1,\dots,m+n$ in the set of nodes; $N_0= 1, \dots, m$ refers to m number of depo nodes $N_c= m+1, \dots, m+n$ refers to n number of customer nodes. $N = N_0 \cup N_C$

Assumptions

- Each depot nodes are known.
- Each customer node is known.
- Airports are depots and customer nodes.
- The demand of each customer node is specific and cannot be split.
- The vehicle fleet is known and the vehicles are located at the depot nodes.
- The company was used its resource and investment costs were ignored.
- Speed of vehicles: Average speed in road vehicles, cruise speed at 35000 feet (ft) altitude specified by the European Air Navigation Safety Organization (Eurocontrol) for aircrafts Airbus and Boeing, cruise speed at an altitude of 25000 ft for Cessna based on.

Indices and sets

i, j: set of all nodes $i, j = 1, \dots, m + n$
k, h: set of customers nodes $k, h = 1, \dots, m$
t: the set of vehicle types $t = 1, \dots, p + r$
In the set of vehicle type; $1, \dots, p$ indicates road vehicles, $p+1, \dots, p+r$ indicates airway vehicles.

Parameters
c_{ijt}: the distance from node i to j when traveling with vehicle type t
H_{ijt}: the traveling time between node i and j when traveling with vehicle type t
F_{tv}: the fixed cost of vehicle v of type t
L_{tv}: fuel cost of vehicle v of type t during take-off, climb, and landing for aircraft, it denoted as Landing-Take Off (LTO) cost
α_t: the unit transportation cost of the vehicle type t
β_{kt}: penalty cost for the empty capacity of the vehicle type t in depot k (for aircrafts)
A_t: the number of vehicles type t
B_t: capacity ratio parameter of vehicle type t (1/ Q_t)
Q_t: the vehicle capacity of type t
d_j: the demand for customer j
R_{tv}: the range of vehicle v of type t (Since there is no range limitation for road vehicles, roadway distances were examined and a value larger than the maximum distance value between two points was used as the range value in the matrix.)
l_i: the service time of vehicle at customer i
a_i: the earliest service time at customer i
b_i: the latest service time at customer i
M: large positive number
n: customer number

Decision variables

$$x_{ijtv} = \begin{cases} 1, & if\ a\ vehicle\ v\ of\ type\ t\ travels\ from\ node\ i\ to\ j \\ 0, & otherwise \end{cases}$$

$$z_{kj} = \begin{cases} 1, & if\ customer\ j\ is\ assigned\ to\ depot\ k \\ 0, & otherwise \end{cases}$$
s_{ktv}: empty capacity percent of the vehicle v of the type t departured from depot k
$0 \leq s_{ktv} \leq 1$

u_i: decision variable used to prevent subtours

$0 \leq u_i$

w_{itv}: the starting service time of vehicle v of type t at customer i

$0 \leq w_{itv}$

Objective function

$$Minimize\ f = \sum_{k=1}^{m}\sum_{j=m+1}^{m+n}\sum_{t=1}^{p+r}\sum_{v=1}^{A_t} F_{tv}\, x_{kjtv} + \sum_{i=1}^{m+n}\sum_{\substack{j=1\\ j\neq i}}^{m+n}\sum_{t=p+1}^{p+r}\sum_{v=1}^{A_t} L_{tv}\, x_{ijtv} + \sum_{k=1}^{m}\sum_{t=p+1}^{p+r}\sum_{v=1}^{A_t} \beta_{kt}\, s_{ktv} + \sum_{i=1}^{m+n}\sum_{\substack{j=1\\ j\neq i}}^{m+n}\sum_{t=1}^{p+r}\sum_{v=1}^{A_t} \alpha_t\, c_{ijt} x_{ijtv} \tag{1}$$

Constraints

$$\sum_{k=1}^{m} z_{kj} = 1 \qquad j = m+1, \dots, m+n \tag{2}$$

$$\sum_{i\neq j}^{m+n}\sum_{t=1}^{p+r}\sum_{v=1}^{A_t} x_{ijtv} = 1 \qquad j = m+1, \dots, m+n \tag{3}$$

$$\sum_{k=1}^{m}\sum_{j=m+1}^{m+n} x_{kjtv} \leq 1 \qquad \forall t, \forall v \tag{4}$$

$$\sum_{j\neq i}^{m+n} x_{ijtv} \leq \sum_{k=1}^{m}\sum_{j=m+1}^{m+n} x_{kjtv} \qquad i = m+1, \dots, m+n;\ \forall t, \forall v \tag{5}$$

$$\sum_{k=1}^{m}\sum_{j=m+1}^{m+n} x_{kjt(v+1)} \leq \sum_{k=1}^{m}\sum_{j=m+1}^{m+n} x_{kjtv} \qquad \forall t, \forall v = 1, \dots, (A_t - 1) \tag{6}$$

$$\sum_{i=1}^{m+n} x_{ijtv} - \sum_{i=1}^{m+n} x_{jitv} = 0 \qquad \forall j, \forall t, \forall v \tag{7}$$

$$\sum_{t=1}^{p+r}\sum_{v=1}^{A_t} x_{kjtv} \leq z_{kj} \qquad \forall k;\ j = m+1, \dots, m+n \tag{8}$$

$$\sum_{t=1}^{p+r}\sum_{v=1}^{A_t} x_{jktv} \leq z_{kj} \qquad \forall k;\ j = m+1, \dots, m+n \tag{9}$$

$$\sum_{t=1}^{p+r}\sum_{v=1}^{A_t} x_{ijtv} + z_{ki} + \sum_{h\neq k}^{m} z_{hj} \leq 2 \qquad \forall k; i, j = m+1, \dots, m+n;\ i \neq j \tag{10}$$

$$\sum_{i=1}^{m+n}\sum_{j=m+1}^{m+n} d_j\, x_{ijtv} \leq Q_t \qquad \forall t, \forall v \tag{11}$$

$$c_{ijt} x_{ijtv} \leq R_{tv} \qquad \forall i, \forall j, i \neq j,\ t = p+1, \dots, p+r; \forall v \tag{12}$$

$$\sum_{j=m+1}^{m+n} x_{kjtv} - B_t \sum_{i=1}^{m+n} \sum_{\substack{j=m+1 \\ j \neq i}}^{m+n} d_j \, x_{ijtv} \leq s_{ktv} \qquad \forall k;\ t = p+1, \dots, p+r; \forall v \tag{13}$$

$$w_{itv} + \left(l_i + H_{ijt}\right) x_{ijtv} - w_{jtv} \leq (1 - x_{ijtv}) M \qquad \forall i;\ j = m+1, \dots, m+n,\ i \neq j, \forall t, \forall v \tag{14}$$

$$w_{jtv} + (l_j + H_{jkt} - b_k) x_{jktv} \leq (1 - x_{jktv}) M \qquad \forall k;\ j = m+1, \dots, m+n,\ \forall t, \forall v \tag{15}$$

$$a_i \sum_{\substack{j=m+1 \\ j \neq i}}^{m+n} x_{ijtv} \leq w_{itv} \qquad \forall i, \forall t, \forall v \tag{16}$$

$$w_{itv} \leq b_i \sum_{j \neq i}^{m+n} x_{jitv} \qquad \forall i, \forall t, \forall v \tag{17}$$

$$u_i - u_j + n \sum_{t=1}^{p+r} \sum_{v=1}^{A_t} x_{ijtv} \leq n - 1 \qquad i, j = m+1, \dots, m+n, i \neq j \tag{18}$$

In this formulation, the objective function *f (s, x)* (1) minimizes the total fixed cost, the total LTO costs for the aircraft, the total penalty costs for aircraft, and the total routing costs. Constraints (2) allows each customer node to be assigned to a depot node. Constraints (3) ensure that each customer must be visited exactly once. Constraints (3) allow that each customer node must be visited once. Constraints (4) ensures the vehicle *v* of type *t* to be used on one route at most. It is an important constraint in terms of vehicle route tracking. Constraints (5), if the vehicle *v* of type *t* has not left any depot and is not assigned to a route, it ensures that it cannot be used at customer nodes. Constraints (6), without the use of the vehicle *v* of type *t,* ensure that the vehicle *v + 1* is not used. In other words, it ensures that the vehicle with the last index is not used without using the vehicle with the previous index of a type. Constraints (7), are the vehicle flow constraints. If any vehicle reaches a customer node and serves, they ensure that it leaves with the same vehicle. Constraints (8) - (10) prevent illegal routes that do not start and end in the same depot nodes. They ensure that the route starts and ends in the same depot nodes. These provide the relationship between *x* decision variable and *z* decision variable. Firstly constraints were applied by Labbé et al. (2004) in the location routing problem. Later, they were developed by Karaoglan et al. (2012). In this study, these constraints were modified and strengthened for use in MDHFVRPTW problems. Constraints (11) are the vehicle capacity constraints. These constraints ensure that the load of the vehicle will carry along its route does not exceed the vehicle's capacity. Constraints (12) are the range constraints for aircraft. Constraints (13) determine the amount of empty capacity of the aircraft. Constraints (12) and (13) have been proposed firstly in this study. Constraints (14) and (15) are time flow constraints. Constraints (16) and (17) allow each customer to be served in its time windows. Constraints (14 - 17) were suggested as time constraints by Toth and Vigo (2014). These constraints have been developed in this study for MDHFVRPTW. Constraints (18) are subtour elimination constraints. These were developed by Miller et al. (1960).

3.2. Model Parameters

Since airway and roadway vehicles are used together in this study, it is necessary to know the costs of both transportation types. Especially airline costs differ according to the roadway. Airline costs are divided into direct and indirect operating costs. Direct operating cost; these are the costs that vary depending on the aircraft type and flight. The direct operating cost includes all flight expenses (flight crew, wages, fuel), maintenance - repair, and depreciation of the aircraft. Costs that are independent of the aircraft type and use of the aircraft are considered indirect operating costs. Indirect operating costs include passenger-related passenger service costs, ticketing, and sales costs, station and ground service costs, and general administrative costs. The sum of both costs gives the fixed cost for the aircraft (Doganis, 1991).

When various companies engaged in air freight transportation in the world are examined, it is seen that Airbus A300, A330, or Boeing 747, 757, 767 type cargo aircraft are used. When the companies such as FedEx and UPS in America, where aviation is more developed than other countries, are examined, it is seen that MD11 type aircraft are also used. Besides, companies such as FedEx and DHL use small aircraft such as Fokker and Cessna as well as large aircraft in regional freight transportation (DHL, 2020; FedEx, 2020; Lufthansa, 2020; THY, 2020; UPS, 2020).

In the proposed study, Airbus A330-200F and Boeing 747-400ERF, which are used extensively in freight transport as large wide-body aircraft types, are taken into consideration. Also, Cessna Grand Caravan Ex aircraft are used, especially in regional flights and in cases where the load is low, due to the low purchase cost. The proposed mathematical model could be expressed as three indices. But it was considered as four indices. The same aircraft having different engine options is one of the most important reasons for this. For instance, there are different engine options as CF6-80E1A2, PW-4164 or Trent 772 in the Airbus A330-200F aircraft (Airbus, 2020). If a different engine type is used, the fuel cost will change. For this reason, the same aircraft should be considered as a different aircraft type when the engine types are different. The characteristics of the aircraft considered are given in Table 1 (Airbus-A330, 2020; Boeing-747, 2020; Cessna-GrandCaravan, 2020).

Table 1
Aircraft types and specifications.

Specifications	Airbus A330-200F	Boeing 747-400ERF	Cessna GrandCaravan Ex
Maximum range(*km*)	7400	9230	1689
Maximum cruise speed	35 000 ft altitude	35 000 ft altitude	25 000 ft altitude
(*km/h*)	876	896	343
(*kts*)	473	484	185
Maximum take-off weight (*tonnes*)	233	412	4
Maximum payload (tonnes)	65	112	1.4
Maximum fuel capacity (*litres*)	97 530	204 350	752
Engine type	CF6-80E1A2	PW 4062	PT6A-140

The idle, take-off-climb, and approach flight phases of the aircraft are called the LTO cycle. The times in the LTO cycle are obtained from the ICAO document and are given in Table 2. The cost of fuel consumption during the LTO cycle is considered as the LTO cost. LTO cost is a fixed cost type that occurs when the aircraft is assigned to the route. LTO cost varies according to the type of engine but does not vary with the distance flown. The total amount of fuel consumed by the aircraft was determined by adding the amount of fuel burned during the LTO cycle and the amount of fuel burned during the cruise. The amount of fuel consumption during the LTO cycle was calculated using the ICAO document, and the amount of fuel consumption during cruise flight was calculated using Eurocontrol's BADA document. When the time elapsed in the LTO cycle shown in Table 2 is subtracted from the total flight time, cruise flight time was found. Fuel cost was calculated for each flight phase, taking into account the relevant times, speed, and fuel consumption. LTO fuel data was given related to engine type by ICAO, while the cruise phase flight fuel data was given related to aircraft type by Eurocontrol BADA (ICAO, 2020; Nuic, 2004).

Table 2
LTO cycle time

LTO Time (min)	
Take-Off	0.7
Climb-Out	2.2
Approach	4
Idle	26
Total Time	**32.9**

Vehicle types used for the roadway are van, light truck, medium truck, and heavy truck. The costs of the vehicle types used in road freight transportation are the fixed cost of the vehicle, the cost of the driver, the cost of an assistant, and the cost of fuel. On the other hand, in the study, the fixed cost of the vehicle, the cost of the driver, and the cost of the assistant were considered within the total fixed cost. The fixed cost for both types of transport is known as the cost that occurs depending on the vehicle's assignment. It is expressed as stated in Eq. (19) and (20). In this study, real operational data are used as fixed cost data (Dursun, 2017; EKOL, 2020; Oktal & Ozger, 2013). Also, by assuming that the aircraft fly according to instrument flight rules, real distance data of the airline and roadway were used (GoogleMaps, 2020; RocketRoute, 2020).

For aircraft;

$$F_{tv} = Direct\ operating\ cost + Indirect\ operating\ cost \tag{19}$$

For roadway vehicle;

$$F_{tv} = Fixed\ cost + Driver\ cost + Assistant\ cost \tag{20}$$

In the proposed study, penalty cost (β_{kt}) that was handled in the heterogeneous vehicle routing problem by Gheysens et al. (1984) and the penalty approaches used in flexible time windows vehicle routing problem was adapted to the MDHFVRPTW (Balakrishnan, 1993; Solomon & Desrosiers, 1988; É. Taillard, Badeau, Gendreau, Guertin, & Potvin, 1997; Taş, Jabali, & Van Woensel, 2014; Yan, Chu, Hsiao, & Huang, 2015; Zare-Reisabadi & Hamid Mirmohammadi, 2015). The set of customer nodes that may belong to the depots was heuristically considered when calculating the β_{kt}. Routes were calculated with the savings algorithm, a heuristic algorithm proposed by Clarke-Wright, between the set of customer nodes and the depots associated with customer nodes (Clarke & Wright, 1964). The sum of routing cost, fixed cost, and LTO cost, which arise when an airplane with a full load capacity visits all customer nodes, gives the penalty cost of the airplane type for that depot node. The penalty cost should be considered together with the constraints (13). If the aircraft is not assigned to the route, the S_{ktv} decision variable will take the value zero. If it is assigned to a route, S_{ktv} will change between zero and 1, depending on the rate of load it carries. In other words, if the load has filled its capacity, S_{ktv} will be zero, if not, S_{ktv} will take a value between zero and 1 according to the occupancy rate. The penalty cost is shown mathematically in Eq. (21).

$$\beta_{kt} = \sum_{j=m+1}^{m+n}\sum_{v=1}^{A_t} F_{tv}\, x_{kjtv} + \sum_{i=1}^{m+n}\sum_{\substack{j=1 \\ j\neq i}}^{m+n}\sum_{v=1}^{A_t} L_{tv}\, x_{ijtv} + \sum_{i=1}^{m+n}\sum_{\substack{j=1 \\ j\neq i}}^{m+n}\sum_{v=1}^{A_t} \alpha_t\, c_{ijt} x_{ijtv} \qquad \forall k, \forall t \qquad (21)$$

4. Algorithmic approach

In this section, we developed an HGA-VNS algorithm to solve our proposed mathematical model. GA is one of the metaheuristic methods that has been widely used to solve NP-hard problems. Although GAs have effective global search and excellent convergence, their weak local search capacity and efficiency in the last iteration are low (Baniamerian, Bashiri, & Tavakkoli-Moghaddam, 2019; Zhen et al., 2020). Therefore, a new state-of-the-art HGA-VNS algorithm has been developed that can perform effective local searches for global solutions.

4.1. HGA-VNS algorithm

The proposed algorithm consists of some special algorithmic approaches. These are initial population, roulette wheel selection, Partial Mapped Crossover (PMX), swap mutation, VNS for local search. Roulette wheel selection, PMX, swap mutation is often based on traditional GA (Bezerra et al., 2018; Gen, Cheng, & Lin, 2008; Pečený et al., 2020). It also forms the basis of our algorithm. Although these operators belonging to GA are known, they differ depending on the developed mathematical model. The same is valid for the shaking procedure in the VNS algorithm. The general structure of HGA-VNS is summarized in Algorithm 1.

Algorithm 1. (*HGA-VNS framework*)

Subroutine separate subtour
Subroutine ensure all constraints (assign vehicle, demand constraints, time constraints, special aviation constraints)
Subroutine calculate fitness function
Subroutine improve fitness function (VNS)

Initialize population (first parent generated with NNS) # Section 4.2
separate subtour (all parents chromosome in populations)
ensure all constraints
calculate fitness function # Section 4.3
improve fitness function (VNS)

while the termination condition is not met
generate new generations from population (roulette wheel selection) # Section 4.4
select randomly two parents p_1 and p_2 from the population
create two child c_1 and c_2 from two parents p_1 and p_2 by crossover routine # Section 4.5
create two new offspring o_1 and o_2 from two child c_1 and c_2 by mutation routine # Section 4.6
separate subtour (o_1 and o_2)
ensure all constraints (o_1 and o_2)
calculate fitness function (o_1 and o_2)
improve fitness function (o_1 and o_2) # Section 4.7
if fitness function is improved
    delete p_1 and p_2 from the population
    add o_1 and o_2 to the population
endif
end while

4.2. Initial population construction

The initial population has a significant effect on the GA search space. So first, to create the first population efficiently, we assigned the customers to depots. Next, NNS is used to create the first individual of the population. The NNS framework is shown in Algorithm 2. Thus, individuals with different chromosome lengths and populations as many as the number of depots are obtained. Other individuals are also produced by randomly shuffling the chromosomes of the first individual. Initial parent and randomly generated parents are shown in Fig.3.

Algorithm 2. (*NNS framework*)

```
while the termination condition is not met
if the first individual is not empty
        for genes of the first individual
                select the minimum distance (among all genes including depot gene)
delete gene with minimum distance from the first individual
add gene with minimum distance to new individual
                end for
        endif
        else
                break

end while
```

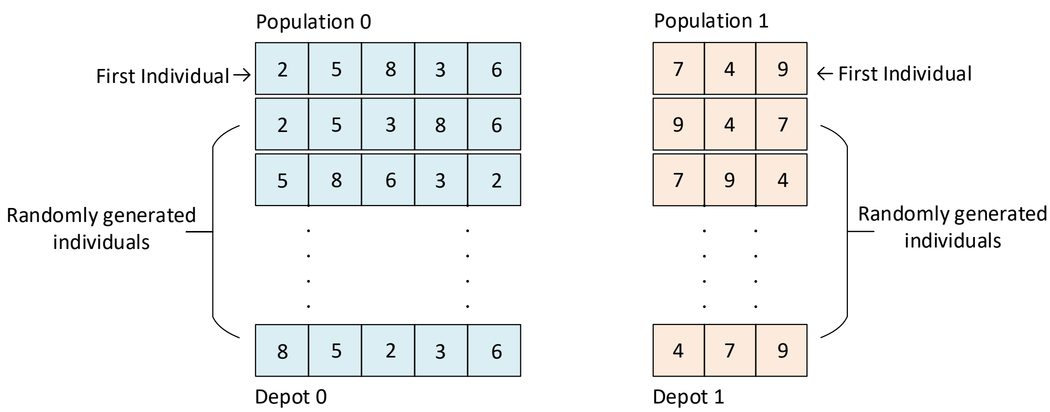

Fig. 3. Structure of the initial populations

4.3. Evaluation of fitness function

Since the proposed model includes both aircraft and road vehicles, the objective function has been developed for both types of transport. So, due to the structure and the characteristic of the model the fitness function used for the algorithm is evaluated as shown in Eq. (1).

4.4. Selection method

The roulette wheel selection based on the probabilistic selection of individuals related to their fitness function is applied in the developed algorithm for selection methodology. In the selection procedure first, new generations are generated from the populations. The fitness function of the individuals in the populations related to the depots is calculated according to the equation in Eq. (1). Since the model is a minimization problem, the inverse of the fitness function $1 / f(x)$ is taken. The selection probability of each individual is obtained with the inverse of the fitness function. The cumulative probabilities of individuals are also taken into account. Each individual is placed on the roulette wheel to the selection probabilities percentage of individuals. Therefore, individuals with a high selection probability are more likely to be selected. The roulette wheel is rotated to the number of generations and the selection of individuals is made randomly. At each turn of the roulette wheel, an individual is added to the new generation. This selection is made for each population. Thus, new generations are generated from populations. The selection procedure and steps are shown in Fig. 4 (Gen et al., 2008; Tasan & Gen, 2012). Nodes 0 and 1, represent the depots. For the elite to survive, two parents with the maximum fitness function are selected for the crossover operator.

4.5. Crossover operator

One of the main genetic operators is a crossover in which two different parent chromosomes are crossed to reproduce the new offspring. The new offspring don't have completely different characteristics from their parents. Some characteristics of parents are passed on to them. So, new characteristics are not obtained with the crossover operator. In the study PMX which one of the most proposed is used for crossover operators (Gen et al., 2008). (see Table 3).

Table 3
PXM crossover operator.

PMX Procedure	
Input	Two Parent
Step 1	Select two positions at random called mapping sections
Step 2	Exchange the genes of both parents in mapping sections
Step 3	Determine mapping relationship between parents genes
Step 4	Obtain offsprings
Output	Two Offsprings

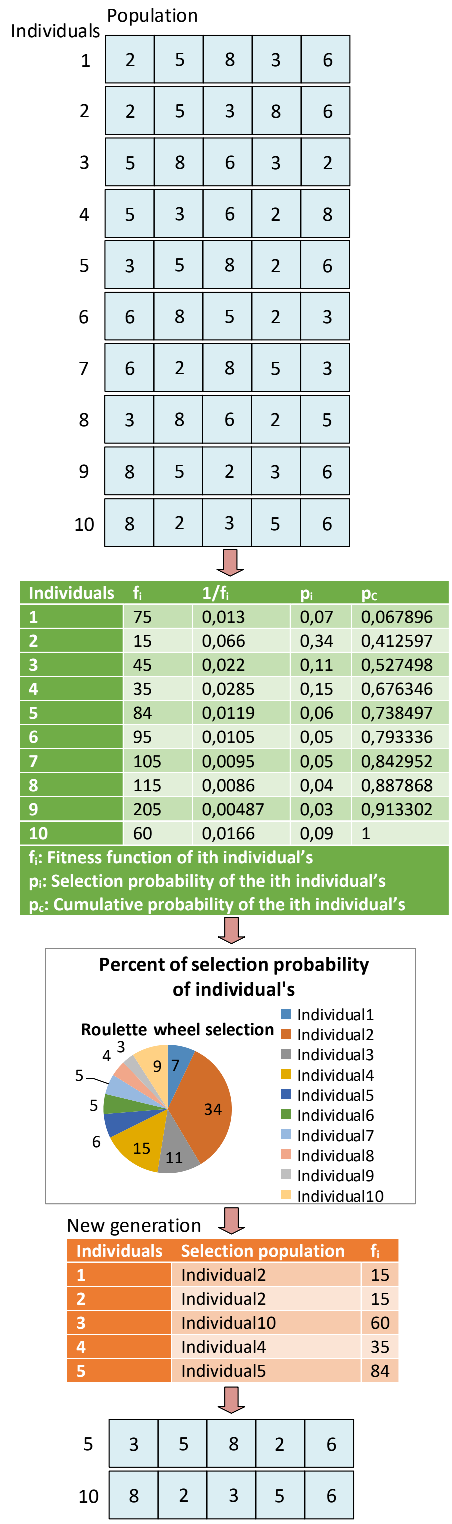

Fig.4. Selection method steps

4.6. Mutation operator

The mutation is the other genetic operator. A completely new individual is created in the mutation. Small changes are made in the solution with the mutation operator. With the random changes made, new characteristics are gradually added to the population. New characteristics cannot be added to the population by the crossover operator. Swap mutation was used in the study. Unlike other studies, the mutation rate was determined in swap mutation. The determined mutation rate was multiplied by the chromosome length of the individual. Thus, the number of genes to be swapped was obtained. If the number of genes is odd, it is increased by one and completed to an even number. This is because the exchange is only possible between two genes. Genes were selected randomly as much as the resulting even number and were changed randomly. The swap mutation procedure is shown in Table 4 (Gen et al., 2008).

Table 4
Swap mutation operator

Swap Mutation Procedure	
Input	Parent
Step 1	Select the number of genes by mutation rate multiplied by chromosome length
Step 2	Check the number of genes, if it is odd, increase it by one and obtain an even number
Step 3	Swap two genes randomly selected, repeat it during gene length
Step 4	Obtain offspring
Output	Offspring

4.7. VNS algorithm

VNS used to search the solution space is an algorithm based on local search. VNS obtains better solutions by improving the search space with different neighborhood structures. It offers effective solutions in combinatorial optimization problems such as traveling salesman, vehicle routing, and scheduling (Dong, Zhang, Xu, & Shen, 2021). Effective solutions are provided with the help of the shaking procedure (Baniamerian et al., 2019). Diversity is increased by investigating different neighborhood structures with the shaking procedure. By making more local searches, the proposed algorithm converges to the global optimum solution. In this study, a three-step shaking procedure is proposed. Within VNS, iteration can be done in two ways. One of them is the termination of the iteration if there is an improvement in the fitness function. Another is the investigation of neighborhoods as much as the given iteration value. A different iteration method was applied in the proposed study. Since there is no neighborhood for genes with a single element, the VNS was terminated without any action in the algorithm. For subtour or number of the genes, greater than one and less than six, the number of iterations was determined by taking the factorial of the number of the gene. Factorial is used because the cost matrix of the problem consists of both road vehicles and airway vehicles and at the same time this matrix is not symmetrical. If the cost matrix were symmetrical, it would be better to take half of the factorial as an iteration. A certain number of iterations is taken for subtour with six and greater genes. In the first stage of the VNS algorithm, the swap is applied to each subtour of the parent that is divided into subtour. Considering the gene length of the subtour, the swap procedure was applied both randomly selected as a single gene or as a group with half the number of genes. In the second stage, the reversion procedure was applied. In reversion, a random gene sequence is selected in the subtour and inserted in reverse order. In the last stage, the insertion process was applied. During the insertion process, a randomly selected gene was taken and placed in a randomly determined location. The best objective function was obtained by calculating the fitness function at each stage by the number of iterations. One of the differences of the proposed VNS algorithm from the basic and the previous studies is that the swap, reversion, and insertion procedures are applied to the subtour obtained from the parent. The shaking structure steps of the VNS algorithm in this study are shown in Fig. 5.

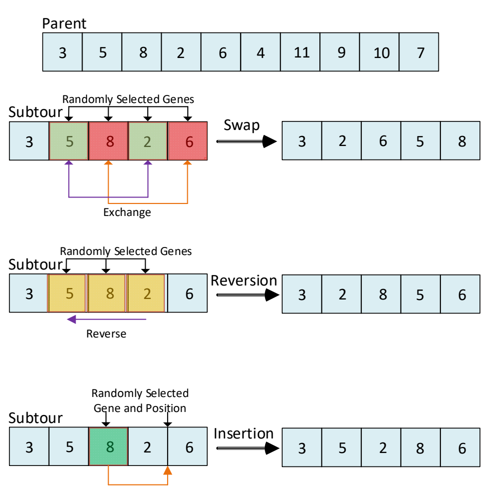

Fig. 5. Shaking structure of VNS

5. Experimental Results

In this section, numerical results are presented for the proposed MDHFVRPTW. The problem expressed as narrow time windows and large time windows has been addressed in two ways namely fast demand fulfillment and normal demand fulfillment. At the same time, the problem addressed is expressed as a hard time window according to the time window. In other words, customer demands must be fulfilled within the specified time windows. Experimental results of real data have been calculated using a computer with Intel Core I7 (TM) -8750H CPU and 2.20 GHz 16 GB RAM. While calculations were made, the real data, which were small in size, were first solved using The General Algebraic Modelling Language (GAMS) with the CPLEX solver and MATLAB with the Intlinprog solver. It was seen that the developed model could not reach the global optimum even with small data using off-the-shelf solvers. For this reason, a new HGA-VNS algorithm was developed in Python for the solution of the model. The problem was solved by using the proposed HGA-VNS algorithm and genetic algorithm. Both algorithms were compared numerically. At the same time, the developed HGA-VNS algorithm was tested with large and small size test samples using the instances of Cordeau (Cordeau et al., 2001). Also, the proposed HGA-VNS algorithm was compared with Bae and Moon's heuristic and GA (Bae and Moon, 2016).

5.1. Fast demand fulfillment

In the proposed study, a total of 24 nodes are considered, including 12 nodes from Europe and 12 from Turkey. Istanbul Sabiha Gokcen Airport in Turkey and Bonn / Koln Airport in Europe were selected from these nodes as the depots. The remaining airports were expressed as customer nodes. Node IDs 1 and 2 express the depots and the others represent customers. Since all nodes occur from airports, they are expressed together with the airport's codes determined by ICAO. The nodes and their ICAO codes are shown in Table 5.

Table 5
Depot and customer nodes.

Node ID	ICAO Code	Depots and Customers (Airports)
1	LTFJ	Istanbul Sabiha Gokcen
2	EDDK	Bonn/Koln
3	LTAC	Ankara
4	LTAI	Antalya
5	LTBD	Aydin
6	LTFD	Balikesir
7	LTBR	Bursa
8	LTAY	Denizli
9	LTCA	Elazig
10	LTBJ	İzmir
11	LTAU	Kayseri
12	LTBV	Mugla
13	LTAR	Sivas
14	LTBU	Tekirdag/Corlu
15	EDDT	Berlin
16	EDDW	Bremen
17	EDLW	Dortmund
18	EDDF	Frankfurt
19	EDNY	Friedrichshafen
20	EDQM	Hof/Plauen
21	EDSB	Karlsruhe
22	ELLX	Luxembourg
23	EDDM	Munich
24	LIRN	Napoli

In the developed mathematical model, both roadway and airline vehicles were used as heterogeneous vehicle fleets. It was assumed that each vehicle type consists of a fleet of 10 vehicles. The vehicles and their capacities used in MDHFVRPTW are presented in Table 6 (Airbus, 2020; Boeing-747, 2020; Cessna-GrandCaravan, 2020; EKOL, 2020).

Table 6
Vehicle types and capacities.

Vehicle ID	Vehicle Type	Capacity (kg)
1	Van	1000
2	Light truck	1300
3	Medium truck	3400
4	Heavy truck	13000
5	Cessna Grand Caravan Ex	1600
6	Airbus A330-200F	65000
7	Boeing 747-400ERF	112000

Vehicles used in roadway transportation offer cheaper transportation in terms of cost compared to aircraft used in airline transportation. In the study, the costs of the trucks used for the roadway and the aircraft used for the airline are shown in Table 7 and Table 8 (EKOL, 2020; ICAO, 2020; Nuic, 2004; Oktal & Ozger, 2013).

Table 7
Roadway vehicle costs

Vehicle Type	Fixed cost ($/day)	Fuel cost ($/km)	Driver cost ($/day)	Assistant cost ($/day)
Van	25	0.2	27	21
Light truck	24	0.1	27	21
Medium truck	48	0.2	27	21
Heavy truck	29	0.3	27	21

Table 8
Airline costs

Aircraft Type	Transport cost ($/km)	LTO cost ($)
Cessna Grand Caravan Ex	0.5	31
Airbus A330-200F	2.6	888
Boeing 747-400ERF	4.2	843

Roadway transportation is insufficient for the loads requiring rapid transportation such as food, medicine, and vaccines, especially during the pandemic period. In such cases, airline transportation, which is the fastest and most reliable type of transportation regardless of the cost, comes to the fore. In the study, it was assumed that some customer points had fast demands. The customers who have a fast freight demand are given in Table 9.

Table 9
Customers with fast demand freights.

Node ID	ICAO Code	Customers
4	LTAI	Antalya
9	LTCA	Elazig
14	LTBU	Tekirdag/Corlu
15	EDDT	Berlin
16	EDDW	Bremen
24	LIRN	Napoli

The narrow time windows [6, 192] were applied for the customers who have a fast demand. In the specified time windows, the demands of these customers must be met within 6 hours by a vehicle. Also, the vehicle serving the customer must return to its depot within 192 hours. The time windows [96, 192] were considered for the customers who have no fast demand. The service time for each customer was assumed as one hour. The depots, customers who have a fast demand or not, demand quantities and the latest service times are shown in Table10.

Table 10
Demands and latest service time.

Node ID	ICAO Code	Depots/Customers	Demand (kg)	Latest Service Time (h)
1	LTFJ	İstanbul Sabiha Gokcen	0	192
2	EDDK	Bonn/Koln	0	192
3	LTAC	Ankara	727	96
9	LTCA	Elazig	2	6
10	LTBJ	Izmir	47	96
17	EDLW	Dortmund	3810	96
18	EDDF	Frankfurt	8447	96
21	EDSB	Karlsruhe	2808	96
24	LIRN	Napoli	781	6

Developed MDHFVRPTW to fulfill fast demand was solved using some off-the-shelf solvers CPLEX and Intlinprog. The best objective function value was found as 23458.54 at the end of 9000 seconds with the CPLEX. Although a result was obtained with a CPLEX solver in solving the problem, no result was found with Intlingprog solver for fast demand fulfillment (see Fig. 6). When only the x decision variable x_{ijtv} (i=24, j=24, t=7, v=10) was considered in the problem, the number of decision variables x was found to be 40320 (24 × 24 × 7 × 10). Considering the constraint numbers of the problem and the number of other decision variables, it could be seen how complex and NP-hard the problem was. Therefore, the problem was solved by using GA which is one of the metaheuristic methods, and the newly developed HGA-VNS algorithm. Solution results are shown in Table 11. Customers who have a fast demand are stated in bold fonts in Table 11. When the first route of the CPLEX algorithm in Table 11 was considered, freight distribution was carried out on the Istanbul Sabiha Gokcen - Tekirdag/Corlu - Mugla - Izmir - Balikesir - İstanbul Sabiha Gokcen route by light trucks.

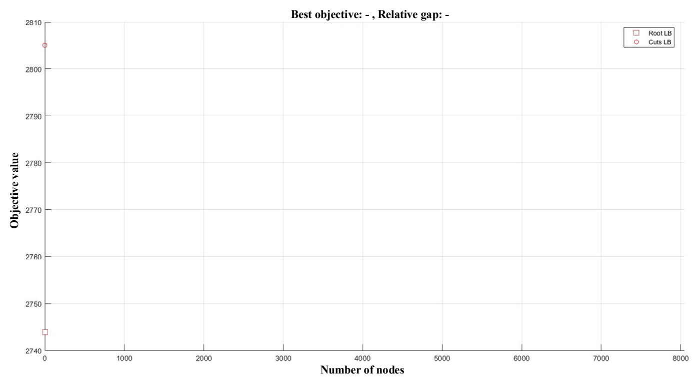

Fig. 6. Fast demand fulfillment results of the Intlinprog solver

Table 11
Fast demand fulfillment performance of the solvers and algorithms

Solver/Algorithm	Routes (Node ID)	Vehicle type (Vehicle ID)	Vehicle number	Objective function ($)	Time (seconds)
CPLEX	1 - **14** -5 -12 - 10 - 6 - 1	2	1	23458.54	9000
	2 - 17 - **16** - 23 - 21 - 2	4	1		
	1 - **4** - **9** - 13 - 11 - 3 - 8 - 7 - 1	5	1		
	2 - 19 - **24** - 22 - 2	5	2		
	2 - **15** - 20 - 18 - 2	6	1		
Intlinprog	-	-	-	-	-
GA	1 - **14** - 5 - 10 - 12 - 13 - 11 - 1	2	1	23974.83	469
	1 - 6 - 1	2	2		
	1 - 8 - **4** - 1	4	1		
	2 - **16** - 22 - 17 - 23 - 21 - 2	4	2		
	1 - **9** - 3 - 7 - 1	5	1		
	2 - 20 - 2	5	2		
	2 - **24** - 19 - 2	5	3		
	2 - 18 - **15** - 2	6	1		
HGA-VNS	1 - 7 - 6 - 5 - 12 - 8 - 11 - 1	2	1	19776.62	8724
	1 - **14** - 3 - 1	2	2		
	2 - **16** - 17 - 20 - 23 - 19 - 21 - 2	4	1		
	1 - 10 - **9** - 1	5	1		
	1 - 13 - **4** - 1	5	2		
	2 - 22 - **24** - 2	5	3		
	2 - 18 - **15** - 2	6	1		

The developed MDHFVRPTW was solved with GA one of the metaheuristic algorithms. The best objective function value was found as 23974.83 in 469 seconds solution time in GA. Detailed information about the solution is given in Table 11. Also, the change with time of the average fitness function obtained from the solution of the problem with GA is shown in Fig. 7 (a), and the change of the best fitness function with time is shown in Fig. 7 (b).

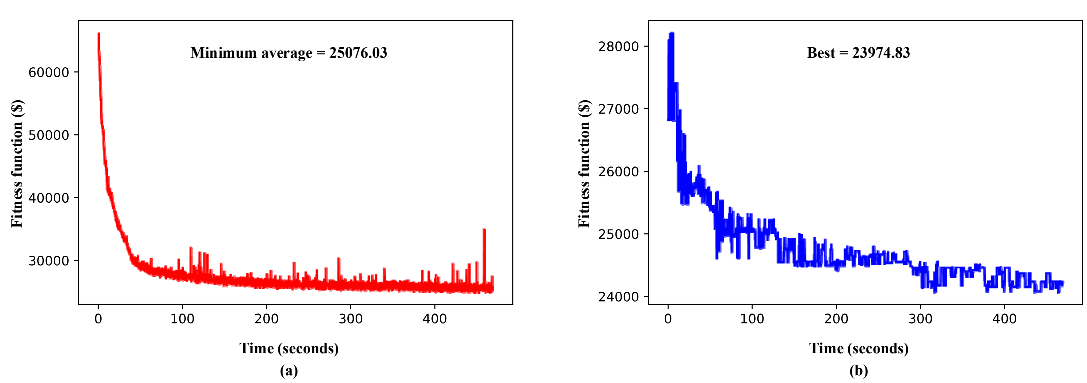

Fig. 7. **(a)** The change of the average fitness function of the GA with time for the fast demand fulfillment, **(b)** The change of the best fitness function of the GA with time for the fast demand fulfillment

The problem was solved by using the newly developed HGA-VNS algorithm to fulfill the fast demand and effective results were found. With the HGA-VNS algorithm, the best objective function was obtained as 19776.62 in 8724 seconds solution time. In Fig. 8 (a), the change of the average fitness function of the HGA-VNS algorithm with time, and in Fig. 8 (b) the change of the best fitness function of the HGA-VNS algorithm with time are given.

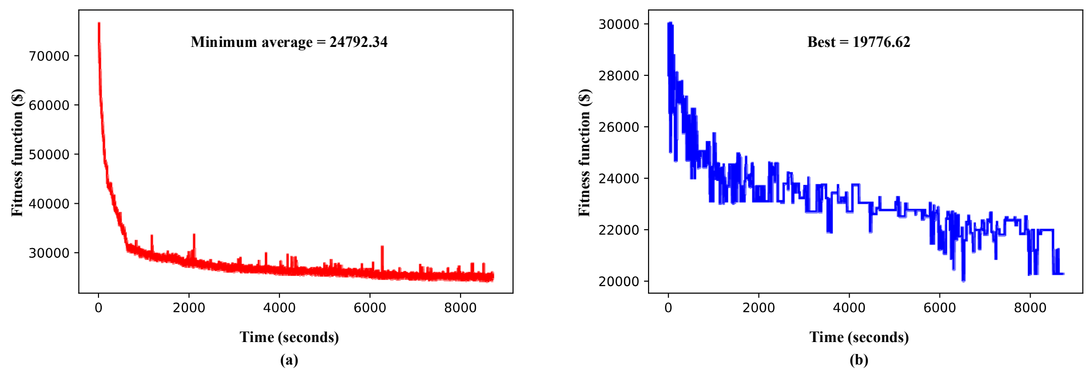

Fig. 8. (a) The change of the average fitness function of the HGA-VNS with time for the fast demand fulfillment, **(b)** The change of the best fitness function of the HGA-VNS with time for the fast demand fulfillment

As seen in Table 11, the best objective function was obtained with the HGA-VNS algorithm. The routes that were found in the HGA-VNS algorithm to fulfill the fast demand were shown in Fig. 9.

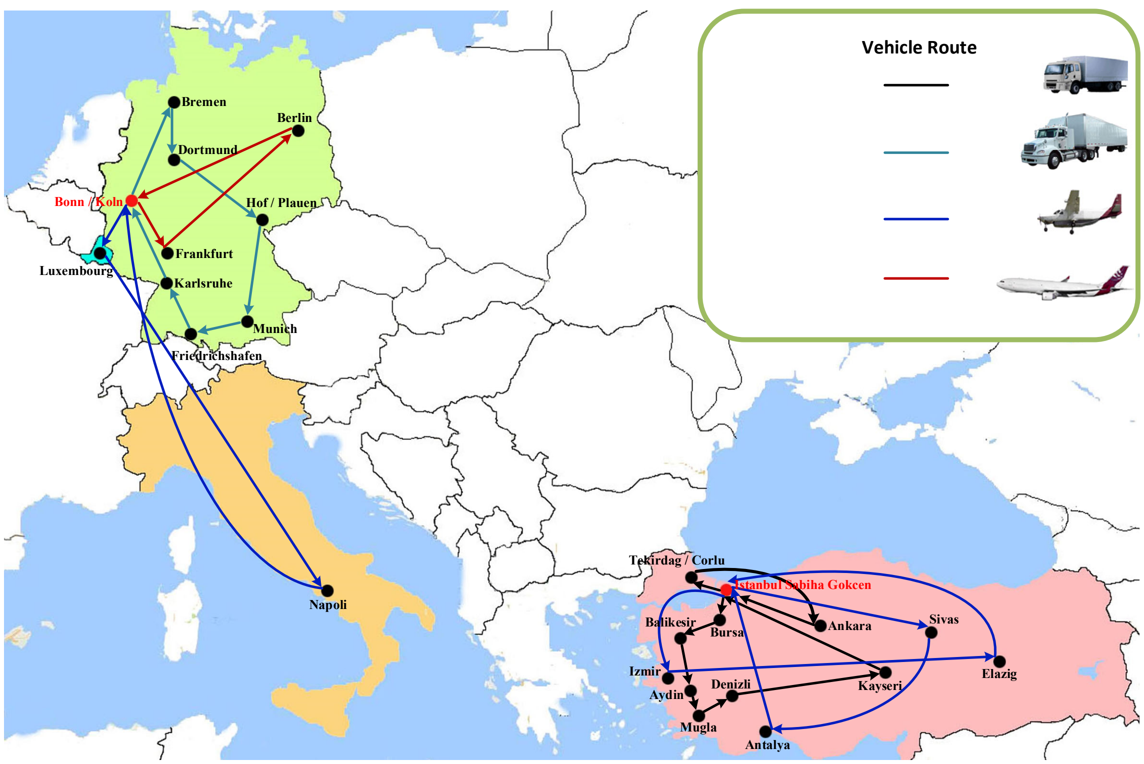

Fig. 9. The routes found by the HGA-VNS algorithm for the fast demand fulfillment

5.2. Normal (Non-fast) demand fulfillment

The large time windows [96, 196] were used to fulfill the normal (non-fast) demands of the customers. It was assumed that each customer was served in the 96 hours. Vehicles must return to their depots where they start their route in the 196 hours. The difference between normal and fast demand fulfillment is the use of a large time windows. It was observed that all solvers and algorithms achieve a result when customer demands were fulfilled using a large time windows. When the problem was solved with CPLEX, the best objective function value in 1000 seconds solution time was calculated as 2131.4. The best objective function value was obtained as 3576.3 in 14580 seconds solution time with the Intlinprog solver. The solution graph of the Intlinprog solver is shown in Fig. 10.

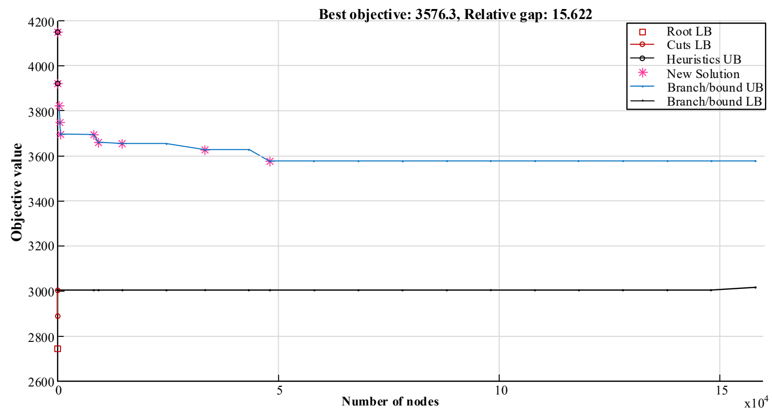

Fig. 10. Normal demand fulfillment results of Intlinprog solver

When the first route of the Intlinprog solver shown in Table 12 was examined, it was seen that the light truck started its route from the Bonn/Koln depot and returned to the Bonn/Koln depot after serving Napoli and Luxembourg customers. The details about the solution to the problem are given in Table 12.

Table 12
Normal demand fulfillment performance of the solvers and algorithms.

Solver Algorithm	Routes (Node ID)	Vehicle type (Vehicle ID)	Vehicle number	Objective function ($)	Time (seconds)
CPLEX	1 - 14 - 6 - 10 - 5 - 12 - 4 - 8 - 7 - 1	2	1	2131.4	1000
	2 - 22 - 24 - 2	2	2		
	1 - 3 - 11 - 9 - 13 - 1	2	3		
	2 - 20 - 23 - 19 - 21 - 2	4	1		
	2 - 15 - 16 - 17 - 2	4	2		
	2 - 18 - 2	4	3		
Intlinprog	2 - 24 - 22 - 2	2	1	3576.3	14580
	1 - 3 - 11 - 13 - 9 - 4 - 8 - 5 -12 - 10 - 6 - 1	2	2		
	1 - 14 - 7 - 1	2	3		
	2 - 20 - 23 - 19 - 21 - 2	4	1		
	2 - 16 - 15 - 17 - 2	4	2		
	2 - 18 - 2	4	3		
GA	1 - 10 - 5 - 12 - 8 - 7 - 1	2	1	2141.6	3234
	1 - 3 - 11 - 13 - 9 - 4 - 6 - 14 - 1	2	2		
	2 - 22 - 24 - 15 - 16 - 17 - 2	4	1		
	2 - 20 - 23 - 19 - 21 - 2	4	2		
	2 - 18 - 2	4	3		
HGA-VNS	1 - 7 - 6 - 10 - 8 - 4 - 12 - 5 - 14 - 1	2	1	2068.2	197
	1 - 9 - 13 - 11 - 3 – 1	2	2		
	2 - 24 - 2	2	3		
	2 - 17 - 20 - 21 - 22 - 2	4	1		
	2 - 18 - 2	4	2		
	2 -19 - 23 - 15 - 16 - 2	4	3		

As in the narrow time windows, the problem was solved using GA. When the problem was solved with GA, the best objective function value was found as 2141.6 in 3234 seconds solution time. GA graphs to the fulfillment of the normal demand are shown in Fig. 11. By using the large time windows, the change of the average fitness function of the GA algorithm with time in Fig. 11 (a), and the change of the best fitness function of the GA algorithm with time in Fig. 11 (b) are given.

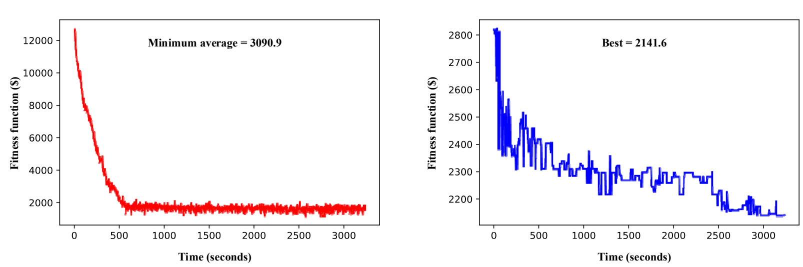

Fig. 11. (a) The change of the average fitness function of GA with time for the normal demand fulfillment, **(b)** The change of the best fitness function of GA with time for the normal demand fulfillment

The problem was successfully solved using the large time windows with the proposed HGA-VNS algorithm. The best objective function value was found as 2068.2 in 197 seconds solution time with the HGA-VNS algorithm. HGA-VNS algorithm graphs obtained using large time windows are seen in Fig. 12. The change of the average fitness function of the HGA-VNS algorithm with time for the normal demand fulfillment is shown in Fig. 12 (a), and the change of the best fitness function of the HGA-VNS algorithm with time for the normal demand fulfillment is shown in Fig. 12 (b).

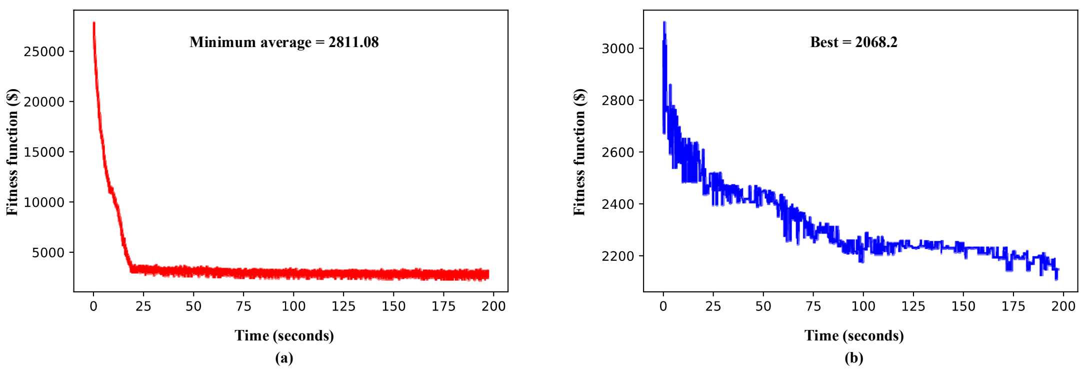

Fig. 12. (a) The change with time of the average fitness function in HGA-VNS for the normal demand fulfillment, **(b)** The change with time of the best fitness function in HGA-VNS for the normal demand fulfillment

As seen in Table 12, the best objective function was obtained with the HGA-VNS algorithm for the normal demand fulfillment. The routes that were found in the HGA-VNS algorithm to fulfill the normal demand were shown in Fig. 13.

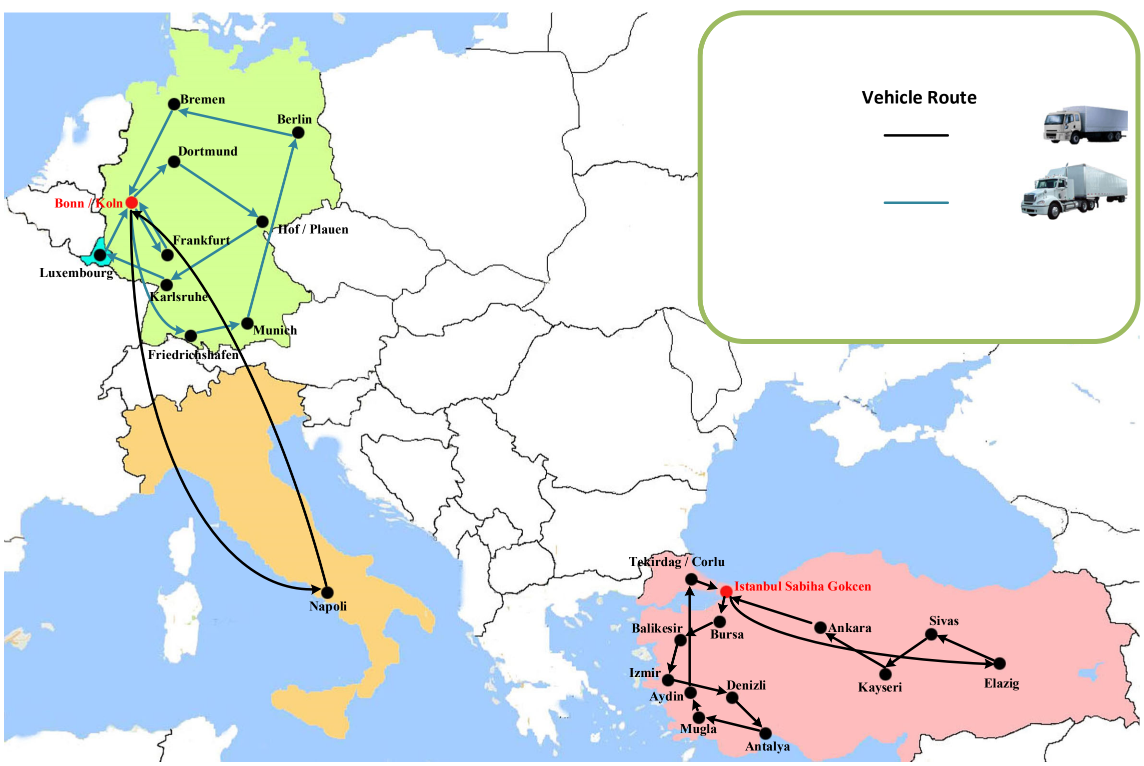

Fig. 13. The routes found by the HGA-VNS algorithm for the normal demand fulfillment

5.3. Parameters selection of GA and HGA-VNS

The selection of population size, generation size, mutation rate, and crossing rate parameters for both GA and HGA-VNS algorithms were performed sensitively. The population size and generation size were considered as 50, 100, 150. It is quite difficult to evaluate all tuning parameters in terms of performance. In the proposed study was preferred mutation rate and crossover rate parameters frequently used in the literature. The predefined parameter values were applied as 0.003 for mutation rate and 0.9 for crossover (Hassanat et al., 2019; Yıldırım & Kuvvetli, 2021). The parameter settings with the best objective function are shown in Table 13.

Table 13
Parameters of algorithms.

Algorithm	Population size	Generation size	Mutation rate	Crossover rate
GA	100	50	0.03	0.9
HGA-VNS	150	50	0.03	0.9

5.4. Comparison of algorithms

The proposed algorithm was compared with previous algorithms using instances developed by Cordeau et al. (2001). Cordeau et al.'s instances consist of two groups as (a) and (b). The group (a) consists of a narrow time window by selecting the uniform random e_i in the intervals [60, 480] and then the uniform random l_i in the interval $[e_i + 90, e_i + 180]$. The group (b) consists of a large time window by selecting the uniform random e_i in the intervals [60, 300] and then the uniform random l_i in the interval $[e_i + 180, e_i + 360]$. The instances of Cordeau et al. are presented in Table 14. As seen in Table 14, m refers to the number of vehicles, n number of customers, t number of depots, D maximal route duration, Q vehicle capacity.

Table 14
Cordeau's instances characteristics

No.	*m*	*n*	*t*	*D*	*Q*	*No.*	*m*	*n*	*t*	*D*	*Q*
1a	2	48	4	500	200	1b	1	48	4	500	200
2a	3	96	4	480	195	2b	2	96	4	480	195
3a	4	144	4	460	190	3b	3	144	4	460	190
4a	5	192	4	440	185	4b	4	192	4	440	185
5a	6	240	4	420	180	5b	5	240	4	420	180
6a	7	288	4	400	175	6b	6	288	4	400	175
7a	2	76	6	500	200	7b	1	72	6	500	200
8a	3	144	6	475	190	8b	2	144	6	475	190
9a	4	216	6	450	180	9b	3	216	6	450	180
10a	5	288	6	425	170	10b	4	288	6	425	170

Cordeau et al.'s instances include one type of vehicle that is homogeneous. The mathematical model proposed in the study includes heterogeneous vehicle types, as well as different types of transportation. For this reason, a new model has been obtained according to the homogeneous vehicle type by modifying the proposed mathematical model to make a comparison with the algorithm proposed by Bae and Moon (2016). The model is considered as two indices. Besides, all constraints except assignment, time, vehicle capacity, depot, and subtour elimination constraints are ignored. Bae and Moon compared their Heuristic algorithm with the genetic algorithm, taking into account the fixed costs of the vehicle's departure from the depots. However, in this study, Bae and Moon's heuristic and genetic algorithm, and proposed HGA-VNS algorithm were compared according to the distance. As a result of the comparison, it is seen that the proposed HGA-VNS algorithm gives better results than the results obtained from previous algorithms compared to both groups in terms of distance. Comparison results are shown in Table 15.

Table 15
Comparison results

No.	Bae et al. Heuristic		Bae et al. Genetic algorithm		Proposed HGA-VNS algorithm	
	CPU Time (seconds)	Distance	CPU Time (seconds)	Distance	CPU Time (seconds)	Distance
1a	13	2225.4	180	2691.0	180	2053.4
2a	25	3759.6	180	3885.0	180	3441.3
3a	37	6186.5	180	6600.6	180	5873.7
4a	49	8265.5	180	9821.4	180	7869.0
5a	60	8401.4	180	8996.5	180	8223.1
6a	71	8898.3	180	9654.4	180	8557.0
7a	20	3156.1	180	3248.3	180	3046.5
8a	38	6234.3	180	6651.1	180	6115.8
9a	55	7225.0	180	8305.7	180	6874.5
10a	72	10659.7	180	11232.9	180	10126.3
1b	14	1854.0	180	1904.6	180	1732.5
2b	25	3061.9	180	3061.9	180	2556.8
3b	38	5222.0	180	5195.1	180	4862.0
4b	50	6236.3	180	6236.3	180	5883.6
5b	61	5989.4	180	5989.4	180	5697.8
6b	71	7729.7	180	7729.7	180	7554.9
7b	20	2638.6	180	3097.6	180	2441.5
8b	38	4635.6	180	4635.6	180	4156.8
9b	55	5511.0	180	6149.2	180	5216.7
10b	73	7559.3	180	7559.3	180	7148.6

6. Discussion

The node-based MDHFVRPTW model, which includes a heterogeneous fleet of airlines and roadways, was developed. New constraints on aviation were presented in the model. The developed model was solved using real data. The model was analysed in two scenarios as narrow time windows and large time windows. Firstly, it was ensured that customers who were determined by using narrow time windows were fulfilled their demands quickly. The problem was solved with the basic solvers CPLEX, Intlinprog solvers while fulfilling the fast demand. While CPLEX, one of the solvers, obtained the objective function value of 23458.54 at the end of 9000 seconds, Intlinprog could not find any solution. Therefore, the problem was solved with the metaheuristic methods GA and the developed HGA-VNS algorithm. The objective function value of 23974.83 was found in 469 seconds with GA. The objective function value was obtained in 8724 seconds as 19776.62 using the HGA-VNS algorithm. Considering the solution time, it was seen that GA provides a fast solution. However, considering the objective function, the HGA-VNS algorithm gave a better result than both CPLEX and GA. Also, because the time was crucial for distribution while the fast demand was fulfilled the distribution was made to long distances by aircraft. If the customer who needs fast demand was close to the depot, the distribution was made by roadway vehicle. If it was in a remote location, the distribution carried out with small or wide-body aircraft types, taking into account the amount of demand. The distribution was fulfilled to customers with low demand by small aircraft, and to customers with a large demand by wide-body aircraft type. It is seen in detail in Table 11. In the other scenario, the demands of the customers were fulfilled by using the large time windows. In the second scenario, the problem was evaluated with CPLEX, Intlinprog solvers, GA, and HGA-VNS algorithms. In this scenario, a solution was obtained with all of them, including Intlinprog. The objective function value 2131.4 with the CPLEX solver in 1000 seconds solution time, the objective function value 3576.3 with the Intlinprog solver in 14580 seconds solution time, the objective function value 2141.6 with the GA in 3234 seconds solution time and the HGA-VNS objective function value was obtained as 2068.2 in 197 seconds solution time. When the second scenario was evaluated in terms of both the solution time and the best objective function, the superiority of the HGA-VNS algorithm was seen. In this scenario where the large time windows were used, only roadway vehicles were used. Because there was no time limit and also the airline was costly, the aircraft wasn't used in the normal demand fulfillment. The HGA-VNS algorithm, which was evaluated in two scenarios with a dataset based on real data, was also examined with test data developed by Cordeau. The HGA-VNS algorithm examined with Cordeau instances was compared in terms of distance with the heuristic and GA proposed by Bae and Moon. As seen in Table 15, it obtained better results than both algorithms. The significant contributions of this study are as follows:

- Node-based MILP model MDHFVRPTW is developed.
- Different aircraft fleets and road vehicles are considered for the heterogeneous fleet.
- New constraints and costs are achieved for aviation, for instance, range constraint, penalty cost, and real unit transportation cost depending on aircraft type, landing and taking off cost varying to aircraft engine type.
- Real aviation flight path data considering instrument flight rules and real highway distance data are used.
- The HGA-VNS algorithm is proposed for the solution of MDHFVRPTW.

The limitation of the study can be considered as not using the other transportation types such as railway and maritime transport.

7. Conclusions

In this study q node-based, mixed-integer linear programming model and a new hybrid genetic algorithm with variable neighborhood search for its solution were proposed for MDHFVRPTW. Especially, airline and roadway integrated heterogeneous fleet was used for this model. New constraints were presented for airline transportation and these were examined in detail. Hard time windows that the arrival times of delivery of customers must not exceed the maximum allowable time interval were evaluated in terms of time. The results of the computational experiments also showed the applicability and effectiveness of the developed HGA-VNS algorithm. MDHFVRPTW model and the HGA-VNS algorithm used in its solution are thought to play an effective role in freight transportation for logistics and different types of transport. Considering other modes of transport such as rail and maritime transport, re-evaluation of the constraints of the mathematical model according to the type of transport used should also be taken into account. In future studies, different transportation types and their constraints will also be considered.

Acknowledgments

This research was supported by Anadolu University within the scope of a scientific research project in cooperation with EKOL Logistics Company (1406F322).

References

Airbus-A330. (2020). Airbus. Retrieved August 30, 2020, from https://www.airbus.com/aircraft/freighter/a330-200f.html#details

Airbus. (2020). Airbus A330 Characteristics. Retrieved February 15, 2021, from https://www.airbus.com/content/dam/corporate-

topics/publications/backgrounders/techdata/aircraft_characteristics/Airbus-Commercial-Aircraft-AC-A330.pdf
Airbus Global Market Forecast. (2020). Global Market Forecast. Retrieved August 9, 2020, from https://www.airbus.com/aircraft/market/global-market-forecast.html
Aksoy, Ö., & Kapanoglu, M. (2012). Multi-Commodity , Multi-Depot , Heterogenous Vehicle Pickup and Delivery Problem for Air Transportation in the Turkish Air Force. *Journal of Aeronautics and Space Technologies*, *5*(4), 53–57.
Bae, H., & Moon, I. (2016). Multi-depot vehicle routing problem with time windows considering delivery and installation vehicles. *Applied Mathematical Modelling*, *40*, 6536–6549. https://doi.org/10.1016/j.apm.2016.01.059
Balakrishnan, N. (1993). Simple Heuristics for the Vehicle Routeing Problem with Soft Time Windows. *The Journal of the Operational Research Society*, *44*(3), 279–287.
Baniamerian, A., Bashiri, M., & Tavakkoli-Moghaddam, R. (2019). Modified variable neighborhood search and genetic algorithm for profitable heterogeneous vehicle routing problem with cross-docking. *Applied Soft Computing Journal*, *75*, 441–460. https://doi.org/10.1016/j.asoc.2018.11.029
Bezerra, S. N., Souza, S. R. De, & Souza, M. J. F. (2018). A GVNS Algorithm for Solving the Multi-Depot Vehicle Routing Problem. *Electronic Notes in Discrete Mathematics*, *66*, 167–174. https://doi.org/10.1016/j.endm.2018.03.022
Boeing-747. (2020). Boeing. Retrieved August 30, 2020, from http://www.boeing.com/resources/boeingdotcom/company/about_bca/startup/pdf/freighters/747-400f.pdf
Carosi, S., Frangioni, A., Galli, L., Girardi, L., & Vallese, G. (2019). A matheuristic for integrated timetabling and vehicle scheduling. *Transportation Research Part B: Methodological*, *127*, 99–124. https://doi.org/10.1016/j.trb.2019.07.004
Cessna-GrandCaravan. (2020). Cessna. Retrieved August 30, 2020, from https://cessna.txtav.com/en/turboprop/grand-caravan-ex#_model-specs
Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research*, *12*(4), 568–581. https://doi.org/10.1287/opre.12.4.568
Cordeau, J. F., Laporte, G., & Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, *52*(8), 928–936. https://doi.org/10.1057/palgrave.jors.2601163
Crevier, B., Cordeau, J. F., & Laporte, G. (2007). The multi-depot vehicle routing problem with inter-depot routes. *European Journal of Operational Research*, *176*(2), 756–773. https://doi.org/10.1016/j.ejor.2005.08.015
Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Management Science*, *6*(1), 80–92.
Desrochers, M., Desrosiers, J., & Solomon, M. (1992). A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows. *Operations Research*, *40*(2), 342–354. https://doi.org/10.1287/opre.40.2.342
DHL. (2020). DHL. Retrieved August 30, 2020, from https://aviationcargo.dhl.com/fleet-information
Doganis, R. (1991). *Flying Off Course The Economics of International Airlines* (2nd ed.). New York, USA: Routledge.
Dong, X., Zhang, H., Xu, M., & Shen, F. (2021). Hybrid genetic algorithm with variable neighborhood search for multi-scale multiple bottleneck traveling salesmen problem. *Future Generation Computer Systems*, *114*, 229–242. https://doi.org/10.1016/j.future.2020.07.008
Dursun, Ö. O. (2017). *Mathematical Model Suggestion For A Vehicle Routing Problem With The Fleet Of Air And Roadway Vehicles*. Anadolu University.
EKOL. (2020). Ekol. Retrieved August 30, 2020, from http://www.ekol.com/tr
FedEx. (2020). Fedex. Retrieved August 30, 2020, from http://www.fedex.com/us/charters/airplanes.html
Franceschetti, A., Honhon, D., Laporte, G., Woensel, T. Van, & Fransoo, J. C. (2017). Strategic fleet planning for city logistics. *Transportation Research Part B: Methodological*, *95*, 19–40. https://doi.org/10.1016/j.trb.2016.10.005
Gen, M., Cheng, R., & Lin, L. (2008). *Network Models and Optimization: Multiobjective Genetic Algorithm Approach*. Springer-Varlag.
Gendreau, M., Laporte, G., Musaraganyi, C., & Taillard, É. D. (1999). A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Computers & Operations Research*, *26*(12), 1153–1173. https://doi.org/10.1016/S0305-0548(98)00100-2
Gheysens, F., Golden, B., & Assad, A. (1984). A Comparison of Techniques for Solving the Fleet Size and Mix Vehicle Routing Problem. *OR Spektrum*, *6*, 207–216.
Golden, B., Raghavan, S., & Wasil, E. (2008). The Vehicle Routing Problem: Latest Advances and New Challenges. In B. Golden, S. Raghavan, & E. Wasil (Eds.), *Information Systems Journal*. https://doi.org/10.1007/978-0-387-77778-8
GoogleMaps. (2020). Google Map. Retrieved August 30, 2020, from https://www.google.com/maps/@?dg=dbrw&newdg=1
Hassanat, A., Almohammadi, K., Alkafaween, E., Abunawas, E., Hammouri, A., & Prasath, V. B. S. (2019). Choosing mutation and crossover ratios for genetic algorithms-a review with a new dynamic approach. *Information (Switzerland)*, *10*(12). https://doi.org/10.3390/info10120390
Ho, W., Ho, G. T. S., Ji, P., & Lau, H. C. W. (2008). A hybrid genetic algorithm for the multi-depot vehicle routing problem. *Engineering Applications of Artificial Intelligence*, *21*, 548–557. https://doi.org/10.1016/j.engappai.2007.06.001
Hsu, C. I., Hung, S. F., & Li, H. C. (2007). Vehicle routing problem with time-windows for perishable food delivery. *Journal of Food Engineering*, *80*(2), 465–475. https://doi.org/10.1016/j.jfoodeng.2006.05.029
ICAO. (2020). EASA. Retrieved August 30, 2020, from https://www.easa.europa.eu/document-library/icao-aircraft-engine-emissions-databank#1
Karaoglan, I., Altiparmak, F., Kara, I., & Dengiz, B. (2012). The location-routing problem with simultaneous pickup and delivery: Formulations and a heuristic approach. *Omega*, *40*(4), 465–477. https://doi.org/10.1016/j.omega.2011.09.002
Karimi Dastjerd, N., & Ertogral, K. (2019). A fix-and-optimize heuristic for the integrated fleet sizing and replenishment

planning problem with predetermined delivery frequencies. *Computers and Industrial Engineering*, *127*(September 2018), 778–787. https://doi.org/10.1016/j.cie.2018.11.014

Kirby, D. (1959). Is Your Fleet the Right Size? *Operational Research Society*, *10*(4), 252. https://doi.org/10.1055/s-0035-1549902

Knight, K. W., & Hofer, J. P. (1968). Vehicle Scheduling with Timed and Connected Calls : A Case Study. *Operational Research Society*, *19*(3), 299–310.

Koç, Ç., Bektaş, T., Jabali, O., & Laporte, G. (2016). Thirty years of heterogeneous vehicle routing. *European Journal of Operational Research*, *249*(1), 1–21. https://doi.org/10.1016/j.ejor.2015.07.020

Kritikos, M. N., & Ioannou, G. (2010). The balanced cargo vehicle routing problem with time windows. *International Journal of Production Economics*, *123*(1), 42–51. https://doi.org/10.1016/j.ijpe.2009.07.006

Labbé, M., Rodríguez-Martin, I., & Salazar-González, J. J. (2004). A branch-and-cut algorithm for the plant-cycle location problem. *Journal of the Operational Research Society*, *55*(5), 513–520. https://doi.org/10.1057/palgrave.jors.2601692

Laporte, G. (2009). Fifty Years of Vehicle Routing. *Transportation Science*, *43*(4), 408–416. https://doi.org/10.1287/trsc.1090.0301

Laporte, G., Nobert, Y., & Taillefer, S. (1988). Solving a Family of Multi-Depot Vehicle Routing and Location-Routing Problems. *Transportation Science*, *22*(3), 161–172.

Li, J., Wang, R., Li, T., Lu, Z., & Pardalos, P. M. (2018). Bene fi t analysis of shared depot resources for multi-depot vehicle routing problem with fuel consumption. *Transportation Research Part D*, *59*(February), 417–432. https://doi.org/10.1016/j.trd.2018.01.026

Lufthansa. (2020). Lufthansa. Retrieved August 30, 2020, from https://lufthansa-cargo.com/fleet-ulds/fleet

Mancini, S. (2016). A real-life Multi Depot Multi Period Vehicle Routing Problem with a Heterogeneous Fleet: Formulation and Adaptive Large Neighborhood Search based Matheuristic. *Transportation Research Part C: Emerging Technologies*, *70*, 100–112. https://doi.org/10.1016/j.trc.2015.06.016

Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM*, *7*(4), 326–329.

Molina, J. C., Salmeron, J. L., Eguia, I., & Racero, J. (2020). The heterogeneous vehicle routing problem with time windows and a limited number of resources. *Engineering Applications of Artificial Intelligence*, *94*(February), 103745. https://doi.org/10.1016/j.engappai.2020.103745

Montoya-Torres, J. R., López Franco, J., Nieto Isaza, S., Felizzola Jiménez, H., & Herazo-Padilla, N. (2015). A literature review on the vehicle routing problem with multiple depots. *Computers and Industrial Engineering*, *79*, 115–129. https://doi.org/10.1016/j.cie.2014.10.029

Naji-Azimi, Z., & Salari, M. (2013). A complementary tool to enhance the effectiveness of existing methods for heterogeneous fixed fleet vehicle routing problem. *Applied Mathematical Modelling*, *37*(6), 4316–4324. https://doi.org/10.1016/j.apm.2012.09.027

Nuic, A. (2004). *Aircraft Performance Summary Tables for The Base of Aircraft Data (BADA) Revision 3.6*. Cedex, France:Eurocontrol Experimental Centre.

Oktal, H., & Ozger, A. (2013). Hub location in air cargo transportation: A case study. *Journal of Air Transport Management*, *27*, 1–4. https://doi.org/10.1016/j.jairtraman.2012.10.009

Pečený, L., Meško, P., Kampf, R., & Gašparík, J. (2020). Optimisation in Transport and Logistic Processes. *Transportation Research Procedia*, *44*(2019), 15–22. https://doi.org/10.1016/j.trpro.2020.02.003

Qureshi, A. G., Taniguchi, E., & Yamada, T. (2010). Exact solution for the vehicle routing problem with semi soft time windows and its application. *Procedia - Social and Behavioral Sciences*, *2*(3), 5931–5943. https://doi.org/10.1016/j.sbspro.2010.04.008

Renaud, J., & Boctor, F. F. (2002). A sweep-based algorithm for the fleet size and mix vehicle routing problem. *European Journal of Operational Research*, *140*, 618–628. https://doi.org/10.1016/S0377-2217(01)00237-5

Renaud, J., Laporte, G., & Boctor, F. F. (1996). A tabu search heuristic for the multi-depot vehicle routing problem. *Computers & Operations Research*, *23*(3), 229–235. https://doi.org/10.1016/0305-0548(95)O0026-P

Repoussis, P. P., & Tarantilis, C. D. (2010). Solving the Fleet Size and Mix Vehicle Routing Problem with Time Windows via Adaptive Memory Programming. *Transportation Research Part C: Emerging Technologies*, *18*(5), 695–712. https://doi.org/10.1016/j.trc.2009.08.004

RocketRoute. (2020). Rocket Route. Retrieved August 30, 2020, from http://www.rocketroute.com/

Salhi, S., Imran, A., & Wassan, N. A. (2014). The multi-depot vehicle routing problem with heterogeneous vehicle fleet: Formulation and a variable neighborhood search implementation. *Computers & Operations Research*, *52*, 315–325. https://doi.org/10.1016/j.cor.2013.05.011

Savelsbergh, M. W. P. (1985). Local search in routing problems with time windows. *Annals of Operations Research*, *4*(1), 285–305. https://doi.org/10.1007/BF02022044

Solomon, M. M. (1987). Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research*, *35*(2), 254–265. https://doi.org/10.1287/opre.35.2.254

Solomon, M. M., & Desrosiers, J. (1988). The Window Constrained Routing and Scheduling Problems. *Transportation Science*, *22*(1), 1–13.

Taillard, É., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J.-Y. (1997). A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science*, *31*(2), 170–186. https://doi.org/10.1287/trsc.31.2.170

Taillard, E. D. (1999). A Heuristic Column Generation Method for the Heterogeneous Fleet Vrp. *RAIRO Rech. Opér*, *33*(1), 1–14.

Takan, A. M., & Kasimbeyli, R. (2021). Multiobjective mathematical models and solution approaches for heterogeneous fixed fleet vehicle routing problems. *Journal of Industrial & Management Optimization*, *17*(4), 2073-2095. https://doi.org/10.3934/jimo.2020059

Tang, J., Pan, Z., Fung, R. Y. K., & Lau, H. (2009). Vehicle routing problem with fuzzy time windows. *Fuzzy Sets and Systems*, *160*(5), 683–695. https://doi.org/10.1016/j.fss.2008.09.016

Taniguchi, E., & Shimamoto, H. (2004). Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times. *Transportation Research Part C: Emerging Technologies*, *12*(3–4), 235–250. https://doi.org/10.1016/j.trc.2004.07.007

Taş, D., Jabali, O., & Van Woensel, T. (2014). A Vehicle Routing Problem with Flexible Time Windows. *Computers & Operations Research*, *52*(PART A), 39–54. https://doi.org/10.1016/j.cor.2014.07.005

Tasan, A. S., & Gen, M. (2012). A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries. *Computers and Industrial Engineering*, *62*(3), 755–761. https://doi.org/10.1016/j.cie.2011.11.025

THY. (2020). Turkish Cargo. Retrieved August 20, 2020, from http://www.turkishcargo.com.tr/tr/hizmet-agi-ve-filo/filo

Toth, P., & Vigo, D. (2014). *Vehicle Routing, Problems, Methods, and Applications* (Second). Society for Industrial and Applied Mathematics.

UPS. (2020). UPS. Retrieved August 30, 2020, from https://aircargo.ups.com/en-US/aircraft

Wang, Y., Zhang, S., Guan, X., Peng, S., & Wang, H. (2020). Collaborative multi-depot logistics network design with time window assignment. *Expert Systems With Applications*, *140*. https://doi.org/10.1016/j.eswa.2019.112910

Xu, Y., Wang, L., & Yang, Y. (2012). A New Variable Neighborhood Search Algorithm for the Multi Depot Heterogeneous Vehicle Routing Problem with Time Windows. *Electronic Notes in Discrete Mathematics*, *39*, 289–296. https://doi.org/10.1016/j.endm.2012.10.038

Yan, S., Chu, J. C., Hsiao, F.-Y., & Huang, H.-J. (2015). A planning model and solution algorithm for multi-trip split-delivery vehicle routing and scheduling problems with time windows. *Computers & Industrial Engineering*, *87*, 383–393. https://doi.org/10.1016/j.cie.2015.05.034

Yıldırım, Ü., & Kuvvetli, Y. (2021). Solution of capacitated vehicle routing problem with invasive weed and hybrid algorithms. *International Journal of Industrial Engineering Computations*, *12*(4), 441–456. https://doi.org/10.5267/j.ijiec.2021.4.002

Zare-Reisabadi, E., & Hamid Mirmohammadi, S. (2015). Site dependent vehicle routing problem with soft time window: Modeling and solution approach. *Computers & Industrial Engineering*, *90*, 177–185. https://doi.org/10.1016/j.cie.2015.09.002

Zhang, H., Zhang, Q., Ma, L., Zhang, Z., & Liu, Y. (2019). A hybrid ant colony optimization algorithm for a multi-objective vehicle routing problem with flexible time windows. *Information Sciences*, *490*, 166–190. https://doi.org/10.1016/j.ins.2019.03.070

Zhen, L., Ma, C., Wang, K., Xiao, L., & Zhang, W. (2020). Multi-depot multi-trip vehicle routing problem with time windows and release dates. *Transportation Research Part E*, *135*(January 2019), 101866. https://doi.org/10.1016/j.tre.2020.101866