Waste Collection Vehicle Routing Problem using filling predictions

Haizea Rodriguez, Jon Díaz, Jenny Fajardo-Calderín, Asier Perallos, and Enrique Onieva

DeustoTech, Faculty of Engineering, University of Deusto, Av. Universidades, 24, 48007 Bilbao, Spain
{haizearodriguez,jondi}@opendeusto.es, {fajardo.jenny, perallos, enrique.onieva}@deusto.es

Abstract. With the increase in the global population and rising demand for food and other materials, there has been a rise in the amount of waste being generated daily by each house and locality. Most of the waste that is generated is thrown into the garbage containers, from where it is collected by the area municipalities. Improper management of waste in cities results in a huge loss to any smart city. However, with the importance that this has for the health of the citizens and the environment, there are few tools available to manage the planning of waste collection routes efficiently. In this paper, we present a solution to the planning of waste collection routes considering the existence of monitoring capabilities of the containers. The routing of the trucks for each of the days will be formulated as a Capacitated Vehicle Routing Problem. This research focuses on the minimization of the routing times, as well as avoiding the overflow probability. The paper presents the results of a case study with real data on three municipalities in the Basque Country, Spain.

Keywords: Capacitated Vehicle Routing Problem · Garbage Collection · Smart City · Internet of Things

1 Introduction

Technology has disrupted many aspects of our world, creating unlimited paths to solve many complex things in our daily life. Sometimes even taking the next step, not confronting the problem, but rather preventing it from happening, the very basis upon which the smart cities are being built [4]. These cities face a wide variety of challenges, one of them being the accumulation of waste and overfilling of bins in their urban areas. Constant residue collecting routes is the most widespread method of approaching this waste management issue, and it has not evolved in the last century [15].

Residual waste generation is something that technology will not mend in the short term, but it can help to handle it better. One of the approaches to mitigate this problem lies in the proper plan of routes for garbage collection, facing the problem as a variant of the well-known Vehicle Routing Problem (VRP). Usually, the problem is represented employing a graph containing a set of locations that have to be visited only once, additionally, restrictions on the capacity of the trucks or specific time windows are considered in the literature [8].

The most classical VRP is the Capacitated VRP (CVRP). In a typical CVRP, a vehicle is allowed to visit and serve each customer on a set of routes exactly once. The vehicle starts and ends its visit at the central depot such that the total travel cost (distance or time) is minimized and the vehicle total capacity is not exceeded [1].

A smart city is an urban area that uses different types of electronic methods and sensors to collect data. Insights gained from that data are used to manage assets, resources and services efficiently; in return, that data is used to improve the operations across the city. The smart city concept integrates information and communication technology (ICT), and various physical devices connected to the IoT (Internet of things) network to optimize the efficiency of city operations and services and connect to citizens. Smart waste management in a smart city is one service that can be provided [3, 5].

Smart waste management is mainly compatible with the concept of smart cities. Having waste containers with the capability of measuring their current fill will allow the estimation of future states of filling, which will allow effective planning of resources for their collection. The efficient management of waste has a significant impact on the quality of life of citizens.

This work focuses on the planning of waste collection routes considering the existence of monitoring capabilities of the containers, as well as estimation of the filling of them from one day to the next one. Given so, each container may share its actual filling rate, as well as an estimation of the increase for the next day. In this sense, here we propose an optimization problem considering the collection of garbage from different containers in a scenario in which not all of them have to be collected on the same day.

To achieve it, we propose a formulation of an optimization problem for making the decision of which containers will be collected in different days of a time horizon, and, given that, the routing of the trucks for each of the days will be formulated as a CVRP. The formulation has into account both the minimization of the routing times, as well as avoiding the overflow probability.

This paper is structured as follows. Section 2 presents the related literature review. Section 3 states the formulation of the problem to be solved by optimization methods. Section 4 introduces the optimization method proposed. After that, Section 5 presents experimental results obtained. Finally, Section 6 states conclusions and future works.

2 Literature Review

Waste management has a great impact on the environment, managing it efficiently and intelligently has a direct impact on the health of society. The waste collection system named Decision Support System, is enhanced with IoT services that enable dynamic scheduling and routing in smart cities. Different works model waste management as a service of an IoT infrastructure.

The waste collection problem has received some attention in recent years. Gutierrez et al. [10] have proposed a waste collection solution based on providing intelligence to trashcans, by using an IoT prototype embedded with sensors, which can read, collect, and transmit trash volume data over the Internet. The system is simulated in a realistic scenario in the city of Copenhagen and using freely available geolocation data of the municipality-owned trashcans as Open Data.

Buhrkal et al [6] present the study the Waste Collection Vehicle Routing Problem with Time Window, and proposed an adaptive large neighbourhood search algorithm for solving the problem. Huang et al. [11] present the Waste collection problem as a set covering and vehicle routing problem (VRP) involving inter-arrival time constraints, bi-level optimization formula to model the split delivery VRP with several trips to decide the shortest path. Developed an ACO algorithm for the solution obtained.

Reddy et al. [3] proposed work includes a smart dustbin that separates dry and wet waste so that the waste recycling can be done more efficiently. To find the efficient way to find garbage collecting routes, we use shortest path algorithms like Dijkstra's Algorithm.

The idea proposed in [7] considered two garbage bins for waste segregation, and sensors are attached to bins for garbage data collection to avoid overfilling. Overfilling of the bins is prevented using sensors, but no mechanism for waste collection is proposed. Then, Shreyash et al. [12] provides a comparative analysis on different algorithms used for Garbage collection Systems such as Genetic Algorithm Ant Algorithm, Integrated Nearest Neighbor Algorithm and Genetic Algorithm.

Wongsinlatam et al. [16] have proposed the solution to find the minimal total cost for finding the shortest route of waste collection for garbage truck capacity by using metaheuristic techniques. Moreover, Stanković et al. [14] proposed a solution for the collection of waste in a municipality, and in the solution of the model they use the distance-constrained capacitated vehicle routing problem. To solve the problem they used four metaheuristic algorithms, such as: Genetic algorithm, Simulated annealing, Particle swarm optimization, and Ant colony optimization. They develop a case study on the collection and transport of municipal waste in an urban environment in the city of Niš in Serbia.

Wu et al. [17] developed a model for the collection and transportation of urban waste in China, the solution is specific because it is based on the design of algorithms to solve a priority vehicle routing problem. The problem is based on the possibility that waste collection services are immediate for high priority waste containers, for example those containing hospital or medical waste, considering that harmful waste must be collected immediately. They consider the deployment of sensors in the waste containers to make dynamic routes instead of fixed routes. In the solution they used a hybrid search algorithm, which builds the initial solution with the Particle Swarm Optimization, and then a local search is performed on the initial solution, using a Simulated Annealing.

Nurprihatin et al. [13] have proposed a waste collection model using the vehicle routes problem, and add characteristics and constraint, such as: time windows, multiple trips, fractional delivery and heterogeneous fleet. To solve the problem, they use the Nearest Neighbor algorithm, and use data from Cakung District, located in Indonesia, as a case study. Also, Tariq et al. [2] proposed a model IoT-based garbage collection solutions to solid waste management. The proposed system is capable of the effective collection of waste, detection of fire in waste material and forecasting of the future waste generation. The IoT based device performs the controlling and monitoring of the electric bins. Based on the previously collected data from the selected bins, the predictive analytic algorithm namely decision tree and neural network are applied to predict the future waste collection.

After reviewing the literature of various authors, we can see the importance of the Waste Collection Problem for the development of smart cities. We found an increasing need to develop systems that take into account multiple factors, such as capacity of trucks and bins, bin litter level prediction and generates smart routes for the garbage trucks; to reduce the consumption of resources and improve the quality of life of the citizens.

3 Formulation of the Problem

This work faces the problem of the planning of waste collection routes in a dynamic time horizon as a combinatorial optimization problem, where the day in the plan in which a particular container is collected is optimized for solving, and, after that, the particular capacitated VRP for each one of the days is applied. The problem assumes containers having a filling status, as well as a daily increment of it, so the algorithm must ensure that none of the containers overflows. The following subsections explain details of the problem.

3.1 The concept of the system

To find a better way of handling waste management, the proposed system consists of two steps. The first one will generate a plan for a given amount of days, while the second step will be in charge of optimizing said plan.

A plan is defined as an allocation of the day in which a particular container will be collected, within a given time horizon for generating the plan. Selected containers for collection on a particular day of the plan will be visited by a truck of the fleet, considering each particular day as an instance of a CVRP.

It must be noted that, given this definition, a particular plan may not be suitable, because the amount of garbage to be collected exceeds the capacity of the trucks. On the other hand, given the increasing filling of the containers, a plan can derive to a situation in which one (or more) containers overflow. Both situations are considered through the evaluation function, formally presented in Section 3.2, by penalizing those scenarios.

Initially, a feasible plan is created considering the capacity of the trucks, as well as the quantity of garbage to be collected by a greedy procedure. In subsequent iterations of the optimization methods, such an initial plan is slightly modified by a neighbour operator, creating different alternative solutions. The best option among all the possible alternatives is selected for replacing the original one until no more movements can be done, making it so a Hill Climbing Optimization algorithm [9].

Within each day of the plan, an optimization method is in charge of assigning the trucks to locations to visit, as well as establishing the order of the visits, to cover the minimum distance possible. For that procedure, the ORTools[1] framework is used.

The final solution consists of all the information needed to effectively clean all residual waste located in the selected town, which is expressed by a list of routes, a list of arrival times and the load of the truck at each node.

Given this, the plan is done based upon the conditions that every route conforming to a solution must comply with, which are as follows:

1. Both, the starting and ending points are the same for all the vehicles.
2. The total capacity of the trucks may not be exceeded at any point.
3. Each container can only be visited once across the whole plan. After the visit, the filling of the container becomes 0.
4. As every day passed, the load of all the containers is increased by a defined amount.

Finally, in the effort of facilitating the understanding of the outcomes, the routes of each day will be presented in a Folium[2]-generated map, along with a report, addressing specific information, such as the time of the routes and the amount of weight they will manoeuvre.

3.2 Mathematical Model

To proceed with the formulation of the problem, the following nomenclature will be used. A total fleet of N trucks is considered, as well as K containers for collection and D, which will be used to denote the number of days to plan the garbage collection. Each of the trucks has a capacity of N_i $(i = 1 \dots N)$, while each of the containers has an initial filling of F_j, a daily increment in the filling of Δ_j and a maximum capacity of M_j $(j = 1 \dots N)$.

Each day of the plan, the non-collected containers increase their filling according to Equation 1. The overflow of a container is considered when $F_j > M_j$ is fulfilled for any $j \in \{1 \dots N\}$.

$$F_j = F_{(j-1)} + \Delta_j, \forall (j = 1 \dots N) \tag{1}$$

[1] https://developers.google.com/optimization
[2] https://python-visualization.github.io/folium/

Given so, a *plan* to be optimized will be defined and presented as $plan_i = j$, where $i \in \{1 \dots K\}$ and $j \in \{1 \dots D\}$. Therefore, the method will optimize the routes among all the containers that have been selected for collection each day.

For the distribution of collection days to each of the K containers (*plan*), the evaluation (or cost function) defined is presented in Algorithm 1. The rationale of the method is as follows for each day in the planning horizon (D):

1. Initially, the cost is initialized to 10.000, which, in this case is used as a very high starting value denoting a extremely bad solution (equivalent to ∞), for each one of the days on the horizon (line 1).
2. If any of the container's overflow (line 5) or the VRP optimizer is not able to find a solution due to capacity constraints (line 12), the cost of this day is calculated as a very high penalty number plus the quantity of garbage to be collected in the day. This means that, for any two plans that make containers overflow, the one with a minor quantity to collect is less penalized. After that, the loop is stopped, and the following days remain with a high cost (the value of 10.000 is used in this case).
3. In the case of the VRP optimizer being able to find a solution for collecting the containers assigned in the day (line 10), the cost for the day is -10.000 plus the distance travelled by the solution.

As a result of the aforementioned rationale, the algorithm will prioritize short routes that do not make containers overflow during the time horizon. Additionally, if overflow occurs, the optimization function will evaluate better solutions with later overflow.

4 Proposal

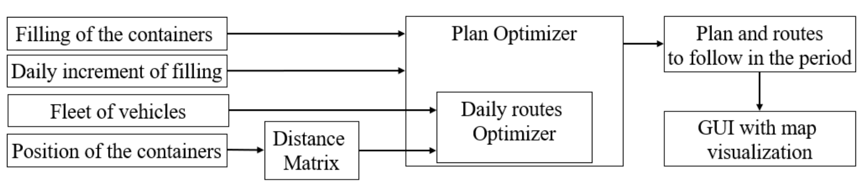

Fig. 1. Overall schema of the solution

To provide an end to end solution, the schema presented in Figure 1 is followed, where input data and several components interact among themselves:

– The position of the containers is extracted from the Open Data portal of the county of Biscay[3] (Spain).

[3] https://opendata.euskadi.eus/catalogo/-/contenedores/

Algorithm 1: Calculation of the cost of a solution (for minimization).

$cost_d = +10.000\ \forall d \in \{1 \ldots D\}$;
for $d \in \{1 \ldots D\}$ **do**
    $collect = \{i \mid plan_i == D_c\}$;
    $noncollect = \{i \mid plan_i! = D_c\}$;
    if $any(F_{collect} > M_{collect})$ **then**
        $cost_d = +10.000 + \sum(F_{collect})$;
        $break$;
    else
        $solution = VRP\_optimizer(collect, F_{collect})$;
        if *solution found* **then**
            $cost_d = -10.000 + solution\_distance$;
        else
            $cost_d = +10.000 + \sum(F_{collect})$;
            $break$;
        end
        $F_{non\_collect} = F_{non\_collect} + \Delta_{non\_collect}$;
        $F_{collect} = 0$
    end
end
$return(\sum cost)$

- Both, the filling and daily increment of the containers are simulated in this study. In Section 5, the concrete values used in the experimentation will be specified.
- Distance matrix among the containers is calculated by using Open Source Routing Machine[4].
- The plan optimizer is in charge of assigning the collection of different containers to one of the days on the horizon.
- For this, it uses the daily routes optimizer component, which is in charge of searching for the optimal solution to the derived CVRP, given a set of containers and a fleet of vehicles. In this proposal, ORTools[5] is used.
- Finally, the system returns a set of routes for each of the days on the horizon, which are presented to the user in a GUI, presenting visualization of the routes in a map powered by Folium[6].

In this work, a Hill Climber optimization algorithm is used within the Plan Optimizer module, whose working operation procedure is presented in Algorithm 2. The algorithm starts building a plan for the collection of the containers by assigning the collection of each container to a random day on the horizon and evaluating it according to the procedure presented in Algorithm 1. After the initial plan is generated, the algorithm, in an iterative way, creates N_s alternative

[4] http://project-osrm.org
[5] https://developers.google.com/optimization
[6] https://python-visualization.github.io/folium/

plans derived from variations of the initial plan, and the cost of each one is subsequently calculated and stored. Once all the alternatives have been evaluated, the plan with the lowest cost replaces the initial one, but only if an improvement is made.

Algorithm 2: Hill Climber optimization procedure.

```
1  plan = randomassignation(K, D);
2  cost = cost(plan);
3  for i ∈ {1 ... iterations} do
4      bestCost = ∞
5      for j ∈ {1 ... N_s} do
6          newPlan = modification(plan);
7          newCost = cost(newPlan);
8          if newCost < bestCost then
9              bestCost = newCost;
10             besPlan = newPlan;
11         end
12     end
13     if bestCost < cost then
14         cost = bestCost;
15         plan = bestPlan;
16     end
17 end
18 return(plan);
```

The $modification$ operator (line 6 in Algorithm 2) randomly increases or decreases by 1 the assigned day of collection of a randomly selected container, always respecting the day of the collection cannot be lower than one or higher than D.

5 Experimentation

5.1 Use Cases

The system includes data from all the municipalities in the Basque Country, Spain, collected from the Open Data portal. For experimentation purposes, a selection of three locations has been done. The selected municipalities ensure a diverse distribution of container, as well as varying size for the problem. The three selected municipalities are:

1. Abadiño: a small municipality with a total of $K = 69$ containers.
2. Durango: a medium-size municipality with a total of $K = 149$ containers.
3. Leioa: a big municipality with a total of $K = 192$ containers.

When generating a solution for any of the locations, parameters such as the number of trucks (N) and the number of days to plan (D) are slightly modified, always taking into consideration that the amount of containers in one location is directly related to the number of trucks needed to collect them. This relationship is of prime importance when defining the values of the parameters because if the value is low, it can lead to failed plans due to the overload of the containers.

The capacity of the containers and trucks are held constant through the entire experimentation. The amount of load a container is capable of bearing is set to $M_j = 400$ kg while trucks can hold up to $N_i = 2000$ kg in each route.

Each container has an initial load and a daily increase. The initial load of the containers (F_j) range from 40 and 200 kg (half of the container capacity), while the daily increase (Δ_j) ranges between 40 and 120 kg. Both of these values have been kept constant across all the experimentation process. Since there are no sensors from which this information can be retrieved, these numbers have been simulated.

5.2 Results and analysis

Table 1 explores different situations of the chosen municipalities, with slight modifications of the parameters D and N. In all configurations, the number of iterations is kept at a constant of 100. Finally, in each execution, the number of alternative plans generated to explore alternative solutions in the Algorithm 2 is established to $N_s = K/3$. Columns in the table are:

- Load: the sum of the load from all the collected containers across the plan, in tons.
- Time: the total amount of time the trucks have been travelling from one container to another, in hours.
- Distance: the total amount of km the trucks have travelled.
- Overflow: whether the plan has been completed without any bin overflowing or not.

One of the glaring observations we can make is that no 5-day configuration has been able to be resolved without the overflow of one of the containers, even when that same configuration has worked for a lower number of days. This phenomenon occurs because containers can only be assigned one day for collection, which makes the bins collected on the first day susceptible to overfill in later days, meaning that the proposed algorithm is not suitable for long-term or periodic planning operations. The adaptation of the coding of the solution to face this issue will be considered in future work.

We can also see that adding more days to the planning may lead to a reduction in the number of trucks (N) needed to collect the garbage. Taking a closer look at the municipality of Leioa (case #13), with the planning of 3 days, at least 12 trucks are needed to find a solution, but if the planning is meant for 4 days, 10 trucks are enough (case #15). Cases #1 and #3 also reflect this fact, which occurs because the containers can be distributed across more days, leading to

#	Location	D	N	Load (tons)	Time (h)	Distance (km)	Overflow
1	Abadiño	3	7	16.95	14.05	118.25	
2	Abadiño	3	10	17.88	15.4	118.25	
3	Abadiño	4	4	17.42	14.18	119.13	
4	Abadiño	4	7	18.73	13.82	80.74	
5	Abadiño	4	10	20.6	16.55	117.18	
6	Abadiño	5	4	9.38	9.73	71.85	x
7	Abadiño	5	7	9.97	10.15	68.51	x
8	Abadiño	5	10	4.52	10.82	67.36	x
9	Durango	3	10	32.92	22.73	88.85	
10	Durango	4	10	37.62	24.31	93.69	
11	Durango	5	10	29.81	20.02	76.94	x
12	Durango	5	15	27.97	17.67	72.10	x
13	Leioa	3	12	42.83	30.33	93.01	
14	Leioa	3	14	43.8	30.57	140.39	
15	Leioa	4	10	47.37	31.13	167.72	
16	Leioa	5	10	38.32	24.55	95.11	x
17	Leioa	5	20	43.48	27.18	123.17	x

Table 1. Experimental results obtained.

a lower number of bins to be collected per day and consequently being able to find a solution with a lower number of trucks.

With regards to the total load collected, in those cases the plan does not end in overflow, we can see minor differences within the same municipality, but higher values in those plans with more days. This is caused to the increment in the load of the containers, which makes the more days a container is not collected, the higher amount of waste to collect.

An example of a possible solution for the configuration of the case #10 can be found in Figure 2.

6 Conclusions and Future Works

This work has presented a system for the planning of waste collection routes for a fleet of trucks for a specified time horizon, in days. The system is built upon the assumption the containers can be monitored, and an estimation of the increase of the filling each day can be provided. Given that, the system results suitable for Smart Cities scenarios.

For the optimization of the planning, it is proposed a formulation of an optimization problem, where the day in which a particular container is collected is encoded. After that, for each particular day, the specific sequence of containers followed by each one of the trucks in the fleet is subsequently optimized using a CVRP optimizer. The fitness function used in the optimization engine considers the minimization of the distance travelled by the trucks, as well as the avoidance of the overflow of the containers within the time horizon.

Experimental results over three municipalities of the Basque Country, in Spain, show the suitability of the approach for the successful planning of the fleet.

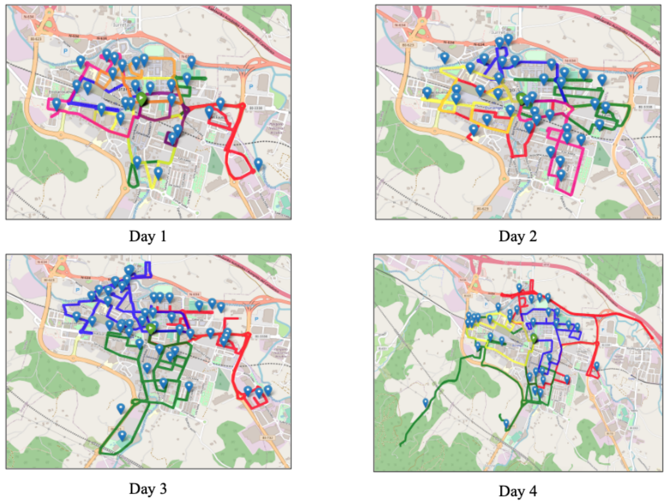

Fig. 2. Result of the optimization for the municipality of Durango

Future works come through the inclusion of the capability of optimizing the collection of the containers in more than a day, to make it suitable for long-term planning of the operations. On the other hand, the improvement of the implemented search heuristic, by using more advanced optimization meta-heuristics are expected to be used in the next stages of the developments. In this regard, the implementation of meta-heuristics able to make use of multiple combination of modification operators, as well as their application and comparative analysis within more use cases will be considered.

Acknowledgements

This work has been supported by The LOGISTAR project, which has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 769142. Also, from the Spanish Government under the project IoTrain (RTI2018-095499-B-C33).

References

1. Adewumi, A.O., Adeleke, O.J.: A survey of recent advances in vehicle routing problems. International Journal of Systems Assurance Engineering and Management **9**(1), 155–172 (feb 2018). https://doi.org/10.1007/s13198-016-0493-4
2. Ali, T., Irfan, M., Alwadie, A.S., Glowacz, A.: IoT-Based Smart Waste Bin Monitoring and Municipal Solid Waste Management System for Smart Cities. Arabian Journal for Science and Engineering **45**(12), 10185–10198 (dec 2020). https://doi.org/10.1007/s13369-020-04637-w

3. Amarnadha Reddy, A., Gangadhar, B., Muthukumar, B., Albert Mayan, J.: Advanced Garbage Collection in Smart Cities Using IoT. In: IOP Conference Series: Materials Science and Engineering. vol. 590. Institute of Physics Publishing (oct 2019). https://doi.org/10.1088/1757-899X/590/1/012020
4. Batty, M.: Smart cities, big data (2012)
5. Bhandari, R., Nidhi, S., Swapnil, R., Dhruvi, D., Harsh, K.: Survey on IOT based Smart City Bin. International Journal of Computer Applications **177**(11), 29–33 (oct 2019). https://doi.org/10.5120/ijca2019919528
6. Buhrkal, K., Larsen, A., Ropke, S.: The Waste Collection Vehicle Routing Problem with Time Windows in a City Logistics Context. Procedia - Social and Behavioral Sciences **39**, 241–254 (jan 2012). https://doi.org/10.1016/j.sbspro.2012.03.105
7. Chowdhury, P., Sen, R., Ray, D., Roy, P., Sarkar, S.: Garbage Monitoring and Disposal System for Smart City Using Iot. In: Proceedings of the 2nd International Conference on Green Computing and Internet of Things, ICGCIoT 2018. pp. 455–460. Institute of Electrical and Electronics Engineers Inc. (aug 2018). https://doi.org/10.1109/ICGCIoT.2018.8753060
8. Defalque, C.M., da Silva, A.F., Marins, F.A.S.: Goal programming model applied to waste paper logistics processes. Applied Mathematical Modelling **98**, 185–206 (2021)
9. Derigs, U., Kaiser, R.: Applying the attribute based hill climber heuristic to the vehicle routing problem. European Journal of Operational Research **177**(2), 719–732 (2007)
10. Gutierrez, J.M., Jensen, M., Henius, M., Riaz, T.: Smart Waste Collection System Based on Location Intelligence. In: Procedia Computer Science. vol. 61, pp. 120–127. Elsevier (jan 2015). https://doi.org/10.1016/j.procs.2015.09.170
11. Huang, S.H., Lin, P.C.: Vehicle routing-scheduling for municipal waste collection system under the "Keep Trash off the Ground" policy. Omega (United Kingdom) **55**, 24–37 (sep 2015). https://doi.org/10.1016/j.omega.2015.02.004
12. Khandelwal, S., Yadav, R., Singh, S.: IoT based Smart Garbage Management-Optimal Route Search. International Research Journal of Engineering and Technology **6** (2019)
13. Nurprihatin, F., Lestari, A.: Waste Collection Vehicle Routing Problem Model with Multiple Trips, Time Windows, Split Delivery, Heterogeneous Fleet and Intermediate Facility. Engineering Journal **24**(5), 55–64 (sep 2020). https://doi.org/10.4186/ej.2020.24.5.55
14. Stanković, A., Marković, D., Petrović, G., Čojbašić, Ž.: Metaheuristics for the waste collection vehicle routing problem in urban areas. Facta Universitatis, Series: Working and Living Environmental Protection **0**(0), 001–016 (may 2020). https://doi.org/10.22190/FUWLEP2001001S
15. Statheropoulos, M., Agapiou, A., Pallis, G.: A study of volatile organic compounds evolved in urban waste disposal bins. Atmospheric Environment **39**(26), 4639–4645 (2005)
16. Wongsinlatam, W., Thanasate-Angkool, A.: Finding the Shortest Route of Waste Collection by Metaheuristic Algorithms using SCILAB. In: Proceedings - 2nd International Conference on Mathematics and Computers in Science and Engineering, MACISE 2020. pp. 20–23. Institute of Electrical and Electronics Engineers Inc. (jan 2020). https://doi.org/10.1109/MACISE49704.2020.00011
17. Wu, H., Tao, F., Yang, B.: Optimization of Vehicle Routing for Waste Collection and Transportation. International Journal of Environmental Research and Public Health 2020, Vol. 17, Page 4963 **17**(14), 4963 (jul 2020). https://doi.org/10.3390/IJERPH17144963