A new approach for solution of vehicle routing problem with hard time window: an application in a supermarket chain

SERAP ERCAN CÖMERT*, HARUN REŞİT YAZGAN, İREM SERTVURAN and HANİFE ŞENGÜL

Department of Industrial Engineering, Sakarya University, Sakarya, Turkey
e-mail: serape@sakarya.edu.tr; yazgan@sakarya.edu.tr; irem-sertvuran@hotmail.com; hanifesengul55@gmail.com

MS received 4 January 2017; revised 16 May 2017; accepted 18 May 2017; published online 27 November 2017

Abstract. In this study, a vehicle routing problem with hard time windows (VRPHTW) that appears to meet demands of customers' service within time intervals in a supermarket chain is solved. In VRPHTW, to reach a solution by an exact method is quite difficult and sometimes impossible if number of constraints is large enough (i.e., NP-hard), and solution time of a VRPHTW grows exponentially. As the size of the problem grows, a near optimal solution can be found using a heuristic method. A hierarchical approach consisting of two stages as "cluster-first route-second" is proposed. In the first stage, customers are assigned to vehicles using three different clustering algorithms (i.e., K -means, K -medoids and DBSCAN). In the second stage, a VRPHTW is solved using a MILP. The main contribution of the article is that the proposed hierarchical approach enables us to deal with a large size real problem and to solve it in a short time using the exact method. Finally, the proposed approach is employed on a supermarket chain. An instance of the algorithm is demonstrated to illustrate the applicability of the proposed approach and the results obtained are highly favourable.

Keywords. Vehicle routing problem with hard time windows; K -means clustering algorithm; K -medoids clustering algorithm; DBSCAN clustering algorithm; cluster first route second approach.

1. Introduction

Vehicle routing problem (VRP), defined by Dantzig and Ramsar for the first time in literature, is a problem to determine an optimal vehicle route to serve from one or more depots to " n " number of customers [1]. Today, distribution systems have become quite complicated and increase the transportation cost of companies. When a vehicle routing plan is made effectively, it can provide a significant savings in the cost. The objective of VRP is to minimize a travelling distance and/or time among customers. The difficulty of problem varies with the addition of different constraints to the VRP. When the constraints of an earliest time and a latest time to start a service are added to the VRP, the problem becomes a vehicle routing problem with time windows (VRPTW).

The VRPTW is a type of VRP where the customers can be served in a time window $[a_i, b_i]$; a_i and b_i represent an earliest and latest time to start the service. The objective of VRPTW is to design an optimal route using vehicle capacity, service time and time windows. There are two types of the problem such as hard time windows and soft time windows. Vehicles

that arrive earlier than its lower bound have to wait and no additional cost is incurred but vehicles that arrive later than its upper bound are prohibited from serving in the VRPHTW. If a customer is visited outside of his/her time windows, additional costs, called penalty costs, are added to the solution cost in the VRPSTW [2].

There are different solution methods to solve the VRPTW in the literature. These methods are summarized as exact methods, heuristic methods and hierarchical methods. Exact algorithms' performances heavily depend on the problems size because they are NP-hard. Hierarchical method enables dividing into different levels of problem. Different methods can be employed for each level. Cluster first and route second and route first and cluster second can be given as examples of the hierarchical methods. In this article, a cluster first and route second hierarchical method was employed.

Clustering is an unsupervised learning that divides data into groups or clusters. Cluster analysis is based on the principle of maximizing similarity within groups and minimizing between-group similarity. The quality of a clustering method is directly proportional to ensuring this principle. The types of data clustering method vary according to data type and purpose of an application [3].

*For correspondence

In this study, a VRPHTW distribution meeting weekly demands of a supermarket chain that has 78 stores and one depot is discussed. The supermarket chain firm is located in Turkey and deals with products sale in a retail sector. Distribution operations performed for customers are realized on only weekdays. The demand data of 21 weeks are obtained from the firms' past months. Vehicle fleet of the firm consists of homogeneous vehicles and all of them have a maximum of 40 pallets capacity. Moreover, the firm does not prefer using vehicles that are loaded to less than 32 pallets; therefore, the total capacity of each cluster that satisfies customers' total demand, must be 32–40 pallets.

We propose a hierarchical approach to solve this VRPTW. The proposed approach consists of two stages, clustering and routing. In the clustering stage, customers are assigned to vehicles using three different clustering algorithms: *K*-means, *K*-medoids and DBSCAN, with controlling capacity of each cluster. In the routing stage, a travelling salesmen problem (TSP) is solved based on a MILP model that aims to minimize total waiting and travel times.

In VRP, to reach a solution by an exact method is quite difficult and impossible when number of constraints is increased; solution time of a VRP grows exponentially as discussed before. As the size of the problem grows, near optimal solution can be found using a heuristic method. The main contribution of the article is that the proposed hierarchical approach enables us to deal with a large size real problem and to solve this problem in a short time using an exact method.

The remainder of the article is organized as follows. A literature review that consists of studies of VRPTW solved using exact, heuristic and clustering methods is given in section 2. We mention about VRP, its types and solution methods. In addition, details of MILP model is given in section 3. The clustering methods and the types used in this study are discussed in section 4. A real life case study consists of one depot and 78 customers in a supermarket chain is explained and results are given for each clustering method in section 5. Finally, conclusions that we reach in this study are presented in section 6.

2. Literature review

The vehicle routing can be defined as distribution products to specific customers from several warehouses and collection of products from different customers in a logistic system. The VRP is a generalized version of the TSP, and also the VRPs proved to be non-deterministic polynomial-time (NP)-hard problem [4]. It was brought for the first time into literature by Dantzig and Ramsar in 1959. In their study, the authors focused on distribution of gasses to petrol stations by establishing the first mathematical programming model. Later, a saving algorithm that is a heuristic

algorithm for a routing problem was proposed by Clarke and Wright [5]. An inclusive literature review related to the VRP was made by Toth and Vigo [6]. There are extensive researches related to VRP based on different constraints and types. However, in this section, only some of studies related to VRPTW in the literature are summarized.

The following is a brief survey of the literature on the VRPTW. The VRP with hard time windows (VRPHTW) and the VRP with soft time windows (VRPSTW) are types of the VRPTW. Vehicles that arrive earlier than its lower bound have to wait and no additional cost is incurred but vehicles that arrive later than its upper bound are prohibited from serving in the VRPHTW. If a customer is visited outside his/her time windows, additional costs, called penalty costs, are added to the solution cost in the VRPSTW [2].

There are different solution methods to solve the VRPTW in the literature. One of these methods is an exact algorithm. MILP model is employed in these algorithms. Dumas *et al* [7] developed an exact algorithm that uses a column generation to solve a VRPTW with pickup and delivery. Aydemir [8] used a goal programming method to solve the VRPHTW and provided the vehicle routes for assuring material to an automotive factory supplier. Boer [9] developed a mathematical model to provide an approximate solution for the VRPTW. A mathematical programming model was used for VRPTW with pickup and delivery by Cetin and Gencer [10] and Tezer [11]. Cetin and Gencer [12] developed a mathematical model for a heterogeneous fleet of the VRP with simultaneous delivery and pick-up and time windows and applied it on some benchmark problems. Çetinkaya [13] studied a two-stage VRP with arc time windows, which was discussed for the first time in the literature. They proposed a mathematical model and a heuristic algorithm to solve this problem. Taş *et al* [14] and Akca [15] solved the VRPSTW using a linear programming method. Çetinkaya [16] developed a mathematical model to solve location and VRP with arc time windows. Hernandez *et al* [17] proposed two exact solution methods for VRPTW that allowed multiple trips of vehicles. The methods based on branch-and-price consisted of two different set covering formulations. Their methods were tested on the benchmark problems. It was shown that the performance of their methods was efficient. Alvarez and Munari [18] dealt with a type of the VRPTW that service times at customers depended on the number of deliverymen assigned to the routes who serve them. They combined a branch-price-and-cut algorithm and two metaheuristic approaches. They claimed that their developed hybrid method outperformed the branch-price-and-cut algorithm. Parragh and Cordeau [19] proposed a branch-and-price algorithm to solve VRPTW that had trucks and trailers in vehicle fleet. They also used an adaptive large neighbourhood search algorithm to compute initial columns. This algorithm, when applied on a benchmark problem, yielded competitive results.

Heuristic algorithms are also employed to solve the VRPs. Liu and Shen [20] improved the solution using the Clarke and Wright saving algorithm for the VRPTW and multiple vehicle types. Demircioğlu [21] developed a heuristics approach based on saving method and then applied it to a distribution company in Mersin. A similar study using the saving method was also performed by Şahin *et al* [22].

Schulze and Fahle [23] implemented a parallel tabu search algorithm to solve the VRPTW. Ho and Haugland [24] solved the VRPTW and split deliveries with a tabu search algorithm. Jiang *et al* [25] used a tabu search algorithm to solve the VRPTW and heterogeneous fleet.

Barán and Schaerer [26] offered a solution using multiple ant colony optimization (ACO) algorithm (ACO), which is a type of ACO algorithm for the VRPTW. Tokaylı [27] developed a decision support system that uses a route elimination method and the ACO for the VRPTW.

Bouthillier and Crainic [28] suggested a cooperative parallel metaheuristic method for the VRPTW. Dondo and Cerda [29] solved the multi-depot heterogeneous fleet VRPTW with a cluster-based optimization approach.

Dursun [30] added a random-number-coded genetic algorithm (GA) to the literature for the VRPTW. Nazif and Lee [31] proposed a GA using an optimized crossover operator and applied to the VRPTW. They tested their proposed algorithm with benchmark problems and compared it to other algorithms in the literature. They claimed that their proposed algorithm provided competitive results. Kiremitci *et al* [32] solved the multiple vehicle pickup and delivery problem with time windows using a GA and compared to some other heuristics in literature. Pierre and Zakaria [33] developed a new crossover operator that could be used to solve the VRPTW with GA. The aim of developing this operator is to remove the inability of the GA in local optimization. Their methods were tested on the benchmark problems. It was shown that GA with the newly introduced crossover operator is competitive with pure GA.

Gülsoy [34] modified a hunting search algorithm using random-key procedure to solve the VRPHTW. This algorithm was tested on benchmark problems in literature and the results were claimed to be competitive with other heuristics. Chiang and Hsu [35] focused on multi-objective VRPTW and aimed at minimizing the number of vehicles and total travel distance. They proposed a heuristic method based on a universal algorithm and claimed that their algorithm produced competitive results. De *et al* [36] solved a ship routing and scheduling problem with time windows constraints, which is a type of VRP. They formulated the problem as a Mixed Integer Non-Linear Programming Model (MINLP) and solved using Particle Swarm Optimization for Composite Particle (PSO-CP). Iqbal *et al* [37] developed a hybrid meta-heuristic based on artificial BCA for a multi-objective VRPSTW. Cetin and Gencer [38] used a heuristic method to solve the VRP with simultaneous pickup and delivery and hard time windows.

Koç *et al* [39] proposed a hybrid evolutionary algorithm to solve four types of VRPTW that had a heterogeneous fleet. These problems are a fleet size and a mixed VRPHTW (F) and heterogeneous fixed fleet VRP with time windows (H). The main objectives of the problem are to minimize route time (T) and distance (D). Four variants of VRP called FT, FD, HT and HD were solved using their proposed algorithm. They reported that their proposed algorithm produced very effective results on an experimental test. Tan *et al* [40] proposed a new algorithm based on the bacterial foraging optimization (BFO) named adaptive comprehensive learning bacterial foraging optimization (ACLBFO) to solve a VRPTW. They employed two improved BFO (BFO-LDC and BFO-NDC) algorithms and compared them to the classical GA, PSO and the original BFO. They investigated and found that the proposed ACLBFO provided a better performance than what others did. Yan *et al* [41] investigated a split delivery VRP with time windows. The authors proposed a two-step solution algorithm to solve this problem. In the first step, an initial solution was generated and in the second step this initial solution was developed. They reported that their proposed algorithm solved this type of problem effectively and efficiently. Bae and Moon [42] dealt with a multi-depot VRPTW and aimed to minimize the expenses related to travel distances of delivery and installation vehicles. They developed a mixed integer mathematical model and a heuristic algorithm to solve their problem. They mentioned that their proposed algorithms produced effective results for large problems. De *et al* [43] presented a MINLP model to solve the VRPTW that introduced carbon emission, fuel consumption and fuel cost constraints. When the size of the problem grows, the mathematical model is insufficient. Therefore, they employed a PSO-CP algorithm. The results obtained from PSO-CP algorithm were compared to those using PSO and GA. They claimed that their proposed algorithm provided competitive results. Haddadene *et al* [44] dealt with a VRPTW, synchronization and precedence constraints. A MILP, a greedy heuristic, two local search strategies and a hybridization of the two others were proposed to solve the problem. They claimed that their heuristics produced the same results, but their hybrid method was effective and very fast on small and medium data set and yielded good quality for large instances. De *et al* [45] dealt with a maritime inventory routing problem with time windows. They proposed a new mathematical model and solved using PSO-CP, Particle Swarm Optimization–Differential Evolution (PSO-DE), Basic PSO and GA. They claimed that the PSO-CP algorithm for different problem instances was superior to other algorithms. Miranda and Conceição [46] developed a new metaheuristic based on iterated local search algorithm to solve VRPTW that had stochastic travel and service times. They also applied their approach on Solomon's instances. They reported that their approach provided better results compared with other heuristics in the literature. Nadjafi and

Nadjafi [47] solved VRPTW that had multi-depot and heterogeneous fleet. This problem was worked out for the first time. They proposed a constructive heuristic method and this algorithm's performance was compared to the results of benchmark problem. They claim that their algorithm provided an effective solution.

Another method that is used to solve VRP is hierarchical method. Different methods can be employed for each level. Cluster first and route second and route first and cluster second can be given as examples of the hierarchical methods. It is seen that a cluster-based optimization approach is used for a multi-depot heterogeneous fleet VRPTW by Dondo and Cerdã [29]. Hiquebran *et al* [48] solved the VRP using a simulated annealing algorithm based on "cluster-first, then-route". Crainic *et al* [49] used a clustering-based heuristic algorithm for two-stage VRP. The ACO with *K*-means clustering method was applied by Çalışkan [50] and aimed to improve cost of the VRP. Boyzer *et al* [51] proposed a heuristic algorithm for the VRP based on "cluster-first, then-route". In the clustering step, they used a fuzzy *C*-Means method, and then in the routing step, the routes were improved using the tabu Search algorithm. Şen [52] implemented the DBSCAN and the DBSCAN improved with the GA to solve VRP and then the results of algorithms were compared.

A summary of the afore-mentioned studies are given in table 1.

Table 1. Solution algorithms and some studies of VRPTW.

Solution algorithms	Studies
Exact algorithm	Aydemir [8], Boer [9], Cetin and Gencer [10], Tezer [11], Cetin and Gencer [12], Çetinkaya [13], Taş <i>et al</i> [14], Akca [15], Çetinkaya [16], Dumas <i>et al</i> [7], Hernandez [17], Alvarez and Munari [18], Parragh and Cordeau [19]
Heuristic algorithms	Liu and Shen [20], Demircioğlu [21], Şahin <i>et al</i> [22], Schulze and Fahle [23], Ho and Haugland [24], Jiang <i>et al</i> [25], Barán and Schaerer [26], Tokaylı [27], Bouthillier and Crainic [28], Dondo and Cerdã [29], Dursun [30], Nazif and Lee [31], Kiremitci <i>et al</i> [32], Pierre and Zakaria [33], Gülsoy [34], Chiang and Hsu [35], De <i>et al</i> [36], Iqbal <i>et al</i> [37], Cetin and Gencer [38], Koç <i>et al</i> [39], Tan <i>et al</i> [40], Yan <i>et al</i> [41], Bae and Moon [42], De <i>et al</i> [43], Haddadene <i>et al</i> [44], De <i>et al</i> [45], Miranda and Conceição [46], Nadjafi and Nadjafi [47]
Hierarchical algorithms	Dondo and Cerdã [29], Hiquebran <i>et al</i> [48], Crainic <i>et al</i> [49], Çalışkan [50], Boyzer [51], Şen [52]

When the problem size of VRPs grows, to solve them using exact methods becomes impossible. Therefore, studies that have large size problems in the literature are based on heuristic methods. However, this article shows that it is possible to hierarchically solve large size problems with exact solution methods, unlike in the literature.

Our study is similar to those of Dondo and Cerdã [29], Hiquebran *et al* [48], Crainic *et al* [49], Çalışkan [50], Boyzer *et al* [51] and Şen [52] in terms of using clustering analysis to solve a VRPTW. Although clustering algorithms are employed in other papers, they adapted a heuristic algorithm after clustering without controlling capacity of vehicles. However, in our approach, a large size real problem can be solved using the proposed approach with considering a capacity controlled cluster and an exact method.

According to our survey, no study has been undertaken that determines routes for the VRPTW by employing capacity controlled clustering and a MILP. The purpose of employing the clustering technique is to be able to take into account a real life problem while carrying out an evaluation. Therefore, we believe that the proposed research in this study can be considered as a distinctive contribution in terms of exploiting the property.

3. VRP

Nowadays logistics systems have become complicated with ever-growing customers' demand and changing product specifications. It also requires more effective distribution plans. All companies that have products to distribute are faced with the VRP. This problem has led to distribution costs to companies. This distribution costs constitute about 15–20% of the product cost [53]. Customer demand will be met by a vehicle at different points in a classical VRP. The aim of this process is determination of route that meets the need in the shortest time, by the shortest route and at the least cost.

The VRP deals with delivering products from one or more depots to a number of customers or cities by vehicles. Classical VRP is a problem that designs a minimum cost route set for a vehicle fleet. Each route starts from a depot and ends at the same depot after providing a service to a customer set whose demands are known. The VRP has different types according to the constraints. Some of these constraints are vehicle capacity, minimum distance that a vehicle can travel once and time constraint [54].

The VRPTW was brought into logistics systems for the first time by Hax and Candea [55]. Unlike the classic VRP, a time interval in which delivery or pick-up is requested is identified for each customer. This time interval is called a time window $[a_i, b_i]$ [56]. Each customer must be provided services in a time window $[a_i, b_i]$. The problem has a service time s_i for distribution or collection products to each

Figure 1. General structure of VRPTW [50].

customer. When a vehicle reaches a customer, it gives the service at a service time and then it moves on to a next customer or a main store. The general structure of the VRPTW is shown in figure 1.

The VRP with hard time windows (VRPHTW) and the VRP with soft time windows (VRPSTW) are types of the VRP. Vehicles that arrive at an earlier than its lower bound have to wait and no additional cost is incurred but vehicles that arrive later than its upper bound are prohibited from serving the VRPHTW. If a customer is visited outside his/her time windows, additional costs, called penalty costs, are added to the solution cost in the VRPSTW [2].

In this study, The VRP with hard time windows (VRPHTW) is discussed. Vehicles that move from the main depot must provide services to each store in a time window $[a_i, b_i]$. Vehicles that arrive at stores earlier than its lower bound $[a_i]$ have to wait and no additional cost is incurred but vehicles that arrive later than its upper bound $[b_i]$ are prohibited from serving. In addition, vehicles delivering products have a homogeneous capacity.

3.1 Solution methods of the VRP

There are many methods for solution of VRP in literature. These methods are divided into three groups: exact methods, heuristic methods and hierarchical methods. Classification of methods to solve VRP is shown in figure 2.

The problem discussed in this study is a type of VRPHTW. If a vehicle reaches before a time interval, there will be a waiting constraint until an earliest start time of service $[a_i]$ in case of the VRPHTW. Node-based linear decision model, which is one of the exact solution methods, is preferred in solving this problem.

3.1a Node-based linear decision model

Index sets:

$G = (V, A)$ in directional serial,

$V = \{1, 2, \dots, +1\}$ nodes (cities, customers) set, $\{1\}$

depot (centre) and

$A = \{(i, j) | i, j \in V, i \neq j\}$ edges set.

Parameters:

a_i the earliest starting service time for customer i

b_i the latest starting service time for costumer i

$[a_i, b_i]$ time window for costumer i

t_{ij} travelling time from customer i to j , $(i, j) \in A$

F_1 total travelling time

F_2 total waiting time

The decision variables:

x_{ij} if a customer j immediately visits after a customer i using a vehicle, 1; otherwise, 0

the vehicle arriving time of customer i

y_i pre-service waiting time

w_i $t_i + w_i$, i^{th} time until services are provided for city i

s_i $\text{Min } Z = F_1 + F_2$

Objective Function:

Constrains:

$$\sum_{i=1}^{n+1} x_{ij} = 1 \quad j = 1, 2, \dots, n+1 \quad (1)$$

$$\sum_{j=1}^{n+1} x_{ij} = 1 \quad i = 1, 2, \dots, n+1 \quad (2)$$

$$F_1 = \sum_{i=1}^{n+1} \sum_{j=1}^{n+1} t_{ij} x_{ij} \quad (3)$$

$$F_2 = \sum_{i=1}^{n+1} w_i \quad (4)$$

$$s_i + t_{i1} x_{i1} \leq F_1 + F_2 \quad i = 2, 3, \dots, n+1 \quad (5)$$

$$s_i \geq a_i \quad i = 2, 3, \dots, n+1 \quad (6)$$

$$s_i \leq b_i \quad i = 2, 3, \dots, n+1 \quad (7)$$

$$y_i - t_{i1} x_{i1} \geq 0 \quad i = 2, 3, \dots, n+1 \quad (8)$$

$$y_i + (b_i - t_{i1}) x_{i1} \leq b_i \quad i = 2, 3, \dots, n+1 \quad (9)$$

$$s_i = y_i + w_i \quad i = 2, 3, \dots, n+1 \quad (10)$$

$$s_i - y_i + (b_i - t_{i1} + t_{ij}) x_{ij} \leq b_i - t_{i1} \quad i, j = 2, 3, \dots, n+1; i \neq j \quad (11)$$

$$y_i - s_i + (b_j - a_i - t_{ij}) x_{ij} \leq b_j - a_i \quad i, j = 2, 3, \dots, n+1; i \neq j \quad (12)$$

Figure 2. Solution methods of VRP.

$$y_i \geq 0 \quad i = 2, 3, \dots, n+1 \quad (13)$$

$$s_i \geq 0 \quad i = 2, 3, \dots, n+1 \quad (14)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in A \quad (15)$$

Constraints (1) and (2) are assignment constraints of the model, which ensure that all nodes come from one node and also go to only one node. F_1 in inequality (3) represents total travelling time in edges. F_2 in inequality (4) represents total waiting time. Constraint (5) related with time since a vehicle left a depot until arrive a depot and including a vehicle's waiting time. Constraints (6) and (7) represent that each node must be between an earliest and a latest arriving time if a service is given. Constraints (8) and (9) are related to the decision variables. Constraint (10) represents that is total of a time till a vehicle service node (i), the moment of vehicle's arriving time to node (i) and a waiting time till a service node (i). Constraints (11) and (12) are constraints that prevent sub-tours. Constraints (13) and (14) represent that the decision variables (y_i) and (s_i) must be positive. Constraint (15) represents that a decision variable (x_{ij}) must be an integer. The objective function minimizes total travelling time in edge and waiting time.

3.1b The proposed hierarchical approach

In this study, it was not possible to solve our VRPTW by an exact algorithm in a reasonable time. Therefore, we proposed an hierarchical approach that consisted of two phases called cluster first and route second.

The cluster first route second method is a sequential two-phase method: clustering followed by routing, and each of these phases consists of stages. An outline of the method is given as follows:

- Phase I: clustering
- Stage I: forming the clusters
- Stage II: capacity control of the clusters
- Phase II: routing

TSP (B&B)

We tested the proposed approach on the real life example given in section 5.

4. Clustering analysis

Clustering analysis is a kind of unsupervised learning that separates data into clusters or group. The main point of a clustering is to minimize similarity between the classes and maximize the similarity inside each class. In this study, clustering methods K -means, K -medoids and DBSCAN algorithms are used to solve the considered problem. The most commonly used clustering methods are shown in figure 3.

4.1 Capacitated K -means clustering algorithm

K -means, developed by MacQueen in 1967, is one of the clustering algorithms [58]. The formal logic of K -means is to divide data into K clusters. The aim of the clustering analysis is based on the principle of maximizing similarity of groups within themselves and minimizing between-group similarity. Cluster similarity is measured by the average distance between cluster centre points and the dataset point [59].

The algorithm needs a fixed number of clusters. Therefore, the algorithm is called as K -means. K represents the number of clusters and is previously known, a constant positive integer number [60].

K -means algorithm gives acceptable results with all types of data despite the fact that some clustering algorithms achieve better results in some series [61]. A disadvantage of the algorithm is to determine the K values. Therefore, the optimal K number must be found by a trial and error to achieve a successful clustering [62].

Figure 3. The most commonly used clustering methods [57].

The steps of *K*-means are given in table 2 [58].

4.2 Capacitated *K*-medoids clustering algorithm

K-medoids algorithm was developed by Kaufman and Rousseeuw to reduce sensitivity of *K*-means in data with noise and exception data [63]. The element at most centre points in the algorithm is accepted as a new cluster centre. Thus, the exception data are not allowed to scroll cluster centre towards edges.

There are many different types of *K*-medoids algorithm. Partitioning Around Medoids (PAM) is the first *K*-medoids algorithm asserted. *K* number is accepted as cluster's centre for the PAM algorithm. When a new element (point) is added to the cluster, if a point contributes to the cluster, this point must be accepted as a new centre of previous old centre. When a point's contribution to cluster's improving is confirmed, this point may be a new centre and an old centre is accepted as a simple element of the cluster. Despite the fact that PAM provides very good results in small databases, it shows underperformance in complex databases.

K-medoids algorithm is summarized in table 3 [64].

4.3 Capacitated DBSCAN clustering algorithm

DBSCAN algorithm, which is one of density-based clustering methods, was developed by Martin Ester, Hans-Peter Kriegel,

Jörg Sander and Xiaowei Xu [65]. In DBSCAN centre-based approach, the density for a point in database is predicted by considering number of points located within an exact distance (the longest neighbourhood radius, radius of Eps) from this point. This point number contains point itself [66].

There are two parameters of DBSCAN algorithm: Eps and MinPts.

Eps is the longest neighbourhood radius. MinPoints represents the minimum number of elements located in a neighbourhood region of radius Eps.

Steps of the DBSCAN algorithm are explained in table 4.

The DBSCAN is a very useful clustering method, especially for large databases and databases that contain noisy data. It is often used for clusters with different volumes and shapes [67].

Eps and MinPts are input parameters for the DBSCAN. All elements in databases are controlled by starting from any element. It is passed on to other elements without doing any operation if the controlled element belongs to any cluster before. If the element is clustered before, neighbours of the element located in Eps Neighbourhood region are determined. If the number of neighbours is more than MinPts, the element and its neighbours are named as a new cluster. Afterwards, new neighbours are determined for each neighbour not clustered before by examining new regions. If the number of points in the neighbourhood region is more than MinPts, it is included in this cluster [68].

Table 2. *K*-means algorithm steps.

Step	Activities
1	Determine the first set centre
2	Calculate the distance between specified point and centre points. All objects are assigned to the set that is the closest to themselves
3	New centre points switch mean values of all objects in that set
4	Repeat steps 2 and 3 until centre points are not changed

Table 3. *K*-medoids algorithm steps.

Step	Activities
1	Determine <i>K</i> set number.
2	Choose <i>K</i> objects as initial medoids
3	Assign the rest of the sets to the closest medoids <i>x</i>
4	Calculate objective function
5	Determine the <i>y</i> point coincidence
6	If changing <i>x</i> and <i>y</i> minimizes the objective function, change <i>x</i> and <i>y</i>
7	If there is no change, keep repeating Steps 3–6

5. Application

5.1 Definition of the problem

A supermarket chain firm located in Turkey deals with product sales in a retail sector. Distribution operations performed for the customers are realized on only weekdays. There is only one depot and 78 stores of the supermarket chain. Locations of the main depot and stores on the map are shown in figure 4. The demand data of 21 weeks are predicted using average values for the past 3 months. Vehicle fleet of the firm consists of homogeneous vehicles and all of them have a maximum of 40 pallets capacity. Moreover, the firm does not prefer using the vehicles that have less than 32 pallets capacity; therefore, the total capacity of the clusters related to customers must be between 32 and 40 pallets. The earliest and latest delivery times $[a_i, b_i]$ are defined for each branch after meetings with the firm. The earliest and latest start times for the service of customers are given in table 5. If a vehicle arrives at a branch earlier, the vehicle must wait until the earliest start time for service; however, in case it arrives at the branch later, the customer does not accept the service.

5.2 Solution of the problem

The distribution problem is solved in two stages. In the first stage, considering the density of customers' weekly demand, the distribution problem is solved using *K-medoids* and *K-means*, which are partitioning methods, and *DBSCAN* algorithm, which is one of the density-based clustering methods according to the distance. In second stage, an optimal route is determined considering the earliest and latest delivery times of customers using the mathematical model. As a result, the actual and the proposed approach results are compared statistically using analysis of variance (ANOVA) test for the supermarket chain.

In this section, solutions of the problem are given by considering the amount of first week demands in detail. The results of 21 weeks and the existing data of the firm are summarized in table 9.

Table 4. The stages of the DBSCAN algorithm.

Step	Activity
1	Eps radius neighbourhood regions of each element in databases are searched
2	In this region, if number of neighbor points is bigger than MinPts, the element and its neighbors are accepted as a new cluster
3	Density-connected clusters are combined
4	The process is ended if no new object is added into the cluster

5.2a Clustering with capacity control

Process of K-means algorithm:

Firstly, the number of clusters is determined by the *K-means* algorithm. There are no rules to define the number of clusters. It mainly depends on data and tuning with trial and error. After determining the number of clusters, the clusters are created using the *K-means* algorithm. The total number of demands of each cluster is calculated and controlled such that capacity constraint is satisfied. The clusters that satisfy a capacity constraint are accepted as favourable clusters. The stores in clusters that do not satisfy the capacity constraint are removed from all clusters and this process is continued till all stores belong to only one cluster. The clusters generated after clustering, the stores belonging to clusters and number of demands are given in table 6.

Process of K-medoids algorithm:

An element which are located in the centre, this element is assumed as the centre point of a new cluster centre. It represents cluster centre of the *K-medoids* algorithm. In addition, *K* number point is assumed as cluster center of the algorithm. Rest of points are added into each cluster with controlling whether the cluster is improved or not. If there is an improvement in a cluster, new added point is going to be as a center, the old center point will be a cluster element.

In the *K-medoids* algorithm, firstly, the number of cluster is identified using trial and error. After the number of cluster is identified, the clusters are generated using the *K-medoids* algorithm. The total number of demands of each cluster is determined and controlled such that capacity constraint is satisfied. The clusters that satisfy the capacity constraint are accepted as favourable clusters. The stores in

Figure 4. Location of the main depot and stores.

Table 5. The earliest and the latest arriving times to customers (min).

Customer number	EAT	LAT	Customer number	EAT	LAT	Customer number	EAT	LAT
1	10	40	27	264	274	53	394	404
2	306	320	28	90	100	54	217	227
3	237	247	29	299	309	55	407	417
4	179	189	30	136	146	56	94	104
5	409	419	31	324	334	57	137	147
6	291	301	32	341	351	58	80	90
7	274	284	33	369	379	59	75	85
8	390	400	34	226	236	60	200	210
9	147	157	35	80	90	61	230	250
10	373	383	36	120	130	62	274	284
11	372	382	37	412	422	63	90	100
12	193	203	38	83	93	64	419	429
13	283	293	39	406	416	65	309	319
14	46	56	40	268	278	66	178	188
15	36	46	41	420	430	67	256	266
16	265	275	42	48	58	68	223	233
17	384	394	43	222	232	69	46	56
18	428	438	44	61	71	70	125	135
19	72	82	45	432	442	71	69	79
20	283	293	46	12	42	72	140	150
21	235	245	47	382	392	73	155	165
22	46	56	48	402	412	74	210	220
23	172	182	49	427	437	75	34	54
24	88	98	50	61	71	76	423	433
25	391	401	51	222	232	77	199	209
26	159	169	52	135	145	78	246	276

Earliest arriving time (EAT), latest arriving time (LAT).

Table 6. Clustered stores for each vehicle as a result of the K -means algorithm.

Cluster number	Sequence of stores	Total pallet
1	72-74-75-76-77-78	5+3+5+9+3+7=32
2	30-52-53-54-55-65-57	3+5-5-9-5-5-7=39
3	7-8-9-10-11-12	7+9+7+5+5+7=40
4	66-67-68-69-71-73	5+5+9+5+3+7=34
5	1-2-3-4-5-6	5+7+9+7+9+3=40
6	17-18-20-22-23-25	5+7+5+5+5+5=32
7	13-14-24-32-34-35-37	5+5+3+5+5+5+5=33
8	15-16-19-21-26-27-29	7+5+5+9+5+3+5=39
9	28-31-33-36-38-39-41-44-46	5+5+3+3+5+3+5+5+5=39
10	40-42-43-45-47-48-50	5+3+5+3+7+7+9=39
11	49-51-58-59-61-62	5+9+5+7+5+7=38
12	60-63-64-65-70	7+9+5+5+7=33

Table 7. Clustered stores for each vehicle as a result of the K -medoids algorithm.

Vehicles	Sequence of stores	Total pallet
1	30-52-53-54-56-57	3+5+5+9+5+7=34
2	4-5-6-67-69-70-71	7+9+3+5+5+7+3=39
3	14-72-73-75-77-78	5+5+7+5+3+7=32
4	15-16-27-40-43-47-48	7+5+3+5+5+7+7=39
5	13-24-26-32-34-35-37-74	5+3+5+5+5+7+7=37
6	17-18-22-41-42-44-45	5+7+5+5+3+5+3=33
7	20-23-25-28-31-33-36-39-46	5+5+5+5+5+3+3+3+5=39
8	1-2-3-19-29-76	5+7+9+5+5+9=40
9	49-50-58-59-61-62	5+9+5+7+5+7=38
10	7-8-9-10-11-66	7+9+7+5+5+5=38
11	12-21-38-55-68	7+9+5+5+9=35
12	51-60-63-64-65	9+7+9+5+5=35

clusters that do not satisfy the capacity constraint are removed from all clusters and the process is continued until all stores belong to only one cluster. The clusters generated after clustering, the stores belonging to clusters and number of demands are given in table 7.

Process of DBSCAN algorithm:

The fundamental principle of the DBSCAN algorithm is to generate a cluster that has the least number of points within a given radius when any point is accepted as the centre point [65]. There must be points in an Epsilon neighbourhood numbered as MinPts for each point in a cluster. Because of the MinPoints express minimum point number in a cluster with considering demands' amount, a vehicle's capacity between 32 and 40 pallets, the MinPoints is calculated as 4. The clusters are generated using the DBSCAN algorithm, starting from Epsilon as 1, while the MinPoints is constant until a favourable cluster is generated. Demands of clusters generated are added and controlled such that capacity constraint is satisfied. The clusters that satisfy the capacity constraint are accepted as favourable clusters. The stores in clusters that do not satisfy the capacity constraint are removed from all clusters and the process is continued until all stores belong to only one cluster. The clusters generated after clustering, the stores belonging to clusters and number of demands are given in table 8.

5.2b Routing

In the first stage, routes of clusters are generated and the routes' total travelling time is calculated using the mathematical model under the earliest and latest delivery time constraints (table 1). The routes to be followed and the total time by vehicle for each algorithm are given in tables 9, 10 and 11.

Remaining results of 21 weeks and the actual values are summarized in table 12.

5.3 ANOVA test

ANOVA is used to determine whether there are any significant differences among the means of three or more

Table 8. Clustered stores for each vehicle as a result of the DBSCAN algorithm.

Vehicles	Sequence of stores	Total pallet
1	7-9-60-63-64-65	7+7+7+9+5+5=40
2	30-52-53-54-56-57	3+5+5+9+5+7=34
3	3-8-58-69-73	9+9+5+5+7=35
4	49-50-51-59-61	5+9+9+7+5=35
5	10-11-12-66-68-70	5+5+7+5+9+7=38
6	15-16-27-40-43-47-48	7+5+3+5+5+7+7=39
7	13-14-24-32-34-35-37-74	5+5+3+5+5+5+5+3=36
8	1-2-62-67-71-76	5+7+7+5+3+9=36
9	6-26-55-72-75-77-78	3+5+5+5+5+3+7=33
10	4-5-17-18-19-29	7+9+5+7+5+5=38
11	20-21-23-25-36-41-45-46	5+9+5+5+3+5+3+5=40
12	22-38-31-33-38-39-42-44	5+5+5+3+5+3+3+5=34

Table 9. The routes and the total time for each vehicle using the K-means algorithm.

Vehicle	Routes	Total time (min)
1	79-75-72-77-74-78-76-79	468.2
2	79-56-57-52-30-54-53-55-79	450.1
3	79-9-12-7-11-10-8-79	439.7
4	79-69-71-73-66-68-67-79	299.5
5	79-1-4-3-6-2-5-79	445.7
6	79-22-23-20-17-25-18-79	470.6
7	79-14-35-24-34-13-32-37-79	459.1
8	79-15-19-26-21-27-16-29-79	345.8
9	79-46-44-38-28-36-31-33-39-41-79	463
10	79-42-50-43-40-47-48-45-79	475.4
11	79-59-58-51-61-62-49-79	459.5
12	79-63-70-60-65-64-79	433
Total time		5209.6

Table 10. The routes and the total time for each vehicle using the K-medoids algorithm.

Vehicle	Routes	Total time (min)
1	79-52-56-54-30-57-53-79	430.5
2	79-71-5-69-67-6-4-70-79	445.7
3	79-14-72-77-73-75-78-79	294.1
4	79-15-16-47-27-43-40-48-79	449.5
5	79-35-24-34-74-13-37-26-32-79	459.1
6	79-22-17-41-44-45-18-42-79	475.4
7	79-31-39-25-23-33-36-28-46-20-79	448
8	79-1-2-29-19-76-3-79	468.2
9	79-50-49-58-61-59-62-79	459
10	79-8-66-11-9-7-10-79	439.7
11	79-38-68-12-55-21-79	450.1
12	79-65-51-64-63-60-79	433
Total time		5252.3

independent groups. A statistical analysis is completed using ANOVA test (table 9). Two hypothesis are defined as follows:

$H_0: \mu_1 = \mu_2$ (there is no difference among means of total times of three algorithms and actual values).

$H_1: \mu_1 \neq \mu_2$ (mean of total times of three algorithms and actual values are different).

Before we use ANOVA test, firstly, variances should be controlled whether homogeneous or not; p (sig.) value of 0.107 is greater than 0.05 in table 13; variances of groups are homogeneous. As variances of groups are homogeneous, results of F test in table 14 are significant.

Because of the p (sig.) value in ANOVA table, H_1 hypothesis is accepted. This implies that there is a

Table 11. The routes to be followed and the total time by vehicle using the DBSCAN algorithm.

Vehicle	Routes	Total time (min)
1	79-65-7-64-60-63-9-79	433
2	79-52-56-54-30-57-53-79	430.5
3	79-8-58-3-73-69-79	439.7
4	79-50-49-51-61-59-79	459
5	79-68-66-70-11-12-10-79	412.8
6	79-15-16-47-27-43-40-48-79	449.5
7	79-35-14-34-74-13-24-37-32-79	459.1
8	79-1-2-76-71-67-62-79	468.2
9	79-26-72-6-77-75-55-78-79	450.1
10	79-18-17-19-4-29-5-79	470.6
11	79-41-25-23-21-36-45-46-20-79	475.4
12	79-39-31-33-38-42-22-28-44-79	448
Total time		5395.9

Table 12. Total time of the *K*-means, the *K*-medoids, the DBSCAN algorithms and actual values (min).

Week	<i>K</i> -means	<i>K</i> -medoids	DBSCAN	Actual values
1	5209.6	5252.3	5395.9	5410.5
2	5380.1	5393	5223.9	5509.2
3	5094.6	5239.6	5110.5	5300.7
4	5137	5572.1	5127.8	5612.5
5	5390	5722.9	5164.7	5803.6
6	5306.8	5839.3	5412.8	5901.5
7	5585.9	5252.3	5159.2	5615.4
8	5203.8	5688.4	5193.4	5713.4
9	5076.1	5366.7	5251.4	5445.2
10	5289	5521	5211.8	5600.9
11	5167.9	5239.2	5554.5	5614.1
12	5094.8	5657.8	5240.5	5744.4
13	5105.5	5560	5074.5	5714.2
14	5017.9	5299.3	5192.8	5212.4
15	4885.5	5044	5103.2	5000
16	5202.7	5181.1	5072.4	5278.3
17	5259.2	5176.9	5371	5341.2
18	5165	5717.7	5088.7	5232.1
19	5010.7	5670.1	5222	5746.8
20	4881.6	5416.6	5566.2	5547.6
21	5017.9	5455.2	5599.5	5423.5

Table 13. Test of homogeneity of variances.

Levene statistic	df1	df2	Sig.
2.094	3	80	.107

significant difference among three algorithms and actual values. Mean total times of three algorithms and actual value are summarized in table 15. Mean total times are found using the *K*-means, the *K*-medoids, the DBSCAN and actual values as 5165.7905, 5441.2381, 5254.1286 and 5512.7381 min, respectively.

Table 14. ANOVA test.

	Sum of squares	Df	Mean square	<i>F</i>	Sig.
Between groups	1632908.287	3	544302.762	13.8	.000
Within groups	3155408.876	80	39442.611		
Total	4788317.163	83			

Table 15. Descriptive statistics.

	<i>N</i>	Mean	Std. deviation	Std. error
1 (<i>K</i> -means)	21	5165.7905	168.96015	36.87013
2 (<i>K</i> -medoids)	21	5441.2143	222.39865	48.53137
3 (DBSCAN)	21	5254.1286	165.26380	36.06352
4 (Actual values)	21	5512.7381	229.01883	49.97601
Total	84	5343.4679	240.18861	26.20673

When the obtained results are examined, mean total times that are found using the *K*-means, the *K*-medoids and the DBSCAN algorithm are better than actual values shown. This analysis shows that businesses can achieve better results when they use the scientific methods to route when they are distributing products, rather than by trial-and-error method.

When we examine the results obtained from the clustering algorithms, *K*-means algorithm provides better solution than others shown. If the business uses the proposed hierarchical method with *K*-means, the *K*-medoids and the DBSCAN, it will save, respectively, 6%, 1% and 5% on the total times.

6. Conclusions

Today, to increase market share, companies should minimize logistics activities and distribution costs, which are a significant proportion of total costs, for rapidly changing and evolving competitive environment. This is occurred by making the distribution plan appropriately. The vehicle routing, especially concerning the distribution and logistics sector, is an area where competition is emerging. Companies will make optimization of the distribution network with solution of the VRP frequently in real life.

There are two types of VRPTW such as hard time windows and soft time windows. In this article, VRP hard time windows are studied. There is an exact time interval over which customers accept service and the customers do not accept the service outside this time interval. In this model, created depending on the time window, the aim is to generate distribution routes by minimizing total travelling and waiting time.

The VRP of the firm serving the retail sector is discussed in order to meet weekly demand of the firm. This firm has

78 stores and demand data of 21 weeks are obtained by calculating customers' demands average for past 3 months. Vehicle fleet of the firm consists of homogeneous vehicles and all of them have a maximum of 40 pallets capacity. Moreover, the firm does not prefer sending the vehicles that have less than 32 pallets capacity; therefore, the total capacity of the clusters related to customers must be between 32 and 40 pallets. The clusters for all customers are obtained by considering weekly demand of customers and pallet constraint of vehicles with the K -means, the K -medoids and the DBSCAN algorithms. Each cluster represents the vehicle that will be servicing. The aim of this mathematical model is to minimize total travelling and waiting time of routes established. The actual data of the firm and the results obtained using three algorithms are compared using ANOVA test. The mean total times are found to be 5165.7905 min (the K -means), 5441.2381 min (the K -medoids), 5254.1286 min (the DBSCAN) and 5512.7381 min (firm's data). The results illustrate that the K -means provides better solutions than others do. Due to these favourable results, the proposed approach will add value to the company if implemented.

We assert that the "cluster-first route-second" should be methodologically determined for a VRPHTW. Based on this we propose our approach to solve a large sized real life problem.

References

- [1] Dantzig G and Ramser J 1959 The truck dispatching problem. *Manag. Sci.* 6(1): 80–91
- [2] Badeau P, Guertin F, Gendreau M, Potvin J and Taillard E 1997 A parallel tabu search heuristic for the vehicle routing problem with time windows. *Transp. Res. C* 5(2): 109–122
- [3] Han J and Kamber M 2001 *Data mining and concepts techniques*. San Francisco: Morgan Kaufmann
- [4] Laporte G and Semet F 2002 Classical heuristics for the capacitated VRP. In: Toth P and Vigo D (Eds.) *The vehicle routing problem*, SIAM Monographs on Discrete Mathematics and Applications, SIAM, Philadelphia, pp. 109–128
- [5] Clarke G and Wright J W 1964 Scheduling of vehicles from a central depot to a number of delivery points. *Oper. Res.* 12: 568–581
- [6] Toth P and Vigo D 2002 *The vehicle routing problem*. USA: Society for Industrial and Applied Mathematics
- [7] Dumas J E, Wolf L C, Fisman S N and Culligan A 1991 Parenting stress, child behavior problems, and dysphoria in parents of children with autism, Down syndrome, behavior disorders, and normal development. *Exceptionality* 2(2): 97–110
- [8] Aydemir E 2006 *Vehicle routing problem with soft time windows and a case study*. Master Thesis, Gazi University, Ankara
- [9] Boer J 2008 *Approximate models and solution approaches for the vehicle routing problem with multiple use of vehicles and time windows*. Master Thesis, Middle East Technical University, Ankara
- [10] Cetin S and Gencer C 2010 Vehicle routing problems with hard time windows and simultaneous pickup and delivery: a mathematical model. *J. Fac. Eng. Arch. Gazi Univ.* 25(3): 579–585
- [11] Tezer T 2009 *An exact approach for a vehicle routing problem with pickup and delivery time windows and sample applications*. Master Thesis, Balıkesir University, Balıkesir
- [12] Cetin S and Gencer C 2011 Heterogeneous fleet vehicle routing problems with time windows and simultaneous pickup and delivery: mathematical model. *Int. J. Res. Dev.* 3(1): 19–27
- [13] Çetinkaya C 2011 *Two stage vehicle routing problem with arc time windows*. Master Thesis, Gazi University, Ankara
- [14] Taş D, Jabali O and Woensel T V 2014 A vehicle routing problem with flexible time windows. *Comput. Oper. Res.* 52: 39–54
- [15] Akca K 2015 *Supply chain mathematical model for the vehicle routing problem with time windows*. Master Thesis, Uludağ University, Bursa
- [16] Çetinkaya C 2014 *Plant location and vehicle routing problem with arc time windows for military shipment to terror region*. PhD Thesis, Gazi University, Ankara
- [17] Hernandez F, Feillet D, Giroudeau R and Naud O 2016 Branch-and-price algorithms for the solution of the multi-trip vehicle routing problem with time Windows. *Eur. J. Oper. Res.* 249: 551–559
- [18] Alvarez A and Munari P 2017 An exact hybrid method for the vehicle routing problem with time windows and multiple deliverymen. *Comput. Oper. Res.* 83: 1–12
- [19] Parragh S N and Cordeau J F 2017 Branch-and-price and adaptive large neighborhood search for the truck and trailer routing problem with time windows. *Comput. Oper. Res.* 83: 28–44
- [20] Liu F H and Shen S Y 1999 A method for vehicle routing problem with multiple vehicle types and time windows. *Proc. Natl. Sci. Coun. ROC(A)* 23(4): 526–536
- [21] Demircioğlu M 2009 *A heuristic approach to vehicle routing problem and an application*. PhD Thesis, Çukurova University, Adana
- [22] Şahin M, Şahin G, Çavuşlar G, Özcan T and Tüzün D 2010 Separable weight pickup and delivery problem using tabu search algorithm. In: *Proceedings of the 30th National Meeting on Operational research and Industrial Engineering, Sabancı University, Istanbul*
- [23] Schulze J and Fahle T 1999 A parallel algorithm for the vehicle routing problem with time window constraints. *Ann. Oper. Res.* 86: 585–607
- [24] Ho S and Haugland D 2004 A tabu search heuristic for the vehicle routing problem with time windows and split deliveries. *Comput. Oper. Res.* 31: 1947–1964
- [25] Jiang J, Ng K M, Poh K L and Teo K M 2014 Vehicle routing problem with a heterogeneous fleet and time windows. *Exp. Syst. Appl.: Int. J.* 41(8): 3748–3760
- [26] Barán B and Schaerer M 2003 A multiobjective ant colony system for vehicle routing problem with time windows. In: *Proceeding of the 21st IASTED International Conference on Applied Informatics, Innsbruck, Austria*, 10–13 February, pp. 97–102
- [27] Tokaylı M A 2005 *A decision support system for the vehicle routing problem with time windows*. Master Thesis, Gazi University, Ankara

- [28] Bouthillier A L and Crainic T G 2005 A cooperative parallel meta heuristic for the vehicle routing problem with time windows. *Comput. Oper. Res.* 32: 1685–1708
- [29] Dondo R and Cerda J 2007 A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. *Eur. J. Oper. Res.* 176: 1478–1507
- [30] Dursun P 2009 *Modeling vehicle routing problem with time windows with genetic algorithm*. Master Thesis, İstanbul Technical University, İstanbul
- [31] Nazif H and Lee L S 2010 Optimized crossover genetic algorithm for vehicle routing problem with time windows. *Am. J. Appl. Sci.* 7(1): 95–101
- [32] Kiremitci B, Kiremitci S and Keskintürk T 2014 A real valued genetic algorithm approach for the multiple vehicle pickup and delivery problem with time windows. *Istanbul Univ. J. Sch. Bus.* 43(2): 391–403
- [33] Pierre D M and Zakaria N 2017 Stochastic partially optimized cyclic shift crossover for multi-objective genetic algorithms for the vehicle routing problem with time-windows. *Appl. Soft Comput.* 52: 863–876
- [34] Gülsoy N 2013 *The solution vehicle routing problem with hard time windows with hunting search algorithm*. Master Thesis, Erciyes University, Kayseri
- [35] Chiang T C and Hsu W H 2014 A knowledge-based evolutionary algorithm for the multiobjective vehicle routing problem with time windows. *Comput. Oper. Res.* 45: 25–37
- [36] De A, Awasthi A and Tiwari M K 2015 Robust formulation for optimizing sustainable ship routing and scheduling problem. *IFAC-PapersOnLine* 48(3): 368–373
- [37] Iqbal S, Kaykobad M and Rahman M S 2015 Solving the multi-objective vehicle routing problem with soft time windows with the help of bees. *Swarm Evol. Comput.* 24: 50–64
- [38] Cetin S and Gencer C 2015 A heuristic algorithm for vehicle routing problems with simultaneous pick-up and delivery and hard time windows. *Open J. Soc. Sci.* 3: 35–41
- [39] Koç Ç, Bektaş T, Jabali O and Laporte G 2015 A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows. *Comput. Oper. Res.* 64: 11–27
- [40] Tan L, Lin F and Wang H 2015 Adaptive comprehensive learning bacterial foraging optimization and its application on vehicle routing problem with time windows. *Neurocomputing* 151: 1208–1215
- [41] Yan S, Chu J C, Hsiao F Y and Hu H J 2015 A planning model and solution algorithm for multi-trip split-delivery vehicle routing and scheduling problems with time windows. *Comput. Ind. Eng.* 87: 383–393
- [42] Bae H and Moon I 2016 Multi-depot vehicle routing problem with time windows considering delivery and installation vehicles. *Appl. Math. Model.* 40: 6536–6549
- [43] De A, Mamanduru V K R, Gunasekaran A, Subramanian N and Tiwari M K 2016 Composite particle algorithm for sustainable integrated dynamic ship routing and scheduling optimization. *Comput. Ind. Eng.* 96: 201–215
- [44] Haddadene S R A, Labadie N and Prodhon C 2016 A GRASP x ILS for the vehicle routing problem with time windows, synchronization and precedence constraints. *Expert Syst. Appl.* 66: 274–294
- [45] De A, Kumar S K, Gunasekaran A and Tiwari M K 2017 Sustainable maritime inventory routing problem with time window constraints. *Eng. Appl. Artif. Intell.* 61: 77–95
- [46] Miranda D M and Conceição S V 2017 The vehicle routing problem with hard time windows and stochastic travel and service time. *Expert Syst. Appl.* 64: 104–116
- [47] Nadjafi B A and Nadjafi A A 2017 A constructive heuristic for time-dependent multi-depot vehicle routing problem with time-windows and heterogeneous fleet. *Eng. Sci.* 29: 29–34
- [48] Hiquebran D T, Alfa A S, Shapiro J A and Gittos D H 2007 A revised simulated annealing and cluster-first route-second algorithm applied to the vehicle routing problem. *Eng. Optim.* 22: 77–107
- [49] Crainic T G, Mancini S, Perboli G and Tadei R 2008 Clustering-based heuristics for the two-echelon vehicle routing problem. Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation
- [50] Çalışkan K 2011 *Improving the cost of vehicle routing problem by using ant colony optimization with clustering techniques*. Master Thesis, TOBB University of Economics and Technology, Ankara
- [51] Boyzer Z, Alkan A and Fiğlalı A 2014 Cluster-first, then-route based heuristic algorithm for the solution of capacitated vehicle routing problem. *Int. J. Inf. Technol.* 7(2): 29–37
- [52] Şen T 2014 *Solution of the capacity constraint vehicle routing problem with cluster and genetic algorithm based approach: a retail chain application*. Master Thesis, Sakarya University, Sakarya
- [53] Rushton A, Croucher P and Baker P 2006 *Handbook of logistics and distribution management*, 3rd edn. Kogan Page, Limited
- [54] Ho W, Ho G T S, Ji P and Lau H C W 2008 A hybrid genetic algorithm for the multi-depot vehicle routing problem. *Eng. Appl. Artif. Intell.* 21: 548–557
- [55] Hax A C and Candea D 1984 *Production and inventory management*. Englewood Cliffs, NJ: Prentice-Hall
- [56] Glynn P and Robinson S 1997 *Springer series in operations research*. New York: Springer
- [57] Gerdan O 2007 *The vehicle routing problem with simultaneous delivery and pick-up with material flows among customers and heuristic solution*. PhD Thesis, Gazi University, Ankara
- [58] MacQueen J B 1967 Some methods for classification and analysis of multivariate observations. In: *Proceedings of the Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297
- [59] Berkhin P 2002 *Survey of clustering data mining techniques*. San Jose, California, USA: Accrue Software Inc
- [60] Xu R and Wunsch II D 2005 Survey of clustering algorithms. *IEEE Trans. Neural Netw.* 16(3): 645–678
- [61] Yünel Y 2010 *Development of k-means clustering algorithm using genetic algorithm*. Undergraduate Thesis, İstanbul Technical University, İstanbul
- [62] Berkhin P 2006 A survey of clustering data mining techniques. In: *Grouping multidimensional data: recent advances in clustering*. Berlin–Heidelberg: Springer
- [63] Kaufman L and Rousseeuw P 1987 *Clustering by means of medoids*. No. 87, Reports of the Faculty of Mathematics and Informatics, Delft University of Technology
- [64] Işık M 2006 *Data mining applications using partitional clustering methods*. Master Thesis, Marmara University, İstanbul
- [65] Ester M, Kriegel H P, Sander J and Xu X 1996 A destiny-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the second International Conference on Knowledge Discovery and Data Mining*, Portland, pp. 226–231

- [66] Güven A, Bozkurt Ö and Kalıpsız O 2007 The future of data mining. In: *Reports of IX Informatics Conference, Dumlupınar University, Kütahya*
- [67] Moreira A, Santos M Y and Carneiro S 2005 *Destiny-based clustering algorithms—DBSCAN and SNN*. University of Minho, Portugal
- [68] Bilgin T T and Çamurcu Y 2005 Comparative comparing of DBSCAN, OPTICS and K-Means algorithms. *J. Polytechn.* 8(2): 139–145