Solving the Vehicle Routing Problem In the Open-Source Software ‘ODL Studio’

Stančić, Nikola; Kovačević, Jovan; Cvijetinović, Željko;
Brodić, Nenad; and Mihajlović, Dragan

Abstract: ***Transport management is one of the most important logistical tasks, and designing the optimal transport routes plays a major role in that process. Vehicle routing is primarily of interest to companies that organize the transportation of goods to end users, intending to minimize the total cost of the routes. Being a well-known NP-hard problem, the theoretical background of the vehicle routing problem has been researched for many years and it has been the basis for the development of a large number of commercial software. However, affordable software solutions are lacking. Difficult access to high-quality and reliable spatial data related to the road network is also a limitation. This paper discusses the possibilities of applying open-source software for solving the VRP. Open spatial data from a collaborative project of the virtual community was used as the basis for modeling the road network graph. The experiment was carried out on data on vehicles and customers of a certain retail chain and refers to the territory of Serbia.***

Index Terms: ***GIS, location-based services, open data, optimal routes, VRP***

1. *Introduction*

THE Vehicle Routing Problem (VRP) is the process of determining the set of optimal routes in the road network along which the vehicles should move when serving a set of customers. It represents a generalization of the travelling salesman problem (TSP). Vehicle routing is present in various fields, most notably in the optimization of the transport of people or, more often, goods. This process is most often done by companies with the goal to visit specified locations where some activity needs to be carried out (e.g., delivery of goods). Routing and scheduling of vehicles have a significant share in the total operating costs of distribution and transportation systems. To reduce such costs, operational research techniques are successfully used in many situations. A route includes the complete path traveled by a vehicle starting from the central depot, visiting the designated locations and finally returning to the central depot. The task should be completed within a specified time and using the existing road network. At the same time, all user requests must be met, and all imposed restrictions must be respected.

The methods for solving transport problems are based on some of the typical spatial analyses and vehicle routing is primarily a spatial problem. The importance of the Geographic Information System (GIS) as a sophisticated technology is reflected in the collection, storage, management, analysis and presentation of spatial data. Location-based computing (LBC) also plays an important role in the optimization of delivery vehicle routes. Its infrastructure consists of a distributed mobile computing environment in which each mobile device is aware of its location and wirelessly connected, which enables Location-Based Services (LBS) [8]. LBS are made up of a wide range of services that combine the device's current location (usually obtained from global navigation satellite systems) with context data (e.g., information about vehicle behavior and driving parameters). The combination of GIS and LBS technologies in solving the VRP is an established part of the procedure in which these two technologies are used for data collection and as decision-making tools [14].

Theoretical frameworks related to various optimization methods for solving logistics problems have been thoroughly researched and well-known for many years [1–2, 11, 13]. However, despite this, in practice, many

Manuscript received September 26, 2022.

N. Stančić is at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade, Serbia (e-mail: nstancic@grf.bg.ac.rs) – contact person.
J. Kovačević is at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade, Serbia (e-mail: jkovacevic@grf.bg.ac.rs).
Ž. Cvijetinović is at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade, Serbia (e-mail: zeljkoc@grf.bg.ac.rs).
N. Brodić is at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade, Serbia (e-mail: nbrodic@grf.bg.ac.rs).
D. Mihajlović is at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade, Serbia (e-mail: draganm@grf.bg.ac.rs).

companies still apply experiential and manual methods at the operational level of optimizing the transport of goods. There are several reasons for this situation, primarily the lack of affordable software solutions. Therefore, the main idea of this paper is to explore the possibility of creating optimal transport routes using open-source tools and open data, available to everyone.

The rest of the paper is organized as follows. In Section 2, the formulation of the Vehicle Routing Problem is given. Section 3 lists established approaches for solving VRP. Section 4 describes the proposed open-source and open-data approach. Finally, Section 5 summarizes the results and suggests future research directions.

2. *Problem Definition*

The vehicle routing problem was first presented by George Dantzig and John Ramser in 1959 [6], calling it the truck dispatching problem. Current VRP models, however, are immensely different from the original one, as they increasingly aim to incorporate real-life complexities, such as time-dependent travel times (reflecting traffic congestion), time windows for pickup and delivery, and input information (e.g., demand information) that changes dynamically over time [3].

Solving the VRP is equal to the minimization of total costs, that is, the goal is to find an optimal set of feasible routes that require minimum costs in terms of mileage, fuel consumption, required time, number of used vehicles or some other criteria. Every vehicle (i.e., driver) is assigned a route and a schedule. In addition to achieving cost minimization, all constraints and requirements have to be satisfied: working hours, the number of mandatory stops, maximum travel time, special user requirements, etc.

Input data required for solving the VRP includes:

- the starting and ending point(s) representing the depot(s);
- available fleet and vehicles characteristics;
- a set of nodes with known locations that represent the customers to be visited;
- existing road network;
- quantity of goods (there may be several types of goods that need to be collected or delivered);
- the time window for each customer in which it is necessary to serve him;
- time required for unloading or loading of goods;
- additional constraints.

In order to determine the total costs of transportation along a certain route, it is necessary to know the costs of travel between individual customers, as well as between customers and depots. The existing road network is represented by a complete graph, where arcs represent roads, and nodes represent intersections, depots or customers. For each pair of points i and j on the complete graph, an arc (i, j) is associated with a weight c_{ij} that is directly related to the distance and/or travel time of the vehicle. All weights are stored within the weight matrix that can be either symmetric or asymmetric. The weight of the entire route is then calculated as the sum of the weights of all segments that belong to the route.

The most common model of the VRP is characterized by capacity limitations and is called the Capacitated Vehicle Routing Problem (CVRP). In it, each vehicle has a defined maximum load value (its carrying capacity), and for each customer, information is kept on the number of goods to be delivered or collected from him. The number of goods to be delivered to customers in one route must not exceed the maximum carrying capacity of the vehicle.

2.1. Variants of the VRP with capacity constraints

Depending on the additional restrictions that are introduced, there is a wide range of CVRP variants. The basic CVRP adopts several assumptions, such as the use of a homogeneous vehicle fleet, single depot, one route per vehicle rule, etc. These limitations can be removed by introducing additional restrictions.

A simpler case is one in which there are no a priori set limits for the delivery time (such as the situation with, e.g., newspaper delivery). If, however, the time of the customer visit is of primary importance and explicitly defined (as, for example, in the case that customers accept deliveries only within a certain time), then time constraints can no longer be ignored and need to be taken into account when routing and creating time schedules. Also, a priority restriction can be introduced that favors individual customers when planning the order of serving customers. One type of priority restriction is the requirement that a particular customer is served on a route that includes a particular subset of other customers. Another possibility is to require that a specific customer is served before (or after) a subset of customers. A limitation of this type represents the problem of pickup and delivery, where goods collected from one customer must be delivered to another customer by the same vehicle. It is a common request to serve several groups of customers in one route, knowing which group of customers should be served first. If both pickup and delivery are carried out on the route, and the restrictions are related to the process of loading and unloading of goods and where it is undesirable to reorganize the load on the route, all deliveries must be made before pickups. [4]

Four characteristic variants of the vehicle routing problem with capacity constraints, classified depending on the additional constraint that is introduced, are:

- Distance-constrained Vehicle Routing Problem – DVRP;
- Vehicle Routing Problem with Time Windows – VRPTW;
- Vehicle Routing Problem with Backhauls – VRPB;
- Vehicle Routing Problem with Pickup and Delivery – VRPPD.

Additionally, the vehicle routing problem can be observed from two perspectives, as a static or dynamic problem. In static models of the vehicle routing problem, it is assumed that all parameters describing the problem are known in advance and decisions are made only once, at the beginning of the planning process. In contrast to static, dynamic models of the vehicle routing problem, the input information is only partially known at the time of vehicle route planning. New information may arrive during the planning process or during its execution, and also the original information may be modified. Because of this, it is not possible to get a definitive route in one attempt [5], making this type of VRP significantly more difficult to solve.

3. *Existing Solutions*

Solving the VRP is considered to be one of the most challenging combinatorial optimization tasks. The problem is classified as an NP-hard (nondeterministic polynomial time) problem [10], which means that the computational effort and solving time required to solve this problem grow exponentially or even factorially with the size of the problem. Exact algorithms are only efficient in cases where the number of customers is relatively small. According to data in [9], the limit is up to 50 to 100 orders, depending on the applied variant and the required response time. Real problems that a company may have usually have significantly more customers, so they cannot be solved in an acceptable time using an exact approach. The calculation time with this approach significantly exceeds the time in which it is necessary to decide on a new route plan.

Instead, for such problems, it is desirable to obtain an approximate solution that can be found quickly enough and is sufficiently accurate. Due to all of the above, a heuristic approach is much more often used for practical solutions, which provides an approximately optimal solution, but in real-time.

3.1. Approaches to solving the VRP

The approaches to solving the vehicle routing problem and the historical development of the methods used can be traced through four decades of the twentieth century. In the early 1960s, only small problems involving 10 to 20 customers could be solved. In the late 1960s, a breakthrough occurred with the introduction of 3-opt and 2-opt route enhancement mechanisms. Algorithms developed until the beginning of the seventies could solve only some of the problems that contained from 30 to 100 customers. In the 1970s, the concept of two-pass algorithms was introduced, and the speed of algorithm execution itself became more significant. The size of the problems that are solved increased significantly, and in that decade, from 1970 to 1980, some of the problems with 25 to 30 customers are solved optimally. In the 1980s, the rapid development of exact problem-solving based on linear programming began. Apart from the exact one, other approaches to solving the vehicle routing problem based on experience are proposed and developed. Some of the problems with 50 customers are optimally solved. A new concept of metaheuristics is being formed at the general level of solving optimization-combinatorial problems based on many methods newly introduced during the 1990s, which are significantly different from each other. Vehicle routing problems that were optimally solved in the nineties reached up to 100 customers.

Therefore, contemporary approaches to solving the vehicle routing problem can be divided into three main groups:

- exact approach;
- heuristic approach;
- metaheuristic approach.

Since the number of possible routes for the general case of vehicle routing grows very quickly, the practical application of the exact approach is very limited. The algorithms belonging to this group are the Column Generation method, the "Branch and Cut" algorithm and Lagrange relaxation. They are primarily used in hybrid algorithms or special cases when the set of possible solutions is significantly narrowed.

The heuristic approach is based on the use of experience, user intuition and own judgment.

Unlike exact methods, heuristic methods try to reduce the number of possible routes in the problem-solving process. Heuristic algorithms are often based on heuristic route construction, where constructing and improving routes is done iteratively. During this process, the customer is required to actively engage, guide and correct the algorithm based on their knowledge, thereby improving the final output of the algorithm. Some of the representatives are the Clark-Wright Algorithm, Sweep Algorithm, Nearest Neighbor Heuristic (NNH), Farthest Addition Heuristic (FAH) and the Christofides-Mingozzi-Toth Algorithm.

The third and most modern of the three approaches to solving NP-hard problems of combinatorial optimization is a metaheuristic, which, as a term in optimization, represents a set of concepts that define heuristic methods used in solving a large number of different problems. Metaheuristics in practice is a set of algorithms that are used to solve many different optimization problems where the algorithm itself changes very little depending on the problem being solved. A metaheuristic approach to solving VRP is often based on local search guided by processes taken from nature [4]. Typical algorithms that have this approach are the method of Simulated Annealing (SA), Genetic Algorithms (GA), Tabu Search (TS) and Ant Colony Optimization (ACO), while the methods that apply Neural Networks (NN) are increasingly being developed.

In addition to the three basic groups of algorithms mentioned above, so-called hybrid methods have also been developed. They use a combination of at least two methods from the group of exact, heuristic and metaheuristic procedures to solve vehicle routing problems. The hybrid approach is still relatively young and in its infancy and, as stated in the paper [9], based on the available literature little research has been carried out using them.

4. PROPOSED OPEN-SOURCE SOLUTION

Open Door Logistics Studio or just ODL Studio [12] is a free and open-source desktop software designed for finding optimal solutions for problems in the field of logistics. It was developed by the British company Open Door Logistics in 2014 and is available under the GNU Lesser General Public License. Tools within it are developed for analysis and manipulation of data on user locations, shaping and mapping of the territory where deliveries are carried out and – the key point for this paper – optimization of vehicle routing and creation of the order for visiting locations. ODL Studio is organized by scripts that represent separate modules, each of which implements one specific functionality. It is possible to run pre-made scripts or create new ones using intuitive wizards. What makes ODL Studio distinctive is that, unlike classic GIS software and similar software, the data used by this application is in XLS format, i.e., in a simple Microsoft Excel spreadsheet. This enables the data to be edited both in ODL Studio and in MS Excel without any problems. Due to the simplicity of this format, interoperability with other systems is also ensured.

For vehicle routing purposes, ODL Studio uses the free GraphHopper routing library written in Java and the open-source jsprit toolkit, which is specialized for solving traveling salesman and vehicle routing problems. A detailed description of the features, content, and API of the Java GraphHopper routing library is available at [7].

4.1. Experiment

The territory for which the optimization tasks were solved is the territory of Serbia. The road network of Serbia is taken from OpenStreetMap data that is compatible with the GraphHopper library and the ODL Studio software. Data regarding available vehicles and locations to be visited are taken from the fleet management system database of the retail chain. The constraints that were taken into account when creating the optimal solution are expressed through the corresponding columns in the input tables. There are limited payloads of delivery vehicles, in addition to time windows that are defined for both vehicles and customer locations. Apart from that, there are two types of customers: one part of customers requests delivery of goods to their locations, while the other part of customers requests that a certain amount of goods be picked up from their locations. Therefore, the solving of the capacitated vehicle routing problem with time windows and simultaneous pickup and delivery (CVRPTWSPD) is illustrated.

GraphHopper models the road network of the given territory in the form of a graph whose arcs represent traffic roads, and nodes represent intersections, depots or customers. The graph obtained in this way is saved in several different files and then used to calculate distances and travel times between nodes. The basis of the graph of the road network is represented by the most up-to-date original OSM data, which were downloaded from the web server of the German company Geofabrik in PBF format.

ODL Studio requires the creation of three tables: *Stops*, *VehicleTypes* and *Stop-order*, where the first two tables contain the input data. These tables should be inside a single XLS file, so that each one is a separate spreadsheet in it, identically named.

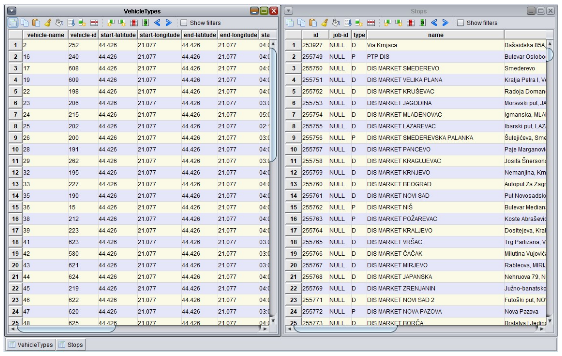


Figure 1: Importing the data on the fleet and the customers

The *Stops* worksheet contains a list of user locations to visit. In addition to basic customer data (primarily delivery addresses and location coordinates), this worksheet also includes time limits set by the customer, the duration of the act of handing over the goods, and the number of goods to be delivered. The purpose of the second worksheet, *VehicleTypes*, is to store general information about the fleet and its vehicles. The coordinates of the initial and final depot, the working hours of the delivery person, the carrying capacity of the vehicle and various parameters used to calculate the driving costs are stored (fixed costs, driving costs per kilometer traveled and per elapsed hour, and waiting costs per elapsed hour). The number of vehicles available in a specific warehouse is also known. The third *Stop-order* depot has the role of a connection table, i.e., each of its rows will carry information about which customer is assigned to which vehicle. This table is initially empty and its filling is done after successfully finding a set of optimal routes.

To populate the *Stops* and *VehicleTypes* tables with real data, SQL scripts on the Microsoft SQL Server database of the trade chain were written. For some columns of the first table, equivalent information was found in the database: location ID and name, as well as longitude and latitude in the WGS84 system. The addresses were obtained by the geocoding process based on the above-mentioned coordinates and the available address system. Filling the other columns with meaningful values had to be simulated in another way. Since the `job-id` and `required-skills` columns do not need to have values, they are set to `NULL`. The value of the `type` column can be `'D'` (for delivery) or `'P'` (for pickup). Each row of the table should receive one of these two values, depending on whether goods should be handed over to a customer or goods should be taken from him (e.g., return of goods). It was assumed that approximately 10% of customers will request a certain amount of goods to be picked up from their locations, while the other 90% will request delivery of goods to their locations. To simulate this real scenario, the basic functions of MS Excel were used.

The limits of the time window, `start-time` and `end-time`, are set to identical values for all users. By default, the user cannot receive the goods before 5:30 a.m. (value `'05:30:00'`), as well as that the delivery of the goods must be done by 8:30 p.m. (value `'20:30:00'`). The duration of the handover is simulated similarly to the `type` column. The assumption is that vehicles will not stay at all locations for the same amount of time. Delivering goods to a small shop will usually involve unloading only a few boxes, which will take significantly less time compared to delivering a larger quantity of goods to a supermarket. In the simulation, it is predicted that the value of the `service-duration` attribute for approximately 60% of locations will be 10 minutes, for about 20% of locations 20 minutes, and for the remaining 20% of locations 40 minutes. The quantities of goods requested by customers (column `quantity`) are assigned through a simulation at the SQL level.

A similar approach applies to the columns of the vehicle fleet characteristics table. Data on vehicle identifiers and names, storage locations, as well as time windows for vehicles are obtained from the corresponding columns. The amount of goods for delivery that can be placed in each vehicle (column `capacity`) depends on the type of vehicle, that is, whether it is a van, a small truck or a tow truck. Available vehicles are classified by capacity based on their garage number.

Additional abilities of drivers or vehicles are not considered, so the value of the `skills` column is equal to `NULL`. Based on the demo input data that can be created within ODL Studio for several countries, the values of the parameters used to calculate the route costs have been entered. Thus, it was assumed that the amount of driving costs per kilometer traveled is `'0.001'`, and per elapsed hour is `'1'`. If the driver is forced to spend a certain amount of time waiting due to time windows, the waiting costs will amount to `'0.5'` per past hour. The part of the costs that are always present and defined by the `fixed-cost` parameter is `100`.

The ODL Studio software module intended for solving vehicle routing problems is called *Vehicle* *routing & scheduling*. The process of finding the optimal solution consists of the following activities:

- import of prepared input data;
- initialization of the tool for optimally solving the vehicle routing problem;
- analysis of resulting routes;
- analysis of Gantt charts;
- (optional) manual changing of routes;
- reports creation;
- final analysis and export.

The created graph of the road network of Serbia and the XLS file with data on available vehicles and existing orders for goods should be loaded into the software. The next step is creating a new script for the *Vehicle routing & scheduling* module with 1 quantity type. This configuration was chosen because the input data refers to one-dimensional quantities of goods. The creation of optimal routes is triggered by the *Optimise* option. The optimization process can take from a few seconds to a few minutes, mostly depending on the amount of data, i.e., the number of locations that vehicles need to visit. In this particular case, the time for processing the input data was just over one minute. An overview of the designed routes is generated by the *View routes in the map* command from the *View solution* menu. It can be seen that OpenStreetMap is used as a background, and also that the route of each vehicle is colored in a different color.

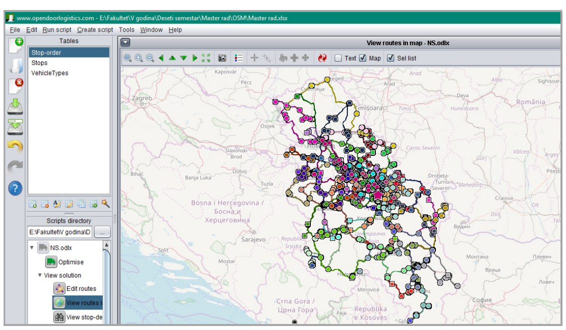


Figure 2: Display of all routes drawn over OpenStreetMap

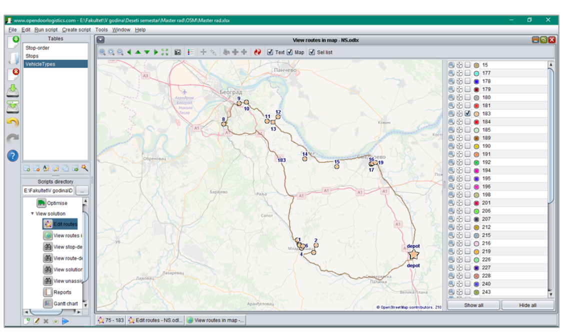


Figure 3: Display of a single route drawn over OpenStreetMap

The map is interactive, meaning that it is possible to pan and zoom the view. By positioning the mouse cursor on an arbitrary location of the customer, insight into the address and time when that location will be visited is obtained. By enlarging the view to the level where the roads can be seen, it is noticeable that the projected routes faithfully follow the road network.

The dynamic plan for the realization of these routes can be viewed using a Gantt chart. The vehicle IDs are listed on the vertical axis, and the time is shown on the horizontal axis. By positioning the mouse cursor on an arbitrary segment of the Gantt chart, information is obtained about the state of the specific vehicle at a specific moment (whether it is serving the customer, standing and waiting, or traveling). Vehicle waits are the consequences of defined time windows.

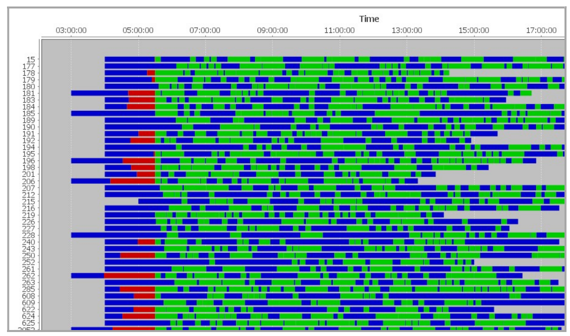


Figure 4: Gantt chart of the obtained solution

A good solution has a small number of inactive periods. The efficiency of the solution is reflected in the degree of time utilization, i.e., the fastest performance of the task (shown in green in the Gantt chart), with the minimization of travel time, which directly increases costs (blue segments in the Gantt chart).

Although the software certainly provided the optimal solution for the given input data, it must be emphasized that this solution is optimal only with respect to the defined constraints. In practice, things need not be so strict. Sometimes it is possible to save resources in total if a slight break of the rules is allowed (a slight exceeding of the defined capacity or a small deviation from the given time window). It is up to the operator to decide whether this is acceptable to him, taking into account other factors and his intuition. Due to the above reasons, it is possible to manually change routes: by excluding a particular location from the solution by moving it to the category of non-aligned locations or by transferring it to another or changing the order of visiting locations within one route. For example, if the company's business strategy allows, it may be decided that a better option is to postpone the delivery of goods to that location to another day. Also, the manual modification could provide a more efficient solution for routes that overlap.

The *View loads* command is very useful. It creates a graph that shows the change in the number of goods in each vehicle over time. Each route is colored in a different color, and there is an option to turn routes on and off by checking the appropriate boxes. It shows how the amount of cargo in vehicles fluctuates over time. This is a consequence of different types of services – in some locations the goods are unloaded from the vehicle, while in some locations it is necessary to pick them up from the customer.

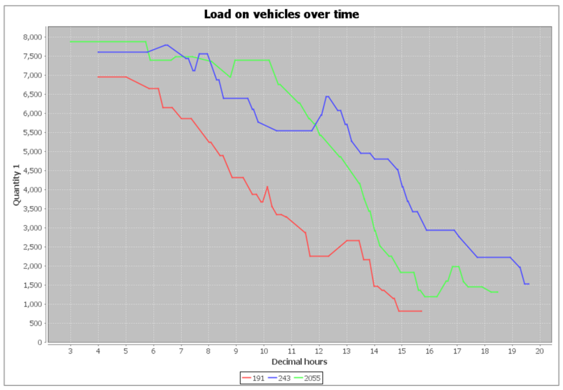
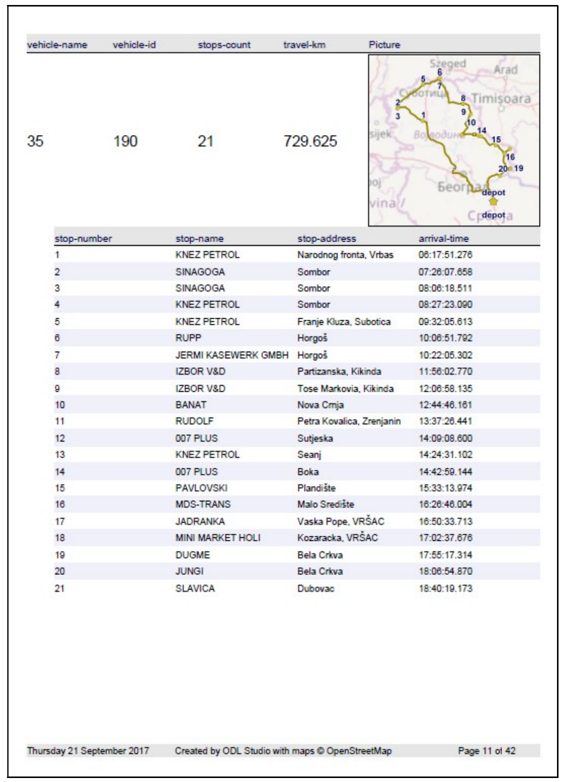


Figure 5: Graph of changes in the number of goods in vehicles over time for three arbitrarily selected routes

In addition to the visual overview of the routes, ODL Studio also offers a high-quality insight into the statistical parameters that describe the obtained solution in detail. All details about planned deliveries, routes and the complete timetable are available within commands under the *View solution* tab. *The View solution-details* command gives the table in which various data about the entire solution are summarized. The *View route-details* option provides a table with similar columns, with the difference that the statistical parameters are listed for each route separately. If there are irregularities in a route in the form of exceeding the limit, that route will be colored red. The *View stop-details* command opens a table with statistics related to the planned locations. This table is the most detailed. *View unassigned stops* is used to list those locations that are excluded from the delivery plan for various reasons. One possibility is that the software itself skipped a certain location because it is in an area for which there is no pre-prepared road network graph. It should be checked that there is no error in the coordinates of the location, which can lead to a situation where a location is mistakenly located in the wrong country and is too far from other locations. It is also possible that the location was deliberately skipped by the decision of the company's management when manually changing the routes.

Reports in PDF format are a convenient form of a visual and statistical overview of the optimized goods delivery plan. The report contains the name and identifier of the vehicle, the number of locations within the route, the length of the trip in kilometers, a graphic representation of the route on the OSM map and a list of locations with data on the name and address of the location and the time when the location will be visited.



vehicle-name	vehicle-id	stops-count	travel-km	Picture
35	190	21	729.625	

stop-number	stop-name	stop-address	arrival-time
1	KNEZ PETROL	Narodnog fronta, Vrbas	06:17:51.276
2	SINAGOGA	Sombor	07:26:07.658
3	SINAGOGA	Sombor	08:06:18.511
4	KNEZ PETROL	Sombor	08:27:23.090
5	KNEZ PETROL	Franje Kluza, Subotica	09:32:05.613
6	RUPP	Horgoš	10:06:51.792
7	JERMI KASEWERK GMBH	Horgoš	10:22:05.302
8	IZBOR V&D	Partizanska, Kikinda	11:56:02.770
9	IZBOR V&D	Tose Markovia, Kikinda	12:06:58.135
10	BANAT	Nova Crnja	12:44:46.161
11	RUDOLF	Petra Kovalica, Zrenjanin	13:37:26.441
12	007 PLUS	Sutjeska	14:09:08.600
13	KNEZ PETROL	Seanj	14:24:31.102
14	007 PLUS	Boka	14:42:59.144
15	PAVLOVSKI	Plandište	15:33:13.974
16	MDS-TRANS	Malo Središte	16:26:46.004
17	JADRANKA	Vaska Pope, VRŠAC	16:50:33.713
18	MINI MARKET HOLI	Kozaracka, VRŠAC	17:02:37.676
19	DUGME	Bela Crkva	17:55:17.314
20	JUNGI	Bela Crkva	18:06:54.870
21	SLAVICA	Dubovac	18:40:19.173

Thursday 21 September 2017 Created by ODL Studio with maps © OpenStreetMap Page 11 of 42

Figure 6: Example layout of one page of a route report

Statistical tables can be viewed in ODL Studio, but they will not be placed in MS Excel tables until they are exported. The commands intended for export are located under the *Export solution tables* tab. After tables are saved, they can be opened and viewed independently of ODL Studio. The exported data is in XLS format, so these tables can be passed to some other system that knows how to read them.

5. Conclusion

This paper considers the possibility of using open-source software on open data to solve the vehicle routing problem. In general, by allowing the computer to create routes for the delivery of goods to user locations, multiple benefits are achieved. Therefore, there is a real interest in companies to implement routing principles in their systems. The performed experiment showed that ODL Studio with the incorporated GraphHopper library is a fairly simple and powerful tool for efficiently solving typical logistics tasks. Moreover, it can be said that the more demanding part of the task is to prepare all the input data appropriately, that is, to satisfy the required form. The steps of processing and finding solutions are simple and logically follow each other. The software is easy to use, which can be attributed to the clarity of the interface, but also to the interactive tutorials from the official website that gradually and intuitively guide the user through the entire procedure. There are different types of analysis of the obtained solution, and the processing speed is quite satisfactory.

The combination of ODL Studio with OpenStreetMap data provides a free, efficient, robust and affordable alternative to commercial software solutions for the same purpose. This confirmed the initial hypothesis that open-source software and free spatial data on the road network are a solid basis for a system intended to solve the problems of organizing the transportation of goods to end users.

The quality of OSM data can compete with the quality of data offered by numerous commercial services and web maps. Their great advantage, which is also used by ODL Studio, is that they are completely open, and available to everyone and anyone can edit and use them for their own needs. As the level of detail, completeness and up-to-dateness of OSM data can vary depending on the location, it is of great importance to improve the quality of OpenStreetMap data and, in general, to further improve and promote this project whose goal is a costless map of the whole world.

Acknowledgement

This research was partially supported by the Serbian Ministry of Education, Science and Technological Development under grant no. 200092.

References

[1] Baker, BM., Ayechew, MA., “A genetic algorithm for the vehicle routing problem”, *Computers & Operations Research*, Vol. 30, No. 5, pp. 787-800, 2003.
[2] Bin, Y., Zhong-Zhen, Y., Baozhen, Y., “An improved ant colony optimization for vehicle routing problem”, *European Journal of Operational Research*, Vol. 196, No. 1, pp. 171-176, 2009.
[3] Braekers, K., Ramaekers, K., Van Nieuwenhuyse, I., “The vehicle routing problem: State of the art classification and review”, *Computers & Industrial Engineering*, Vol. 99, pp. 300-313, 2016.
[4] Carić, T., “*Unapređenje organizacije transporta primjenom heurističkih metoda*”, PhD dissertation, Faculty of Transport and Traffic Sciences, University of Zagreb, 2004.
[5] Chen, WN., “Application of queuing theory to dynamic vehicle routing problem”, *Global Journal of Business Research*, Vol. 3, No. 2, pp. 85-91, 2009.
[6] Dantzig, GB., Ramser, JH., “The Truck Dispatching Problem”, *Management Science*, Vol. 6, No. 1, pp. 80-91, 1959.
[7] *GraphHopper*, Accessed: 19 September 2022, Available: https://www.graphhopper.com
[8] Karimi, HA., Hammad, A., “*Telegeoinformatics: Location-based Computing and Services*”, CRC Press, Boca Raton, Florida, U.S.A., 2004.
[9] Kumar, SN., Panneerselvam, R., “A Survey on the Vehicle Routing Problem and Its Variants”, *Intelligent Information Management*, Vol. 4, No. 3, 2012.
[10] Lenstra, JK., Kan, AR., “Complexity of vehicle routing and scheduling problems”, *Networks*, Vol. 11, No. 2, pp. 221-227, 1981.
[11] Narciso, M., Piera, MA., Guasch, A., “A Methodology for Solving Logistic Optimization Problems through Simulation”, *SIMULATION: Transactions of The Society for Modeling and Simulation International*, Vol. 86, No. 5, pp. 369-389, 2010.
[12] *Open Door Logistics Studio*, Accessed: 19 September 2022, Available: https://www.opendoorlogistics.com
[13] Petrović, G., Madić, M., Antucheviciene, J., “An approach for robust decision-making rule generation: Solving transport and logistics decision-making problems”, *Expert Systems with Applications*, Vol. 106, pp. 263-276, 2018.
[14] Vojinović, M., Cvijetinović, Ž., Đokić, M., Obradović, A., “Primena sistema za nadzor voznog parka u kompanijama koje upravljaju otpadom – primer JKP Mediana Niš”, INFOTECH 2012, Vrnjačka Banja, 29-31 May 2012.

Nikola Stančić is a PhD student and works as a teaching assistant at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade. He graduated as a Bachelor of Geodesy in 2015 and defended his master’s thesis in 2017. His main research areas are GIS, spatial data management, remote sensing and location-based services.

Jovan Kovačević, PhD works as an assistant professor at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade. He graduated as a Bachelor of Geodesy in 2015, defended his master’s thesis in 2017 and received his doctoral degree in 2022. His main research areas are remote sensing, photogrammetry, GIS and machine learning.

Željko Cvijetinović, PhD works as an associate professor at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade. He graduated as a Bachelor of Geodesy in 1989 and received his doctoral degree in 2005. His main research areas are digital terrain modeling, GIS, photogrammetry and LiDAR.

Nenad Brodić is a PhD student at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade. He graduated as a Bachelor of Geodesy in 2010 and defended his master’s thesis in 2012. He works as a teaching assistant at the Faculty of Civil Engineering, University of Belgrade. His main research areas are LiDAR, GIS, remote sensing and spatial data management.

Dragan Mihajlović, PhD works as an associate professor at the Department of Geodesy and Geoinformatics, Faculty of Civil Engineering, University of Belgrade. He graduated as a Bachelor of Geodesy in 1980 and received his doctoral degree in 1991. His main research areas are photogrammetry, remote sensing, GIS and 3D modeling.