Decision Analytics Journal 4 (2022) 100113

Contents lists available at ScienceDirect

Decision Analytics Journal

journal homepage: www.elsevier.com/locate/dajour

The applications of multiple route optimization heuristics and meta-heuristic algorithms to solid waste transportation: A case study in Turkey

Ufuk Dereci [a], Muhammed Erkan Karabekmez [a,b,*]

[a] Istanbul Medeniyet University, Institute of Graduate Studies, Department of Engineering Management, Turkey
[b] Istanbul Medeniyet University, Department of Bioengineering, Turkey

ARTICLE INFO

Keywords:
Vehicle routing
Optimization
Heuristics
Meta-heuristics
Solid waste collection
OR-tools

ABSTRACT

The dense population and high solid waste production make it challenging to determine the optimal routes for garbage collection trucks in metropolitan cities. Moreover, incorrectly planned routes cause waste of fuel, time, and other resources in the municipal garbage collection process. Vehicle Routing Problem (VRP) is a combinatorial optimization problem that aims to find optimal routes for a certain number of vehicles that start from a depot, visit all customers, and return to the depot. This study aims to find the optimal routes that satisfy all the needs by considering the waste collection process of the Istanbul Umraniye district. Data on the currently used routes, daily tonnages, number of truck trips, and the number and locations of containers in the Umraniye district were obtained from the Umraniye municipality. We solve the route optimization problem of the Umraniye municipality with multiple heuristics and meta-heuristic algorithms like the nearest neighbor, savings, Christofides, simulated annealing, greedy descent, guided local search, and Tabu Search algorithms. We show the examined heuristics and meta-heuristics can produce fast and feasible solutions to large-size real-life problems.

1. Introduction

Along the last century, with the rapid pace of industrialization, and urbanization and impact of humanity on the environment has reached to an unmatched level. The aim for using the limited resources in the best way has turned into the aim for using the wastes comes from these sources in the most efficient way over time. The term, "waste", can be defined as any substance discarded into the environment because of lack of activity in the simplest form. In a more detailed way, waste is any substance that are leftovers or whose use is no longer possible for its owner regardless of any chance of reuse or recycle, which are needed or asked to throw away from living environment that may cause environmental diseases or health problems if they are not removed [1].

In Turkey, where our case study conducted, local municipalities are responsible for the waste collection services. Especially in metropolitan cities the management of waste collection is a complex process, since the population density is high and as a result the amount of produced solid waste is also high. For example, an average of 18 thousand tons of household waste is generated daily in Istanbul which has 39 districts [2]. District municipalities aim at maximizing the satisfaction level of the citizens and minimizing the cost while removing urban environmental solid pollution. As a result of any mistake in the plan, wastes can overflow from the containers, pollute the streets, and cause unhealthy conditions. The lack of proper waste management service would cause dissatisfaction of citizens with the responsible municipality. On the other hand, with an inefficient plan, garbage can be collected satisfactorily but it can cause spending more recourses (like personnel, vehicle, fuel, time, etc.) than necessary. Therefore, this plan should be considered as an important engineering problem and should be dealt with comprehensively.

Although there are many studies on both Vehicle Routing Problem (VRP) and Waste Management Route Optimization in the literature, the number of studies dealing with the two issues together is not enough. Han and Cueto [3] searched 8 different databases with special keywords and found only 42 journals and 65 articles published between 1974 and 2015 on this subject. Moreover, since VRP is an np-hard problem, it cannot always be solved optimally, so, heuristics and meta-heuristics are often used. Performance comparison of these algorithms are generally performed on sample datasets with 100–200 nodes in the literature.

In this article, it is aimed to create the shortest routes that will satisfy all the needs and provide route optimization for two scenarios by considering the solid waste collection process of the Umraniye (which is a district of Istanbul, Turkey) as capacitated VRP by using a real dataset which includes almost three thousand nodes. Garbage trucks with a certain number and equal capacity start from a single center, collect the garbage on the streets of their own route and then return to

* Correspondence to: Istanbul Medeniyet Universitesi, Kuzey Kampus, Unalan, Uskudar, 34700 Istanbul, Turkey.
E-mail address: erkan.karabekmez@medeniyet.edu.tr (M.E. Karabekmez).

https://doi.org/10.1016/j.dajour.2022.100113
Received 28 June 2022; Received in revised form 25 July 2022; Accepted 21 August 2022
Available online 28 August 2022

the center every day. The constraints of garbage collection problem are: all locations must be visited, garbage on each street must be collected by only one truck (no double visits), and the truck's capacity must not be exceeded. The aim is to find routes that give the minimum distance throughout the entire process. Both heuristics and metaheuristics approaches like the nearest neighbor, savings, Christofides, simulated annealing, greedy descent, guided local search, and Tabu Search algorithms are applied by using OR-Tools and their performances are compared.

2. Literature review

This section consists of four parts. First, VRP literature, then the variants of VRP are summarized. Moreover, waste collection studies which are handled as VRP also examined. Finally, solution approaches are introduced.

2.1. Vehicle routing problem

VRP is an np-hard optimization problem that aims to find best route between the central depot and warehouses. There may be only one vehicle in the problem, or there may be more than one vehicle. Although the purpose of determining the best route is generally to find the shortest distance for the routes, a different problem can also be defined for different purposes such as minimizing the number of vehicle or duration. Usually, constraints of the problem are related to the vehicles, the customers, or the process. Due to these factors or constraints, several types of VRP have been introduced.

The first study on VRP is "Truck Dispatching Problem" written in 1959 [4]. The problem was introduced as generalized version of the Traveling Salesman Problem [5] and a solution was proposed to find the optimum routes of gasoline trucks. Howbeit no practical application has been made, the mathematical formulation that gives a near-optimal solution has been developed.

Clarke and Wright [6]developed the heuristic Savings Algorithm to solve this problem. Whilst Dantzig and Ramser fill the truck for the minimum distance in their studies, in this method it is filled for the closest capacity. It is stated that proposed method gives better results according to the tests made in this study.

Toth and Vigo [7], examined the best branch and bound algorithms during the written period for the capacitated vehicle routing problem (CVRP) with symmetric and asymmetric cost/distance matrix. Additionally, relaxation methods such as k-tree [8], b-matching [9] and set-partitioning [10] are also discussed. After this relaxation process, the problem size was increased from 25 nodes to 100 nodes and more.

Ralphs et al. [11] engaged routing and packing concepts using capacity constraints of integer programming and proposed a decomposition-based branch and cut algorithm for the CVRP. When common procedures fall down to decompose the nodes, the problem decomposed by converting it to n-tsp tours. Suggested theorems are applied whether capacity constraint fitted or not. They present a decomposition structure that can be adaptable to other combinatorial optimization problems. It was stated that the application results were promising, and it could solve most of the problems faced during that period. Since then, many novel studies have been made based on this study.

Baker and Ayechew [12], suggested a hybrid solution that consist of a genetic algorithm [13] and a local search algorithm to solve the VRP by limiting vehicle capacity and distance traveled. They proved that the suggested method gave competitive results in terms of solution time and quality with Tabu Search [14] and Simulated Annealing [15] which are shown as the most successful heuristic algorithms of the period.

Amous et al. [16] discussed CVRP using Variable Neighbor Search algorithm (VNS) [17], which aims to minimize the total distance traveled. Furthermore, proposed method includes many different search algorithms together with Variable Neighborhood Descent algorithm [18].

Borčinová [19] proposed two different linear programming models for the CVRP. The first of the proposed methods is flow-based linear programming, and the second is the updated assignment formulation. A mixed linear programming model based on the saving algorithm is proposed and it is stated that both methods give optimal results in small-scale problems. This study differs from the studies on the CVRP in terms of the method of eliminating sub-types.

Comert et al. [20] approached the CVRP solution in a hierarchical manner and transformed it into a clustering problem and then a routing problem (Cluster first, route second approach). After clustering with k-means, k-medoids [21] and random clustering algorithm, the routing problem of each cluster is solved with the branch and bound algorithm. Also, an application was carried out for a supermarket chain in the study, and it was understood that the clustering method that gave the best results was k-medoids. It is claimed that the proposed method will reduce the run times of large-scale CVRPs.

Goli et al. [22] proposed a solution for the CVRP with the artificial intelligence supported cuckoo algorithm, which considers customer satisfaction as well as reducing costs. The simulated annealing algorithm is used to speed up the algorithm and improve the solution quality. In the study, first the optimal solution was found with GAMS (General Algebraic Modeling System, a modeling software to solve mathematical programming and optimization problems), then the solution was produced with the classical cuckoo algorithm [23]. By comparing the results, it has been observed that the proposed method gives near-optimal results in a shorter time.

Ibrahim et al. [24] searched a solution for CVRP by using Column Generation [25], Google OR-Tools [26] and Reinforcement Learning algorithms [27] on a small-scale data set and compared the results. Furthermore, Reinforcement Learning and Google OR-Tools algorithms were run on a large-scale data set and their competencies in large problems were compared. Hereby, although the Reinforcement Learning method gave relatively better results, OR-Tools gave much better results in terms of solution generation speed.

Yazgan et al. [28] used the Savings Algorithm and a local search heuristic for solving the CVRP in a paint factory. The results of the two methods were compared with the actual data of the factory and it was stated that the local search heuristic algorithm gave the best results.

İlhan [29] proposed a new population-based simulated annealing algorithm to solve CVRP. In the proposed method, three route development operators, namely displacement, addition, and reversal, were used randomly in each iteration. It was stated that the method used on 63 frequently used data sets, gave optimal results in 23 samples, and gave more balanced results in the remaining 40 data sets compared to other methods.

Rabbouch et al. [30] proposed a new and dynamic algorithm, the Empirical Simulated Annealing algorithm, which will improve the possible weaknesses of the simulated annealing algorithm. The proposed method incrementally uses the last of the worst of reasonable solutions to update the acceptance criterion of the standard simulated annealing algorithm. It is stated that with this method, the randomness of standard simulated annealing is reduced, its convergence is improved, and a good performance is achieved in overall.

Hottung and Tierney [31] stated that the methods approaching VRP with machine learning are behind the standard optimization methods and they proposed a new Local Neighbor Search (LNS) algorithm to close this gap. The proposed method is based on deep neural networks. It has been stated that the proposed new LNS algorithm gives good results compared to the standard LNS and the Improved Lin Kernighan algorithm [32], according to the tests performed.

Sitek et al. [33] used a combination of VRP variants to solve the postal and courier delivery problem. A hybrid approach consisting of constraint programming, genetic algorithm and mathematical modeling methods was applied. It has been stated that the effectiveness of the model and the approach has been proven by the experiments.

2.2. Variants of the VRP

The most basic, classic VRP is that, as in Dantzig and Ramser's [4] studies, a certain number of vehicles start from the main depot and visit all the depots and finally complete their route by returning to the main depot. It is aimed to determine these routes with minimum cost (distance). When the capacity constraint is added to this problem, it is ensured that the vehicles do not exceed their capacities and the problem is called the Capacitated Vehicle Routing Problem (CVRP).

Problems that customers demand in different time zones; hence tours are performed according to time intervals are called VRP with Time Windows [34]. It is called VRP with Stochastic Demand [35] when the customer's demand is uncertain. Moreover, cases that more than one vehicle meets a customer demand instead of a single vehicle piece by piece with different voyages, are called The Split Delivery VRP [36]. Lastly, in cases where customer demands are met in different periods but with single vehicle, it is entitled as Periodic VRP [37]. There are also diverse types of VRPs for similar special cases.

2.3. Waste collection as a VRP

Han and Cueto [3] analyzed different techniques used for routing of wastes classified as domestic, commercial, and industrial. The study can be considered as a roadmap for similar problems in terms of summarizing and classifying the literature. While garbage collection and route optimization process, they stated that environmental impacts were not considered sufficiently and suggested that not only economic but also environmental impacts should be added to the objectives of CVRP.

Assaf and Saleh [38] studied the solid waste collection problem for the Palestinian city of Nablus as VRP and proposed a genetic algorithm-based solution. The aim of the study is minimizing the distance traveled by the garbage trucks and reducing the costs. It has been shown that the distance traveled can be reduced by 66% and the garbage collection time can be reduced from 7 h to 2.3 h.

Akhtar et al. [39] proposed a solution to CVRP with the Backtracking Search algorithm [40] to optimize garbage collection routes with the concept of a smart dustbin. To reduce the distance traveled and to optimize the garbage disposal frequency, the threshold value of the garbage cans was analyzed. It has been shown that the total distance and total fuel cost can be reduced by approximately 37% and 50% respectively.

Erfani et al. [41] solved CVRP, the collection and storage problem of garbage containers in Ahmadabad, Iran, with a Geographical Information System (GIS). It has been stated that the existing garbage collection system is deficient, the number of tours can be decreased by 12.5%, the total number of personnel can be reduced by 41%, and the total distance traveled (the main goal) can be improved by 53%.

Kuo and Zulvia [42] proposed a solution for CVRP using a hybrid algorithm consisting of a combination of Genetic Algorithm and Ant Colony Optimization algorithm [43] and 2 different local search algorithms (Prim's search algorithm [44] and 2-opt search algorithm [45]). It is stated that according to eight different performance tests the proposed method gave competitive results. In addition, a fuzzy demand garbage collection application studied for a city in Indonesia. It also stated that results are promising.

Hannan et al. [46] proposed the Modified Particle Swarm Optimization (PSO) algorithm for the solution of CVRP, and applied concepts including threshold waste level and scheduling for different datasets. It was emphasized that the method, which aims to find the optimal route by minimizing the distance traveled, gives good results in terms of waste collection and route optimization, and can be a significant tool in terms of socioeconomic and environmental effects in waste collection systems.

Asefi et al. [47] created a mixed integer linear programming model (MILP) for solid waste collection, routing, and cost reduction. Two different heuristics are proposed for the solution, in which the Variable Neighbor Search algorithm works alone, and the second: Variable Neighbor Search algorithm and Simulated Annealing algorithm are used together. It is stated that the two proposed methods give near-optimal results.

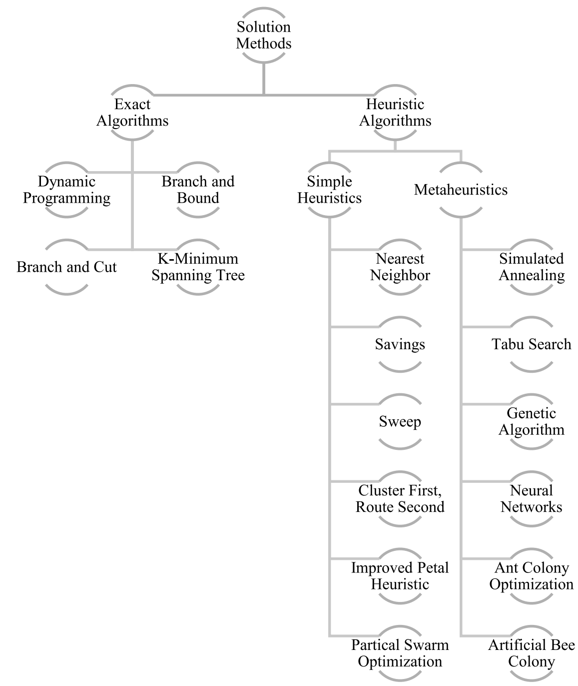

Fig. 1. VRP Solution Methods.
Source: Adapted from [49,50].

Vu et al. [48] modeled the garbage collection system in Austin, Texas, based on data forecasted for 2023. The relationship between parameters such as garbage collection frequency, garbage collection type, garbage truck type and garbage tour were examined using GIS with 48 different scenarios. It was indicated that the total garbage collection time could be improved by increasing the frequency of garbage collection.

2.4. VRP solution approaches

Since VRP is an NP-hard optimization problem, there is no method that consistently gives the optimal result. Particularly as the scale of the problem grows, methods or algorithms that give precise results cannot provide results within a reasonable time or even produce no results. For this reason, heuristic methods have been developed, aiming to give near-optimal result and to produce fast solutions. With the spread of heuristic algorithms, meta-heuristic algorithms have also started to be developed.

VRP solution methods that are frequently studied in the literature are diverse (Fig. 1). Exact methods consist of dynamic programming, which uses mathematical models, and algorithms [7–9,11] that aim to stretch the models. Heuristic algorithms can be divided into simple heuristics and metaheuristics. Simple metaheuristics are problem-dependent, and metaheuristics are problem-independent methods that try to maintain high quality solutions with shorter processing time in complex optimization problems [51]. Simple heuristics usually aim to generate a feasible solution and then try to improve it. Savings [6], Sweep [52], Christofides [53], Nearest Neighbor, Improved Petal [54] algorithms are some of the most used simple heuristics. On the other hand, metaheuristics effectively search the solution space by improving simple heuristics and using different algorithms together [55]. For these algorithms it is crucial to designed as adaptable to different types of optimization problems. Simulated Annealing [15], Tabu Search [14], Genetic Algorithm [13], Neural Networks [56], Ant Colony Optimization [43] and Artificial Bee Colony [57] algorithms are some of the frequently used metaheuristics.

Table 1
Format of route data.

DATE	TRUCK ID	NAME	TRUCK REGION	1ST TOUR TONNAGE	2ND TOUR TONNAGE	3RD TOUR TONNAGE	TOTAL TONNAGE	NUM OF TOUR
…	…	…	…	…	…	…	…	…

Table 2
Descriptive statistics of route data (shortened).

DATE	ATAKENT	ESENEVLER	İSTİKLAL - NAMIKKEMAL	MEHMET AKIF	NECİP FAZIL
"Example Date"	"Corresponding Tonnage Data"	"Corresponding Tonnage Data"	"Corresponding Tonnage Data"	"Corresponding Tonnage Data"	"Corresponding Tonnage Data"
Avg. Tonnage	14.768	16.779	14.525	16.445	17.799
Max Tonnage	17.760	19.500	16.590	20.080	20.440
Min Tonnage	11.800	14.080	12.000	12.820	15.120
Std. Dev.	1.285	1.037	919	1.432	1.039

3. Methodology

3.1. Dataset

Information about garbage collection process that include truck routes, tonnage data of neighborhoods and the number of containers on each street was obtained from Umraniye Municipality. The total length of streets, number of containers and tonnage data, truck region and number of daily tours between June 1, 2020, and June 30, 2021 for each neighborhood were combined in a single data table. This table contains 3327 lines of data. Truck region data indicated that which truck collects the garbage of related street in the current situation.

Municipality currently divides the district into 36 regions/routes. Three of these routes are main arteries and unlike remaining routes, garbage of them collected constantly during the day. Besides, another one of the 36 routes is industrial zone. Its tonnage data is not recorded. Therefore, these four routes excluded from the dataset. This means that the problem is handled with thirty-two routes (32 vehicles).

When the route information data table is examined, it has been observed that some streets are registered in two different neighborhoods, so there is a double counting of the number of locations and containers. Hence, a new list was created by omitting repetitive streets to eliminate that double counting error. With this new street list, the routing and optimization problem has 2998 locations and 6233 containers.

Tonnage data information table taken from municipality contains 14256 lines of data. Each row represents the tonnage data of a route on a certain date in Table 1. It shows the information about how many kilos of garbage of each route was collected with how many tours, by which vehicle and by which driver for each day between June 1, 2020, and June 30, 2021.

Sometimes trucks may not collect garbage on streets due to unplanned obstacles like wedding or funeral ceremonies, or road maintenance works. These exceptional undone collections are postponed to the next day. Which brings outliers to the data and those need to be removed from the dataset. Thus, interquartile ranges (IQR) were determined for each route and lower and upper bounds were found. The data points that reside out of $Q_1 - 1.5$ IQR < … < $Q_3 + 1.5$ IQR range were flagged as outliers and excluded from the rest of the analysis.

Table 1 has been rearranged so that each region is placed in an individual column and descriptive statistics of the data remaining after the exclusion of outliers are shown in Table 2. It contains average, maximum, and minimum tonnages for each region. These maximum tonnage values will be used later in the maximum expected daily amount of garbage for each container concept and calculations section.

3.2. Optimization problem

The linear programming model for classical CVRP is as follows [58]. N cluster consists of n nodes and nodes represent streets. Each street has q kilos of garbage demand. The distance between streets is expressed in c_{ij}. K represents the number of vehicles and Q represents the vehicle capacity.

$$\min \sum_{i=0}^{N} \sum_{j=0}^{N} \sum_{k=1}^{K} c_{ij} X_{ij}^{k} \tag{1}$$

In Eq. (1), the objective function aims to minimize the total distance traveled by vehicles. X_{ij}^{k} expresses whether the vehicle k traveled from node i to node j and it takes the value 0 or 1.

Constraints:

$$\sum_{k=1}^{K} \sum_{i=0}^{N} X_{ij}^{k} = 1 \ j \in \{1, \ldots, N\} : i \neq j \tag{2}$$

$$\sum_{k=1}^{K} \sum_{j=0}^{N} X_{ij}^{k} = 1 \ j \in \{1, \ldots, N\} : i \neq j \tag{3}$$

$$\sum_{i=0}^{N} \sum_{j=0}^{N} X_{ij}^{k} q_i \leq Q \ k \in \{1, \ldots, K\} \tag{4}$$

$$\sum_{j=1}^{N} X_{ij}^{k} \leq 1 \ for \ i = 0 \ k\epsilon\{1, \ldots, K\} \tag{5}$$

$$\sum_{j=1}^{N} X_{ji}^{k} \leq 1 \ for \ i = 0 \ k\epsilon\{1, \ldots, K\} \tag{6}$$

$$\sum_{j=1}^{N} X_{ij}^{k} = \sum_{j=1}^{N} X_{ji}^{k} \ for \ i = 0 \ k\epsilon\{1, \ldots, K\} \tag{7}$$

Eqs. (2)–(3) constraint that each street can be visited by only one vehicle. Eq. (4) ensures that the vehicle capacity is not exceeded. Eq 5–7. provide each tour starts and ends at warehouse, node 0.

3.3. Tools and softwares

Python software was used to find street coordinates and run all other algorithms. Python libraries used in the study are:

(i) Geopy [59] is a geographic information system library included in Python. With this library, the exact location, and coordinates of all streets whose address are available in the dataset were identified. This process is also known as "geocoding". Likewise, the geodesic distance (the length of the curve that indicates the shortest distance between two surface points) between all of the coordinate couples were calculated. The geocoders method in the geopy library collects data by connecting to a geocoding service. The coordinates and locations of all streets were determined by accessing the Yandex Maps [60] database as a geocoding service.

(ii) OR-Tools are optimization tools that allows executing heuristics and meta-heuristics to solve the VRP and developed by Google [61]. In this paper, CVRP is solved by heuristics and meta-heuristics algorithms,

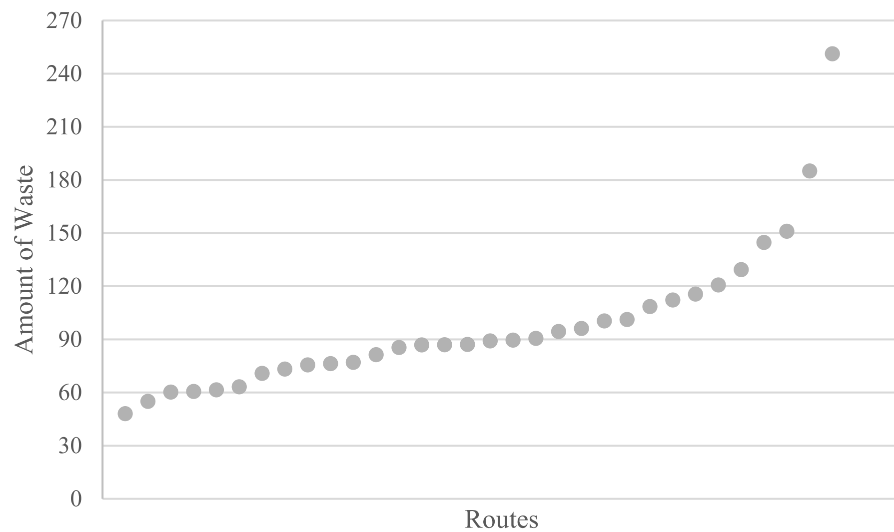

Fig. 2. Maximum expected daily amount of waste for each container (MEDAG) (kg/day).

Table 3
OR-Tools options.

First solution strategy	Local search options
Path cheapest arc	Greedy descent
Path most constrained arc	Guided local search
Christofides	Simulated annealing
Savings	Tabu search

respectively. It takes coordinates, demands (kg), distance matrix, number and capacities of vehicles, id of the main warehouse as inputs and creates the data model. Then, it takes the first solution strategy, local search option, and time limit inputs, executes, and generates a solution.

While first solution strategy ensures that OR-Tools creates a feasible solution. Local search option tries to improve initial solution created with first solution strategy. First solution strategies and local search options used in this paper are shown in Table 3. Time limit refers to the maximum time for algorithm can search for each node.

3.4. Data analysis

When a query, for instance, "PırıltıSokak, Ümraniye" is sent to Geopy, the output is exact address "PırıltıSok., Elmalıkent Mah., Ümraniye, İstanbul, Turkey" and coordinate of the street "(29.092866, 41.039532)". With this method, the exact locations, and coordinates of all streets within the boundary of Umraniye district were found. The output list has been created and a new column, container numbers that taken from dataset, was appended.

There are some streets that are not populated enough to require a garbage container or not wide enough for trucks to enter, but these streets also produce waste. Therefore, trucks collect on road waste even though there are no containers. Thus, this situation should be integrated into the problem. For this purpose, the container number of the streets with zero containers is edited as 0.2. The amount of garbage will be determined by multiplying container amounts with street coefficients that will be calculated later sections.

Maximum expected daily amount of garbage for each container (MEDAG) was determined as the maximum daily collected waste in the route divided by number of containers in the route in the dataset after removal of outliers.

Waste production across the district is not homogeneous. In order to capture this heterogeneity MEDAGs were plotted (Fig. 2). Except the two routes on the right side of the graph, all routes stacked within a certain range. Serving all routes with a single vehicle may cause overload on the two routes that are the outside of the range.

This overload may cause breakdowns on vehicles, or it can decrease workers' performances. Therefore, two scenarios were proposed to address this problem. In the first scenario the demand of the containers on the overloaded routes were divided by two to ensure al MEDAGs are below 151 kg/day (Fig. 3).

In the latter, citizens' satisfaction was taken into account more precisely by preventing garbage accumulation in the containers. A narrower range for the MEDAGs was attained by serving to the routes with different number of trips per day. Proposed scenario 2 achieve a range of 35 to 76 kg/day for all containers by visiting them up to four times a day (Fig. 4).

The implantation stages of the method for both scenarios are as follows:

- The modified MEDAGs are distributed to all streets in the problem. Street garbage amounts are calculated by multiplying the number of containers on the street with the corresponding MEDAG. All values are rounded to an integer so that OR-Tools works efficiently.
- Previously determined (with geopy) coordinates of the streets are given as input to the OR-Tools as an [nx1] matrix. Previously calculated maximum expected amounts of street garbage are also given as input to the OR-Tools as an [nx1] demand matrix.
- The [nxn] distance matrix consists of rounded geodesic distances of the locations to integers in meters is also given as input to OR-Tools. Vehicle capacities were calculated by dividing total demand in single tour by the number of vehicles (32) for each scenario to achieve 100% vehicle usage.
- The number (32) and capacities (Scenario 1: 18 tons and Scenario 2: 10.5 tons) of the vehicles are given as another input to OR-Tools. These steps are the basic inputs of the OR-Tools to execute the CVRP.
- Moreover, there are operational inputs such as first solution strategy, local search option and time limit. All combinations of the first solution strategies and local search options shown in Table 3 were applied for each scenario to find best algorithm pair. Also, trials with three different time limits for local search 100, 500, 1000 s were conducted for each scenario and each algorithm pair.
- All identified routes were plotted on the district map by using open street map [62].

The workflow of the study is visually summarized in Fig. 5.

4. Result and discussion

4.1. Scenario 1: Optimal solution approach

For each of the 2998 streets in the district, demands were calculated by multiplying values for maximum expected daily amount of garbage

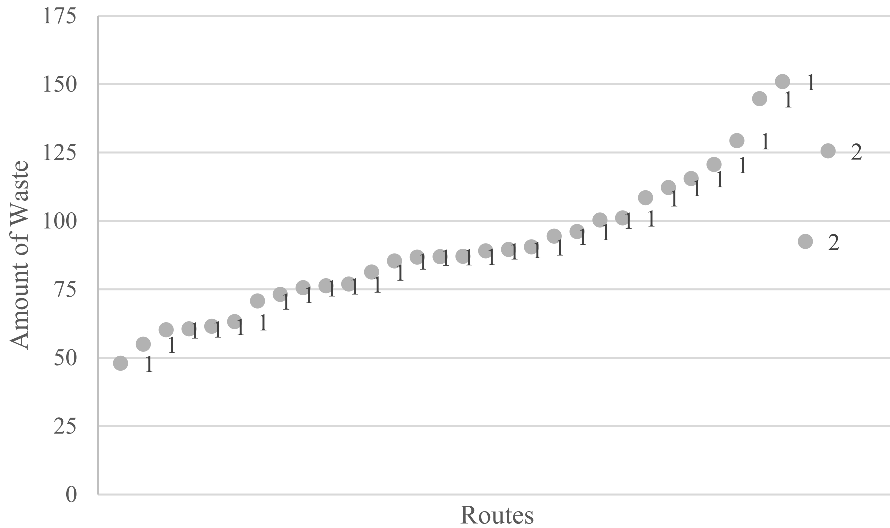

Fig. 3. Modified maximum expected daily amount of waste for each container (MEDAG-1) (kg/day) – Optimal Solution Approach – Scenario 1: Data labels (1 or 2) indicate number of visits to a container in the route.

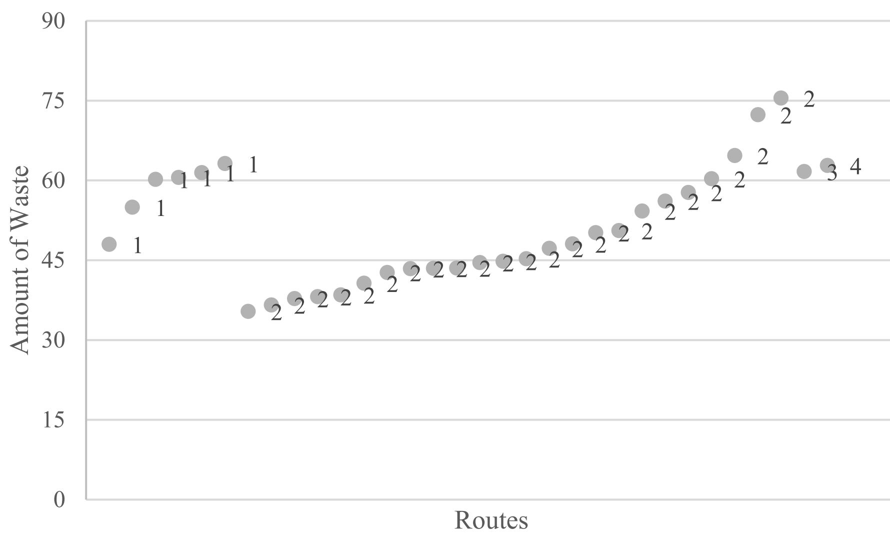

Fig. 4. Modified maximum expected daily amount of waste for each container (MEDAG-2) (kg/day) – Satisfaction Focused Approach – Scenario 2: Data labels (1 to 4) indicate number of visits to a container in the route.

Table 4
Street demand table: Scenario 1.

ID	Location	Latitude	Longitude	Number of container (NC)	MEDAG	Number of tours	Street demand
0	HekimbaşıKatıAtık Aktarma Merkezi	41.05846964	29.11273772	0	0	0	0
1	3016. Sok., Esenevler Mah., Ümraniye, İstanbul, Türkiye	41.008843	29.097232	0.2	112	1	22
2	Neval Sok., Necip Fazıl Mah., Ümraniye, İstanbul, Türkiye	41.015022	29.164866	12	251	2	1506
3	Acun Sok., Ümraniye, İstanbul, Türkiye	41.008673	29.130559	10	63	1	630
…	…	…	…	…	…	…	…

for each container (MEDAG-1) in Fig. 3 with the number of containers (NC) in the street. If there is no container in the street NC was set as 0.2 to ensure inclusion of the street in the problem. The resulting data table is in the form of Table 4 and the full table is available in Table S1.

Location, Latitude and Longitude columns in Table 4 are automatically obtained from Yandex Maps. MEDAG column is calculated and taken from Fig. 2. The number of tours column is taken from Fig. 3. The street demand is calculated with the formula NC * MEDAG/(Number of Tours) and rounded to the nearest integer.

Distance matrix between streets (Table 5), demand matrix of locations (Table 6), number and capacity of vehicles, and warehouse id are the basic inputs of OR Tools. Table 5 is a 2998x2998 matrix calculated with Geopy using the Latitude and Longitude columns of Table 4. Cell values states the geodesic distance between two locations in meters.

The demand matrix (Table 6) is the street demand column of Table 4. Cell values states the daily amount of garbage in the streets in kg.

Warehouse id is 0. All vehicles start and finish the tour from the warehouse. The number of vehicles is 32. Vehicle capacity was

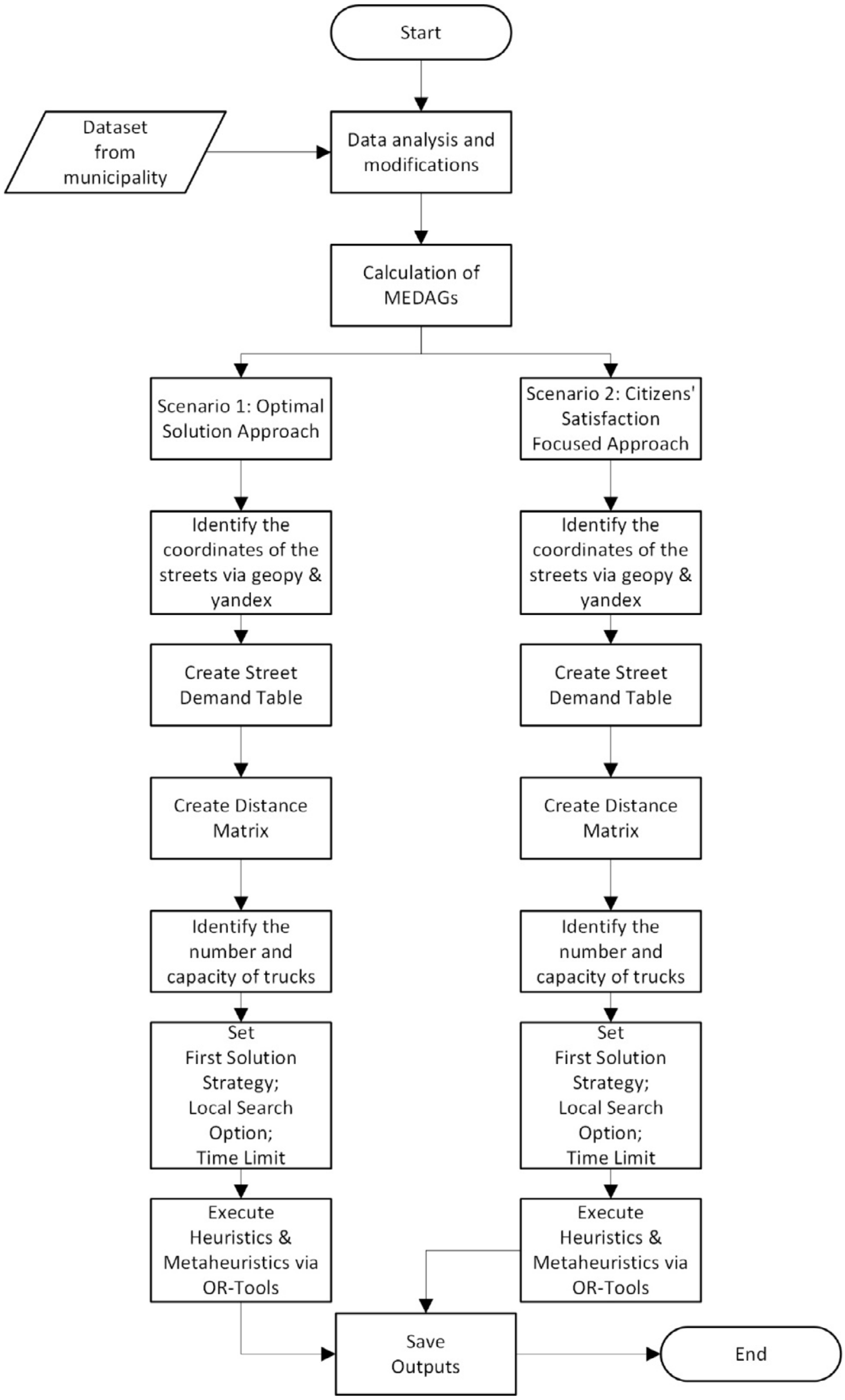

Fig. 5. Overall flowchart of the study.

Table 5
Distance matrix of locations: Scenario 1.

	0	1	2	…	…	…	2997
0	0	6399	6747	…	…	…	6485
1	6399	0	2104	…	…	…	335
2	6747	2104	0	…	…	…	2431
…	…	…	…	0	…	…	…
…	…	…	…	…	0	…	…
…	…	…	…	…	…	0	…
2997	6485	335	2431	…	…	…	0

Table 6
Demand matrix: Scenario 1.

Id	0	1	2	3	…	…	…	2997
Demand	0	600	426	1672	…	…	…	20

calculated as 18 Tons by dividing total demand in single tour by the number of vehicles to guarantee the deployment of all vehicles. The capacity for all vehicles was assumed to be the same.

For all 16 first solution strategy and local search option algorithm pairs and three different time spent in search (100, 500 and 1000 s)

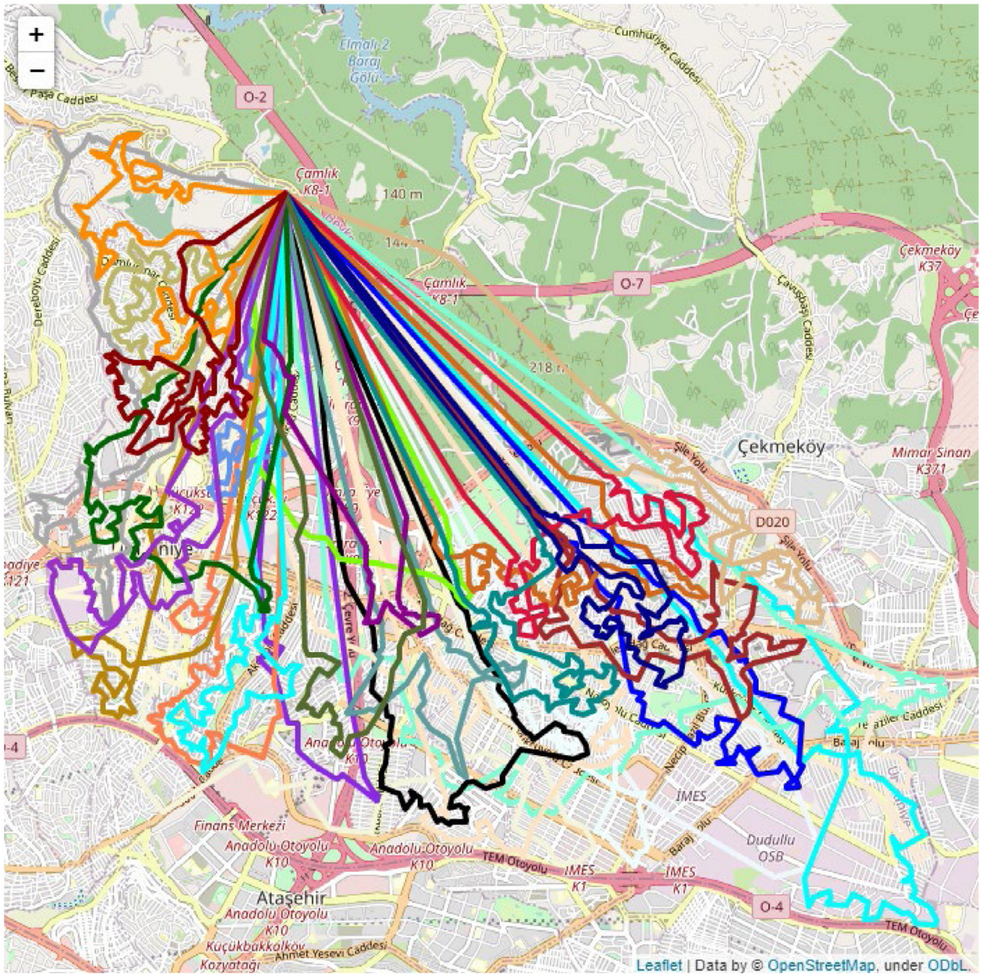

Fig. 6. The shortest routes of 32 vehicles for Scenario1 — Map Data by © [62]. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 7
Results: Scenario 1.

First solutions strategy	Local search options	Objective Function (m)	Operation time (s)
Path Cheapest Arc	Guided Local Search	525.582	640
Path Cheapest Arc	**Simulated Annealing**	**525.443**	**634**
Path Cheapest Arc	**Tabu Search**	**525.443**	**621**
Path Cheapest Arc	Greedy Descent	525.471	630
Savings	Guided Local Search	571.122	677
Savings	Simulated Annealing	571.352	646
Savings	Tabu Search	571.389	676
Savings	Greedy Descent	571.122	604
Christofides	Guided Local Search	558.127	645
Christofides	Simulated Annealing	558.522	662
Christofides	Tabu Search	558.522	657
Christofides	Greedy Descent	558.127	675
Path Most Constrained Arc	Guided Local Search	535.856	665
Path Most Constrained Arc	Simulated Annealing	534.379	629
Path Most Constrained Arc	Tabu Search	534.471	627
Path Most Constrained Arc	Greedy Descent	534.757	640

parameters trials were conducted by using the prepared inputs. When the time spent in search option set to 100 s, OR Tools could not bring any feasible solution. On the other hand, setting the parameter to 500 s or 1000 s do not bring any significant change in the results. Therefore, the results were listed with time limit option of 500 s (Table 7).

In this scenario the combinations of Path Cheapest Arc algorithm as the first solution strategy and Tabu Search or Simulated Annealing algorithms as local search tool gives the minimum objective. However, reckon with the solution time too, it can be said that Tabu Search algorithm gives partially better results. 100% vehicle occupancy rate was achieved by using all 32 vehicles in all experiments and the shortest route length for the problem was determined to be 525,443 m. The route map of all vehicles for the best solution is shown in Fig. 6. Each route is shown in a different color and Hekimbasi Solid Waste Collection Center (main warehouse) is shown with a black star. All tours start and end here.

A sample route and output of a particular vehicle that obtained with OR-Tools is as follows: The route of the Vehicle 8 (randomly selected) that acquired from the best experiment is shown in Fig. 7. The pins show the locations visited, and numbers indicate the order of visits.

“Route for vehicle 8:

*0 Load(0) -> 1297 Load(18) -> 1001 Load(37) -> 806 Load(56) -> 1930 Load(74) -> 1727 Load(92) -> 2476 Load(272) -> 1664 Load(290) -> 1980 Load(380) -> 1525 Load(398) -> … (*output has been shortened). -> 1735 Load(17954) -> 2672 Load(17972) -> 0 Load(17972)*

Distance of the route: 12960 m

Load of the route: 17972 kg”

This sample output explains that the route starts from position 0, the vehicle goes to location 1297 and collects 18 kilos of garbage, then goes to location 1001, then collects another 19 kilos of garbage and the vehicle capacity is 37 kilos. When the vehicle went to the 2672nd location, the vehicle capacity was 17,972 kilos and since the capacity was full, it returned to the starting position and finished the tour. The total distance of the route is 12,960 m, and the total amount of garbage collected on the route is 17,972 kg.

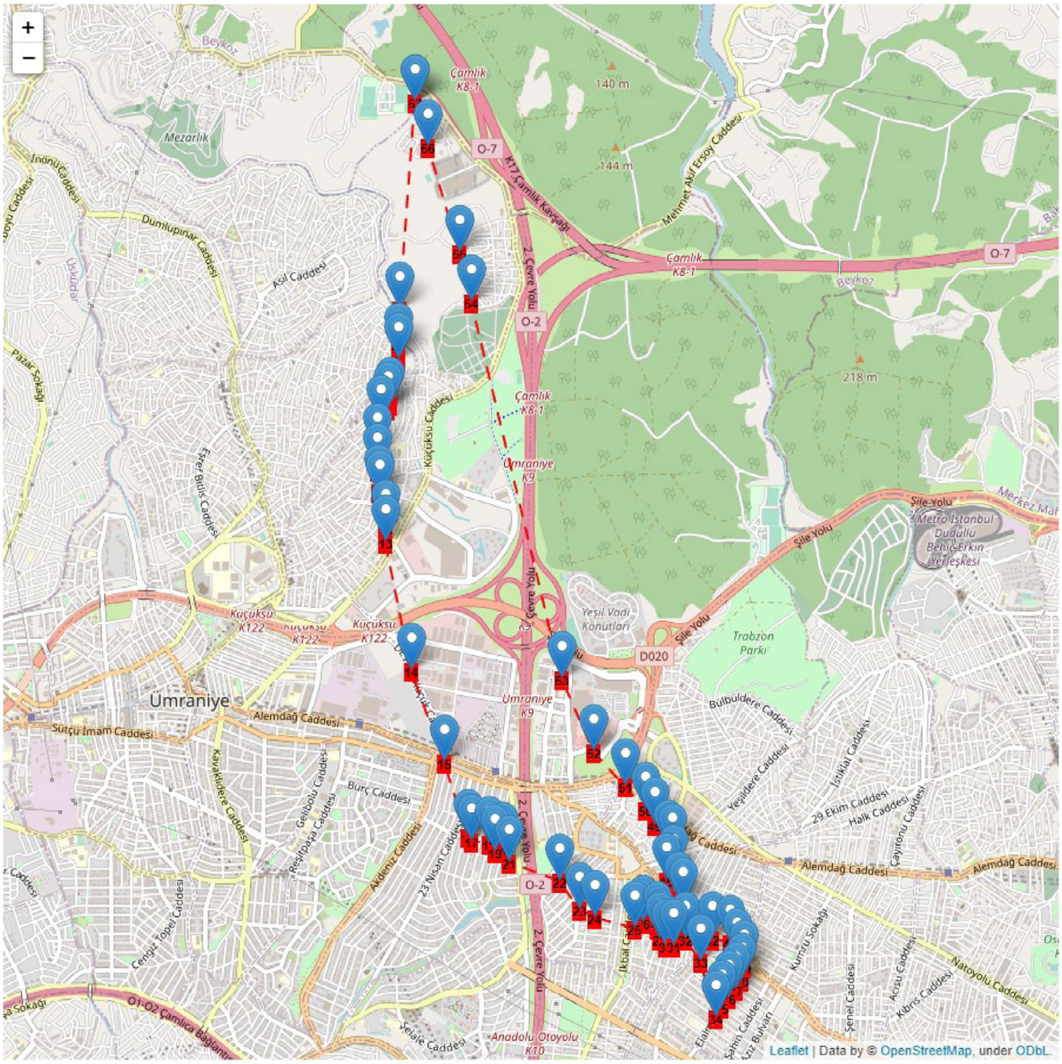

Fig. 7. The shortest route for Vehicle 8 in Scenario1 — Map Data by © [62].

Table 8
Street demand table: Satisfaction Scenario 2.

ID	Location	Latitude	Longitude	Number of container (NC)	MEDAG	Number of tours	Street demand
0	HekimbaşıKatıAtık Aktarma Merkezi	41.05846964	29.11273772	0	0	0	0
1	3016. Sok., Esenevler Mah., ˋUmraniye, İstanbul, T́urkiye	41.008843	29.097232	0.2	112	2	11
2	Neval Sok., Necip Fazıl Mah., ˋUmraniye, İstanbul, T́urkiye	41.015022	29.164866	12	251	4	753
3	Acun Sok., ˋUmraniye, İstanbul, T́urkiye	41.008673	29.130559	10	63	1	630
4	Asil Cad., TopağacıMah., ˋUmraniye, İstanbul, T́urkiye	41.049326	29.105775	3	185	3	185
…	…	…	…	…	…	…	…

4.2. Scenario 2: Citizens' satisfaction focused optimal solution approach

For each of the 2998 streets in the district, demands were calculated by multiplying values for maximum expected daily amount of garbage for each container (MEDAG-2) in Fig. 4 with the number of containers (NC) in the street. If there is no container in the street NC was set as 0.2 to ensure inclusion of the street in the problem. The resulting data table is in the form of Table 4 and the full table is available in Table S2.

Location, Latitude and Longitude columns in Table 8 are automatically obtained from Yandex Maps. MEDAG column is calculated and taken from Fig. 2. The number of tours column is taken from Fig. 4. The street demand is calculated with the formula NC * MEDAG/(Number of Tours) and rounded to the nearest integer.

The distance matrix, the demand matrix of locations (Table 9), number and capacity of vehicles, and warehouse id are the basic inputs of OR Tools in scenario 2. Distance matrix between streets (Table 5) is the same for both scenarios. The demand matrix (Table 9) is the street demand column of Table 8. Cell values states the daily amount of garbage in the streets in kg.

Warehouse id and the number of vehicles are the same as scenario 1. Vehicle capacity was also calculated by the same calculations and found as 10.5 Tons for scenario 2 and the capacity for all vehicles was assumed to be the same again.

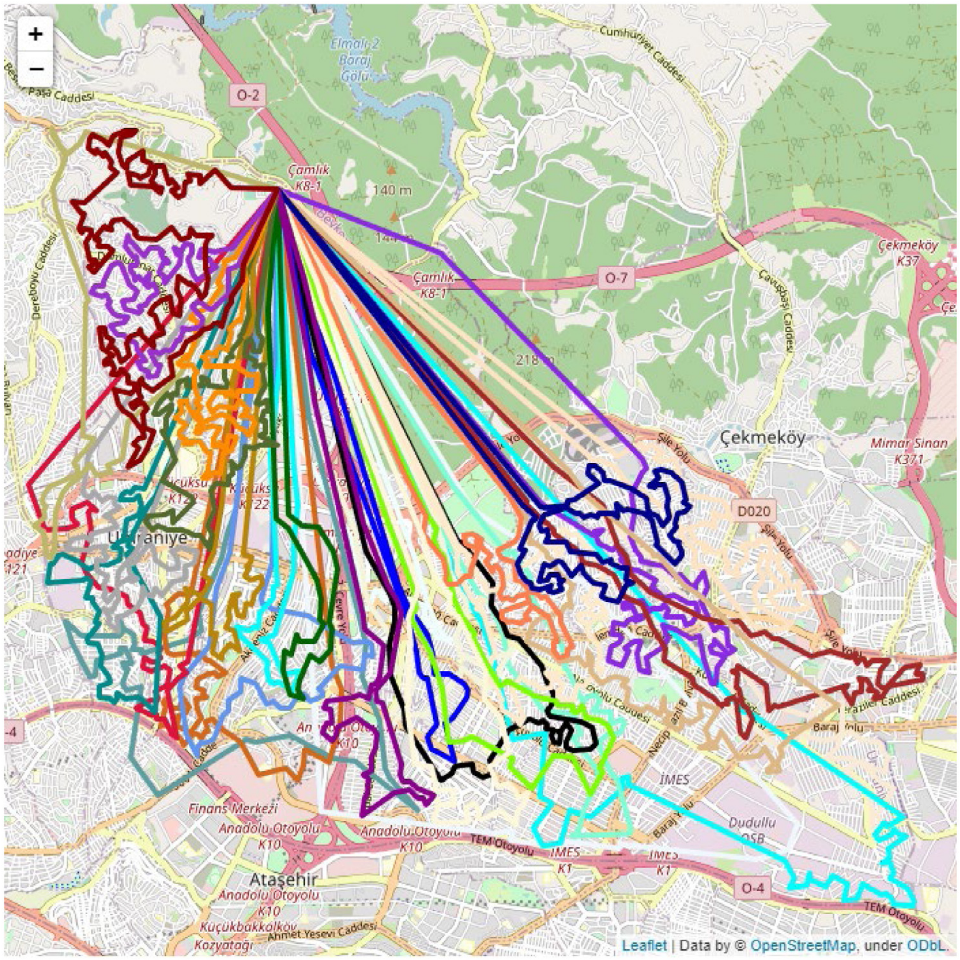

Fig. 8. The shortest Routes of 32 vehicles for Scenario 2 — Map Data by © [62]. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 9
Demand matrix: Satisfaction Scenario 2.

Id	0	1	2	3	…	…	…	2997
Demand	0	300	213	836	…	…	…	10

For all 16 first solution strategy and local search option algorithm pairs and three different time spent in search (100, 500 and 1000 s) parameters trials were conducted by using the prepared inputs for this scenario as well. When the time spent in search option set to 100 s, OR Tools could not bring any feasible solution again. Even for 500 s for some routes OR Tools could not bring any feasible solution. Therefore, the results were listed with time limit option of 1000 s in scenario 2 (Table 10).

As a result, Path Cheapest Arc algorithm gives the best result among all first solution strategies in scenario 2, too. When the first solution produced with this algorithm, and it was observed to be the most feasible to developed with either the Guided local Search or Greedy Descent algorithm. Although the both algorithm pairs bring the same value for the objective function, the combination of Path Cheapest Arc and Greedy Descent algorithm brings it faster. 100% vehicle occupancy rate was achieved by using all 32 vehicles in all experiments and the shortest route length for the problem was determined to be 533302 m. The route map of all vehicles for the best solution is shown in Fig. 8. Each route is shown in a different color and Hekimbasi Solid Waste Collection Center (main warehouse) is shown with a black star. All tours start and end there.

Table 10
Results: Satisfaction Scenario 2.

First solutions strategy	Local search options	Objective Function (m)	Operation Time (s)
Path Cheapest Arc	**Guided Local Search**	**533.302**	**1210**
Path Cheapest Arc	Simulated Annealing	533.716	1151
Path Cheapest Arc	Tabu Search	537.074	1218
Path Cheapest Arc	**Greedy Descent**	**533.302**	**1178**
Savings	Guided Local Search	573.573	1180
Savings	Simulated Annealing	573.568	1238
Savings	Tabu Search	573.527	1168
Savings	Greedy Descent	573.573	914
Christofides	Guided Local Search	553.680	1193
Christofides	Simulated Annealing	555.590	1206
Christofides	Tabu Search	555.590	1246
Christofides	Greedy Descent	553.680	1136
Path Most Constrained Arc	Guided Local Search	535.098	1155
Path Most Constrained Arc	Simulated Annealing	535.945	1202
Path Most Constrained Arc	Tabu Search	534.845	1173
Path Most Constrained Arc	Greedy Descent	534.698	1150

The sample route and output of a particular vehicle that obtained with OR-Tools is as follows: The route of the Vehicle 17 (randomly selected) that acquired from the best experiment is shown in Fig. 9. The pins show the locations visited, and numbers indicate the order of visits.

"Route for vehicle 17:

0 Load(0) -> 610 Load(12) -> 2066 Load(73) -> 293 Load(622) ->83 Load(634) -> 2103 Load(646) ->… (output has been shortened)->

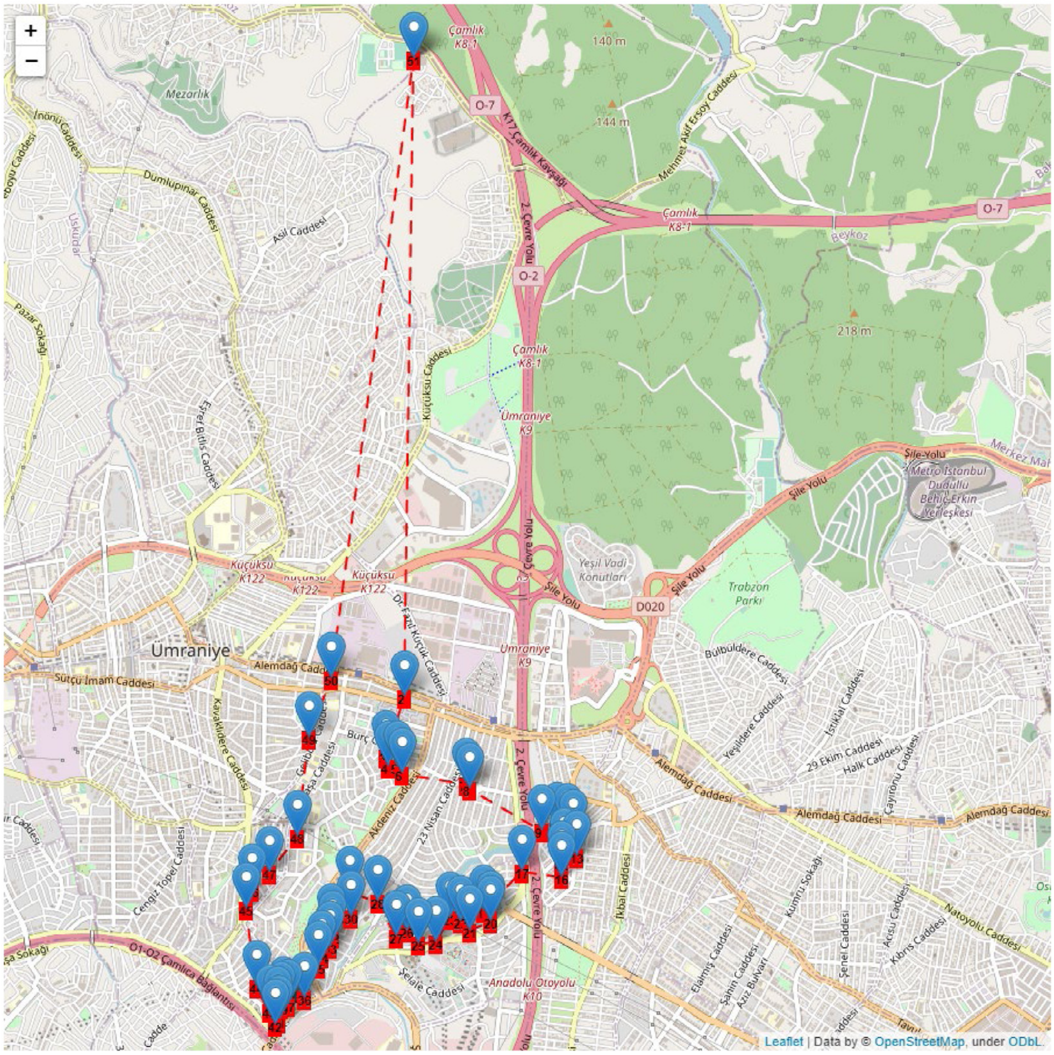

Fig. 9. The shortest route for Vehicle 17 in Scenario 2 — Map Data by © [62].

1605 Load(10468) -> 2164 Load(10480) -> 1764 Load(10488) -> 420 Load(10500) -> 0 Load(10500)

Distance of the route: 15144 m

Load of the route: 10500 kg"

This sample output explains that the route starts from position 0, the vehicle goes to location 610 and collects 12 kilos of garbage, then goes to location 2066, then collects another 61 kilos of garbage and the vehicle capacity is 73 kilos. When the vehicle went to the 420th location, the vehicle capacity was 10500 kilos and since the capacity was full, it returned to the starting position and finished the tour. The total distance of the route is 15144 m, and the total amount of garbage collected on the route is 10500 kg.

In order to validate the time parameter, we have created convergence curves for the best performing algorithm combinations (Fig. 10).

Convergence analysis shows that after 500 s the minimum total route distance (objective function) is assessed for both of the scenarios.

4.3. Discussion

As a result of the experiments for the first scenario shown in Table 7, we understand that the combination of algorithms that gives the best results is the Path Cheapest Arc and Tabu Search algorithms. The length of the shortest route is 525,443 m. 100% vehicle utilization rate is provided and the average distance per route is 16,420 m.

Then as a result of the experiments for the second scenario shown in Table 10, we understand that using the Cheapest Arc and Greedy Descent algorithms together gives the best results. The length of the shortest route of the scenario is 533,302 m. 100% vehicle utilization rate is provided and the average distance per route is 16,665 m.

5. Conclusion and future works

In this study, solid waste optimization and vehicle routing problem (VRP) were discussed together, and solid waste collection route optimization of Istanbul Umraniye district was aimed. The problem is considered as the Capacitated Vehicle Routing Problem and heuristic and meta-heuristic methods are used to solve this problem. It was supported by two different scenarios in which the number of tours was arranged according to the garbage production coefficients of the neighborhoods. The first scenario is a double tour to two neighborhoods, which are highly separated from other neighborhoods in terms of the degree of garbage production, and a single trip to all remaining neighborhoods. The second scenario is the citizen satisfaction scenario, which will provide a more homogeneous workload for garbage trucks and workers and prevent the overload of garbage in the containers.

In the execution phase, different Python libraries were used to run the algorithms. The algorithms were executed with the OR-Tools package, which is frequently used in the literature to compare the performance of heuristics and meta-heuristics.

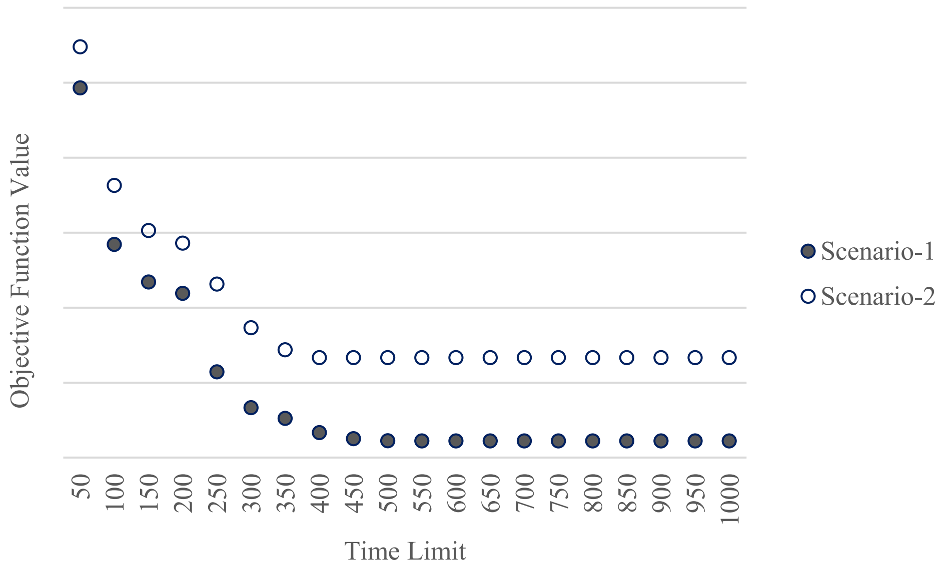

Fig. 10. Convergence curves for the best performing algorithm combinations.

Vu et al. [48] examined 48 different scenarios using GIS and stated that by increasing the truck capacity, the distance traveled could be reduced by up to 21%. In this study, increasing vehicle capacity of 10.5 tons in Scenario 2 to 18 tons in Scenario 1 decreases the path length by 13.5% similarly. Whereas here we commented that collecting garbage more frequently can improve citizens' experience with the municipality. Therefore, the increase in the distance from Scenario 1 to Scenario 2 can be regarded as the cost of marginal satisfaction. In a furthermore detailed study citizens' satisfaction can also be measured and added to the optimization problem in a more quantitative approach. The current study relying on a real case study with a fixed number of vehicles. That is why marginal utility of manipulating number of vehicles/routes is disregarded and focused on best route distribution for a given garbage collection demand.

The current application in the district's municipality and in many other municipalities for route identification is manual and arbitrary. Here we show that open-source and easy to implement software can be used to find optimal routings in considerable time and computation cost with common algorithms.

An important limitation of this study is that the distances between the streets are calculated as the geodetic distance. This weakness can be eliminated by using paid geocoding services by recalculating the distance matrix and rerunning the algorithms.

If smart technologies are added to the study, the fullness ratio of the garbage containers can be determined. Then, it may be possible to provide simultaneous route optimization instead of determining the number of tours according to historical data. Furthermore, garbage trucks can respond to unusual garbage production occurrences (outliers) instead of following usual routines.

Our results show that heuristic and meta-heuristic algorithms can give fast and feasible (near optimal) results even in large scale CVRPs. The method of the study can be used for any other district and provide feasibility. In an even more resource limited age waiting for us, feasibility of the allocations of the resources will remain an important challenge. The study will be a noteworthy contribution in this manner.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors would like to thank to R&D Center of Umraniye Municipality for providing data by "Tezim Ümraniye" Programme.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.dajour.2022.100113.

References

[1] U.S. Environmental Protection Agency, Waste definitions, 1993, https://www.epa.sa.gov.au/files/4771336_guide_waste_definitions.pdf (Accessed 14 Jun 2022).
[2] Istanbul Metropolitan Municipality, Atık Yönetimi, 2021, https://atikyonetimi.ibb.istanbul/ (Accessed 15 October 2021).
[3] H. Han, E.P. Cueto, Waste collection vehicle routing problem: Literature review, PROMET - Traffic Transp. 27 (4) (2015) 345–358, http://dx.doi.org/10.7307/ptt.v27i4.1616.
[4] G.B. Dantzig, J.H. Ramser, The truck dispatching problem, Manage. Sci. 6 (1) (1959) 80–91, http://dx.doi.org/10.1287/mnsc.6.1.80.
[5] M.M. Flood, The traveling-salesman problem, Oper. Res. 4 (1) (1956) 61–75, http://dx.doi.org/10.1287/opre.4.1.61.
[6] G. Clarke, J.W. Wright, Scheduling of vehicles from a central depot to a number of delivery points, Oper. Res. 12 (4) (1964) 568–581, http://dx.doi.org/10.1287/opre.12.4.568.
[7] P. Toth, D. Vigo, Branch-and-bound algorithms for the capacitated VRP, in: The Vehicle Routing Problem, Society for Industrial and Applied Mathematics, 2002, pp. 29–51, http://dx.doi.org/10.1137/1.9780898718515.ch2.
[8] M.L. Fisher, Optimal solution of vehicle routing problems using minimum K-trees, Oper. Res. 42 (4) (1994) 626–642, http://dx.doi.org/10.1287/opre.42.4.626.
[9] D.L. Miller, A matching based exact algorithm for capacitated vehicle routing problems, ORSA J. Comput. 7 (1) (1995) 1–9, http://dx.doi.org/10.1287/ijoc.7.1.1.
[10] E. Hadjiconstantinou, N. Christofides, A. Mingozzi, A new exact algorithm for the vehicle routing problem based onq-paths andk-shortest paths relaxations, Ann. Oper. Res. 61 (1) (1995) 21–43, http://dx.doi.org/10.1007/BF02098280.
[11] T.K. Ralphs, L. Kopman, W.R. Pulleyblank, L.E. Trotter, On the capacitated vehicle routing problem, Math. Program. 94 (2–3) (2003) 343–359, http://dx.doi.org/10.1007/s10107-002-0323-0.
[12] B.M. Baker, M.A. Ayechew, A genetic algorithm for the vehicle routing problem, Comput. Oper. Res. 30 (5) (2003) 787–800, http://dx.doi.org/10.1016/S0305-0548(02)00051-5.
[13] D.E. Goldberg, J.H. Holland, Genetic algorithms and machine learning, Mach. Learn. 3 (1988) 95–99, http://dx.doi.org/10.1023/A:1022602019183.
[14] F. Glover, Tabu search—Part I, ORSA J. Comput. 1 (3) (1989) 190–206, http://dx.doi.org/10.1287/ijoc.1.3.190.
[15] P.J.M. van Laarhoven, E.H.L. Aarts, Simulated annealing, in: Simulated Annealing: Theory and Applications, vol. 1987, Springer Netherlands, Dordrecht, 1987, pp. 7–15, http://dx.doi.org/10.1007/978-94-015-7744-1_2.

[16] M. Amous, S. Toumi, B. Jarboui, M. Eddaly, A variable neighborhood search algorithm for the capacitated vehicle routing problem, Electron. Notes Discrete Math. 58 (2017) 231–238, http://dx.doi.org/10.1016/j.endm.2017.03.030.
[17] N. Mladenović, P. Hansen, Variable neighborhood search, Comput. Oper. Res. 24 (11) (1997) 1097–1100, http://dx.doi.org/10.1016/S0305-0548(97)00031-2.
[18] A. Hertz, M. Mittaz, A variable neighborhood descent algorithm for the undirected capacitated arc routing problem, Transp. Sci. 35 (4) (2001) 425–434, http://dx.doi.org/10.1287/trsc.35.4.425.10431.
[19] Z. Borčinova, Two models of the capacitated vehicle routing problem, Croat. Oper. Res. Rev. 8 (2) (2017) 463–469, http://dx.doi.org/10.17535/crorr.2017.0029.
[20] S.E. Comert, H.R. Yazgan, S. Kır, F. Yener, A cluster first-route second approach for a capacitated vehicle routing problem: a case study, Int. J. Procure. Manag. 11 (4) (2018) 399, http://dx.doi.org/10.1504/IJPM.2018.092766.
[21] J.E. Gentle, L. Kaufman, P.J. Rousseuw, Finding groups in data: An introduction to cluster analysis, Biometrics 47 (2) (1991) 788, http://dx.doi.org/10.2307/2532178.
[22] A. Goli, A. Aazami, A. Jabbarzadeh, Accelerated cuckoo optimization algorithm for capacitated vehicle routing problem in competitive conditions, Int. J. Artif. Intell 16 (1) (2018) 88–112.
[23] X.-S. Yang, Suash Deb, Cuckoo search via Lévy flights, in: 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), 2009, pp. 210–214, http://dx.doi.org/10.1109/NABIC.2009.5393690.
[24] N. A. A. Ibrahim, Lo, J.A. Ishaya, Capacitated vehicle routing problem, Int. J. Res. -GRANTHAALAYAH 7 (3) (2019) 310–327, http://dx.doi.org/10.29121/granthaalayah.v7.i3.2019.976.
[25] L.H. Appelgren, A column generation algorithm for a ship scheduling problem, Transp. Sci. 3 (1) (1969) 53–68, http://dx.doi.org/10.1287/trsc.3.1.53.
[26] L. Perron, Operations research and constraint programming at google, 2–2, 2011, http://dx.doi.org/10.1007/978-3-642-23786-7_2.
[27] L.P. Kaelbling, M.L. Littman, A.W. Moore, Reinforcement learning: A survey, J. Artificial Intelligence Res. 4 (1996) 237–285, http://dx.doi.org/10.1613/jair.301.
[28] H.R. Yazgan, S.E. Cömert, E.K. Kılıç, Araç rotalama probleminin sezgisel algoritmalar ile çözülmesi: Bir boya fabrikasında uygulama, J. Turk. Oper. Manag. 4 (2) (2020) 549–563.
[29] İ İlhan, A population based simulated annealing algorithm for capacitated vehicle routing problem, Turk. J. Electr. Eng. Comput. Sci. 28 (3) (2020) 1217–1235, http://dx.doi.org/10.3906/elk-1902-122.
[30] B. Rabbouch, F. Saâdaoui, R. Mraihi, Empirical-type simulated annealing for solving the capacitated vehicle routing problem, J. Exp. Theor. Artif. Intell. 32 (3) (2020) 437–452, http://dx.doi.org/10.1080/0952813X.2019.1652356.
[31] A. Hottung, K. Tierney, Neural large neighborhood search for the capacitated vehicle routing problem, 2019, http://dx.doi.org/10.3233/FAIA200124.
[32] S. Lin, B.W. Kernighan, An effective heuristic algorithm for the traveling-salesman problem, Oper. Res. 21 (2) (1973) 498–516, http://dx.doi.org/10.1287/opre.21.2.498.
[33] P. Sitek, J. Wikarek, K. Rutczyńska-Wdowiak, G. Bocewicz, Z. Banaszak, Optimization of capacitated vehicle routing problem with alternative delivery, pick-up and time windows: A modified hybrid approach, Neurocomputing 423 (2021) 670–678, http://dx.doi.org/10.1016/j.neucom.2020.02.126.
[34] M. Desrochers, J. Desrosiers, M. Solomon, A new optimization algorithm for the vehicle routing problem with time windows, Oper. Res. 40 (2) (1992) 342–354, http://dx.doi.org/10.1287/opre.40.2.342.
[35] D.J. Bertsimas, A vehicle routing problem with stochastic demand, Oper. Res. 40 (3) (1992) 574–585, http://dx.doi.org/10.1287/opre.40.3.574.
[36] C. Archetti, M.G. Speranza, The split delivery vehicle routing problem: A survey, in: The Vehicle Routing Problem: Latest Advances and New Challenges, Springer US, Boston, MA, pp. 103–122, http://dx.doi.org/10.1007/978-0-387-77778-8_5.
[37] M. Gaudioso, G. Paletta, A heuristic for the periodic vehicle routing problem, Transp. Sci. 26 (2) (1992) 86–92, http://dx.doi.org/10.1287/trsc.26.2.86.
[38] R. Assaf, Y. Saleh, Vehicle-routing optimization for municipal solid waste collection using genetic algorithm: The case of Southern Nablus City, Civ. Environ. Eng. Rep. 26 (3) (2017) 43–57, http://dx.doi.org/10.1515/ceer-2017-0034.
[39] M. Akhtar, M.A. Hannan, R.A. Begum, H. Basri, E. Scavino, Backtracking search algorithm in CVRP models for efficient solid waste collection and route optimization, Waste Manage. 61 (2017) 117–128, http://dx.doi.org/10.1016/j.wasman.2017.01.022.
[40] P. van Beek, Backtracking search algorithms, 2006, pp. 85–134, http://dx.doi.org/10.1016/S1574-6526(06)80008-8.
[41] S.M.H. Erfani, S. Danesh, S.M. Karrabi, R. Shad, A novel approach to find and optimize bin locations and collection routes using a geographic information system, Waste Manage. Res. 35 (7) (2017) 776–785, http://dx.doi.org/10.1177/0734242X17706753.
[42] R.J. Kuo, F.E. Zulvia, Hybrid genetic ant colony optimization algorithm for capacitated vehicle routing problem with fuzzy demand — A case study on garbage collection system, in: 2017 4th International Conference on Industrial Engineering and Applications (ICIEA), 2017, pp. 244–248, http://dx.doi.org/10.1109/IEA.2017.7939215.
[43] M. Dorigo, G. di Caro, Ant colony optimization: a new meta-heuristic, in: Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), 1999.
[44] R.C. Prim, Shortest connection networks and some generalizations, Bell Syst. Tech. J. 36 (6) (1957) 1389–1401, http://dx.doi.org/10.1002/j.1538-7305.1957.tb01515.x.
[45] G.A. Croes, A method for solving traveling-salesman problems, Oper. Res. 6 (6) (1958) 791–812, http://dx.doi.org/10.1287/opre.6.6.791.
[46] M.A. Hannan, M. Akhtar, R.A. Begum, H. Basri, A. Hussain, E. Scavino, Capacitated vehicle-routing problem model for scheduled solid waste collection and route optimization using PSO algorithm, Waste Manage. 71 (2018) 31–41, http://dx.doi.org/10.1016/j.wasman.2017.10.019.
[47] H. Asefi, S. Lim, M. Maghrebi, S. Shahparvari, Mathematical modelling and heuristic approaches to the location-routing problem of a cost-effective integrated solid waste management, Ann. Oper. Res. 273 (1–2) (2019) 75–110, http://dx.doi.org/10.1007/s10479-018-2912-1.
[48] H.L. Vu, K.T.W. Ng, B. Fallah, A. Richter, G. Kabir, Interactions of residential waste composition and collection truck compartment design on GIS route optimization, Waste Manage. 102 (2020) 613–623, http://dx.doi.org/10.1016/j.wasman.2019.11.028.
[49] V.K. Sandhya, Issues in solving vehicle routing problem with time window and its variants using meta heuristics - a survey, Int. J. Eng. Technol. 3 (6) (2013) 2013.
[50] E. Düzakın, M. Demircioğlu, Araç Rotalama Problemleri ve Çözüm Yöntemleri, Çukurova üniversitesi iktisadi Ve idari Bilimler Fakültesi Dergisi 13 (1) (2009) 68–87.
[51] Ş Ebesek, H. Yaman, Kaynak kısıtlı projeçizelgeleme probleminde tekrarsız kromozom destekli paralel genetik algoritma uygulaması, J. Int. Soc. Res. 12 (62) (2019) 519–529.
[52] B.E. Gillet, L.E. Miller, J.G. Johnson, Vehicle dispatching — Sweep algorithm and extensions, in: Disaggregation, Springer Netherlands, Dordrecht, 1979, pp. 471–483, http://dx.doi.org/10.1007/978-94-015-7636-9_30.
[53] N. Christofides, Worst-case analysis of a new heuristic for the travelling salesman problem, 1976.
[54] J. Renaud, F.F. Boctor, G. Laporte, An improved petal heuristic for the vehicle routeing problem, J. Oper. Res. Soc. 47 (2) (1996) 329–336, http://dx.doi.org/10.1057/jors.1996.29.
[55] C. Blum, A. Roli, Metaheuristics in combinatorial optimization, ACM Comput. Surv. 35 (3) (2003) 268–308, http://dx.doi.org/10.1145/937503.937505.
[56] A.K. Jain, Jianchang Mao, K.M. Mohiuddin, Artificial neural networks: a tutorial, Computer (Long Beach Calif) 29 (3) (1996) 31–44, http://dx.doi.org/10.1109/2.485891.
[57] D. Karaboga, B. Basturk, Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems, in: Foundations of Fuzzy Logic and Soft Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 789–798, http://dx.doi.org/10.1007/978-3-540-72950-1_77.
[58] A.M. Altabeeb, A.M. Mohsen, A. Ghallab, An improved hybrid firefly algorithm for capacitated vehicle routing problem, Appl. Soft Comput. 84 (2019) 105728, http://dx.doi.org/10.1016/j.asoc.2019.105728.
[59] GeoPy, Welcome to GeoPy's documentation!, Geopy 2.2.0 Documentation.
[60] Yandex, Yandex Maps API, 2022, https://yandex.com/dev/maps/. (Accessed 12 june 2022).
[61] Google, OR-Tools | Google Developers, 2021, https://developers.google.com/optimization. (Accessed 21 November 2021).
[62] OpenStreetMap contributors, Map data © OpenStreetMap contributors, 2015, Terms available at, 2015. www.openstreetmap.org/copyright.