Vehicle Routing Problem with Distance Constraints aAnd Clustering Using MATLAB

D. Krstev*, S. Dimitrov*, A. Krstev**

*Faculty of Mechanical Engineering, University of Stip, Republic of N. Macedonia,
**Faculty of Computer Science, University of Stip, Republic of N. Macedonia,
sasko.dimitrov@ugd.edu.mk, dejan.krstev@ugd.edu.mk, aleksandar.krstev@ugd.edu.mk,

Abstract: The problem of designing routes for vehicles that should supply different customers with defined locations and specific demand from a single or various depots is known as the vehicle routing problem. The main objective in this case is minimizing the total cost of delivery or maximizing the profit while taking into consideration some constraints that vary from a case to another. In this paper I am going to define this problem, present a mathematical model to describe it, talk about the existing solutions to solve it, and use different tools to solve a real VRP of a company in tangier.

I. INTRODUCTION

Vehicle Routing Problem (VRP) or a vehicle routing problem is illustrated by the figure below which shows a multi-node problem in which two vehicles have to make cargo deliveries to 11 stations and return to the warehouse or starting node. Cargo needs are in parentheses next to each node, and distances in kilometers are shown on the arcs. Initially, the 12-node is grouped into two groups, one for each vehicle. Nodes 2 to 6 are marked for vehicle 1, while nodes 7 to 12 are marked for vehicle 2. Node 1 is the starting node, warehouse. Practically, clustering takes into account physical barriers and obstacles, such as rivers, mountains, etc., as well as geographical areas such as cities that form a natural group. Capacity constraints and payloads are taken into account when grouping is developed. Thus, the payload of one vehicle is 45 t and of the other vehicle 35 t.

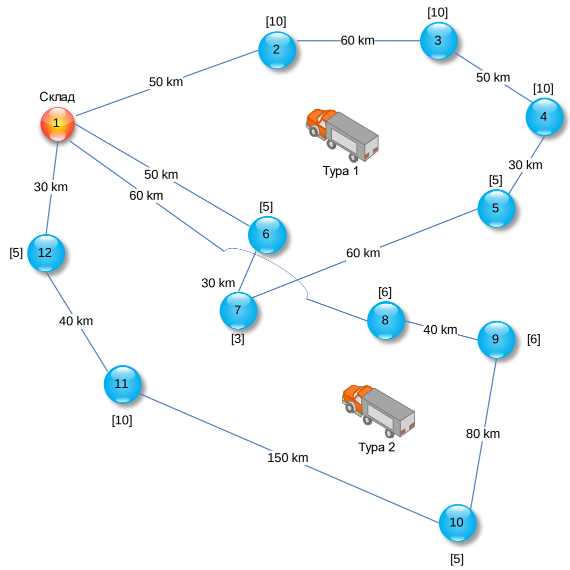

FIGURE 11. VEHICLE ROUTING PROBLEM: REVISED SOLUTION

II. VEHICLE ROUTING PROBLEM (VRP)

Vehicle Routing Problem (VRP) or a vehicle routing problem is identical and extends the MTSP problem to include service requirements for each node at different vehicle capacities. The purpose of these problems is to minimize the total cost or distance across all routes.

TABLE 1. DISTANCE BETWEEN CITIES IN EASTERN MACEDONIA

	1	2	3	4	5	6	7
1		39	99	50	91	168	154
2	39		60	55	66	145	131
3	99	60		116	52	151	156
4	50	55	116		40	109	113
5	91	66	52	40		83	66
6	168	145	151	109	83		85
7	154	131	156	113	66	85	

The quantity of units of a new product L-Carnitine that is transported to individual cities in Eastern Macedonia is shown in Table 2, and the capacity of vehicles (K) is 5000 units.

TABLE 2. COALITIONS OF UNITS REQUIRED

i	(2) Kumanovo	(3) K. Palanka	(4) Veles	(5) Stip	(6) Delcevo	(7) Strumica
di	14 000	900	1 400	2 000	900	1 800

Sij savings are calculated and shown symmetrically with the following values in Table 11.

TABLE 3. THE ESTIMATED SAVINGS SIJ

	2	3	4	5	6	7
2		78	34	64	62	62
3	78		33	138	116	97
4	34	33		101	109	91
5	64	138	101		176	179
6	62	116	109	176		237
7	62	97	91	179	237	

Subordinated savings: [7,6], [7,5], [5,6], [5,3], [6,3], [4,6], [4,5], [7,3], [4,7], [3,2], [5,2], [6,2], [7,2], [4,2], [4,3].

First we consider the case for transport of the product from (7) Strumica to (6) Delchevo. They can be presented on the same route for the need of 2,700 units in a vehicle with a capacity of 5,000 units. A 7 → 6 connection is made, so nodes 7 and 6 will be neighbors of the route in the final solution.

Further we will consider the route from (7) Strumica to (5) Stip. If they are neighbors on the route it would be desirable to connect 6 → 7 → 5 or 5 → 7 → 6. The total number of units which is 4 700 in this route does not exceed the capacity of the vehicle (5000) Since the transported 4 700 units so far reach the capacity of the vehicles, the route of the first vehicle is completed.

We will review the route of the second vehicle in the nodes (3) Kriva Palanka and (2) Kumanovo. They can be presented in the same route because the required units for delivery are 1,400 and 900, ie 2,300 units, which fills the capacity of 5,000 units.

The next route is (4) Veles and (3) Kriva Palanka, which can be connected to the previous route 3 → 2, thus obtaining the desired route 4 → 2 → 3 or 3 → 2 → 4 with 3,700 units.

The delivery of the entire quantity of units (8400) for Eastern Macedonia is performed with two routes and two vehicles, namely 1 → 5 → 7 → 6 → 1 and 1 → 4 → 2 → 3 → 1. The total distance traveled by the first vehicle is 408 km, while for the second vehicle it is 246 km.

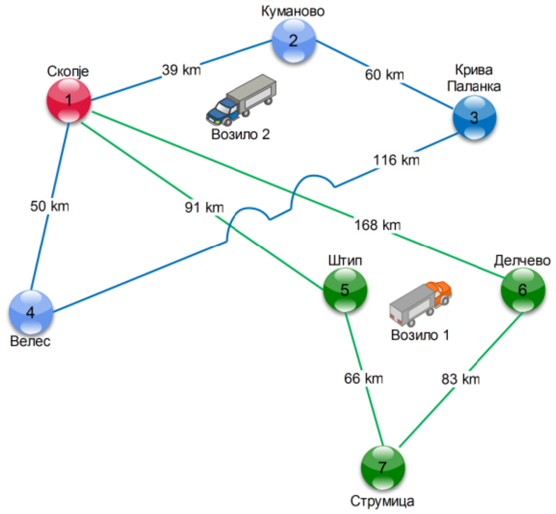

FIGURE 2. VEHICLE ROUTING PROBLEM FOR EASTERN PART OF NORTH MACEDONIA

The savings and quantities of delivered units of a new product with a quantity of 10 800 units in Western Macedonia, the route display and the number of vehicles are given in Tables 1, 2 and 3.

TABLE 4. DISTANCE BETWEEN CITIES IN WESTERN MACEDONIA

	1	8	9	10	11	12	13	14
1		131	176	174	159	67	44	112
8	131		47	106	32	108	132	62
9	176	47		66	52	124	146	78
10	174	106	66		138	107	132	61
11	159	32	52	138		140	164	62
12	67	108	124	107	140		24	46
13	44	132	146	132	164	24		70
14	112	62	78	61	62	46	70	

The quantity of units of a new product that is transported to individual cities in Western Macedonia, while the capacity of one vehicle (K1) is 7000 units and the second (K2) is 4000 units.

TABLE 5. COALITIONS OF UNITS REQUIRED

(8) Prilep	(9) Bitola	(10) Ohrid	(11) Krusevo	(12) Gostivar	(13) Tetovo	(14) Kicevo
1 600	2 000	2 000	1 200	1 200	1 600	1 20

Sij savings are calculated and displayed symmetrically with the following values in the table.

TABLE 61. THE ESTIMATED SAVINGS SIJ

Sij	8	9	10	11	12	13	14
8		260	199	258	90	43	181
9	260		284	283	119	146	210
10	199	284		205	134	86	225
11	258	283	205		86	39	209
12	90	119	134	86		87	133
13	43	146	86	39	87		86
14	181	210	225	209	133	86	

Subordinated savings: [9,10], [9,11], [8,9], [8,11], [10,14], [9,14], [11,14], [10,11], [8,10], [9,13], [10,12], [12,14], [9,12], [8,12], [12,13], [11,12], [13,14], [10,13], [8,13], [11,13].

First we consider the case for transport of the product from (9) Bitola to (10) Ohrid. They can be presented on the same route for the need of 4,000 units in a vehicle with a capacity of 7,000 units. A 9 → 10 connection is made, so nodes 9 and 10 will be neighbors of the route in the final solution.

Further we will consider the route from (9) Bitola to (11) Krushevo. If they are neighbors on the route it would be desirable to connect 10 → 9 → 11 or 11 → 9 → 10. The total number of units which is 5 200 in this route does not exceed the capacity of the vehicle (7000).

The next route with the greatest distance saving is (8) Prilep to (9) Bitola, if they are neighbors in the route it would be desirable to connect 10 → 9 → 8 → 11 or 11 → 8 → 9 → 10. As the 6,800 units transported so far approach the capacity of the first vehicle (7,000 units), the route of the first vehicle is completed.

We will review the route of the second vehicle in the nodes (12) Gostivar and (13) Tetovo. They can be presented in the same route because the required delivery units are 1,200 and 1,600, respectively 2,800 units.

The next route is (12) Gostivar and (14) Kicevo, which can be connected to the previous route 12 → 13, thus obtaining the desired route 13 → 12 → 14 or 14 → 12 → 13 with 4,000 units.

The delivery of the entire quantity of units (10 800) for Western Macedonia is done with two routes and two vehicles, and they are 1 → 10 → 9 → 8 → 11 → 1 or 1 → 11 → 9 → 8 → 10 → 1 and 1 → 14 → 12 → 13 → 1 or 1 → 13 → 12 → 14 → 1. The total distance traveled by the first vehicle is 455 km, while for the second vehicle it is 226 km.

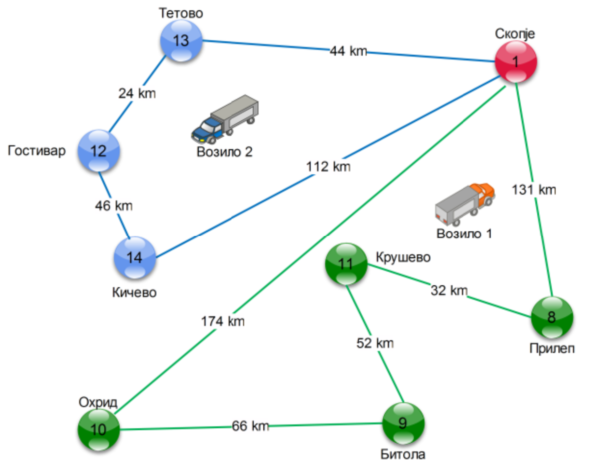

FIGURE 3. VEHICLE ROUTING PROBLEM FOR WESTERN PART OF NORTH MACEDONIA

III. STEPS OF THE IMPLEMENTATION PROCESS

To explain the process, I am going to use the first cluster group to find the optimal route for the first vehicle.

The first thing to do is declaring the number of stations we have. In our first cluster, we have 8 stations (including the depot). Next, it is necessary to get an appropriate matrix; in order to do so, zeros (Number_of_Stations, 1) gives as a 12 by 1 matrix of zeros for both stations' longitudes and latitudes. Then, it is necessary to enter the coordinates of our given stations that will be used to fill our matrices using a for loop. The screenshot below shows the code part of what I just described.

```
Number_of_Stations = 12;  %Declaration of the
number of stations n = 1;

Stations_Longitude = zeros(Number_of_Stations,1);
 %Allocating x-coordinates
Stations_Latitude = Stations_Longitude;
%Allocating y-coordinates

%Entering the coordinates of our given Stations
X = [-5.9206340,-5.839599,-5.824116,-5.809254,-
5.9258994,-5.82757,-5.8627424,-5.8350188 …]
Y=[35.7203120,35.73981,35.754001,35.753916,35.
7359144,35.7500991,35.751059,35.7538851…]

%This loop fills the Matrices with the given
Stations' coordinates  for i=l:Number_of_Stations
Stations_longitude(i)=X(i);
STATIONS^LATITUTDE(i)=Y(i);
end
```

The next step is plotting the stations in their appropriate location using the given coordinates. For accurate visualization, it is necessary to import the real map. In order to do so, I used 'plot_google_map' that uses a code written and saved in the same file as the main code that generates information from google map.

Moving forward, we need to determine the possible paths, what I mean by this is the possible ways of selecting 2 stations from the 12 given stations. Since this is a combination, we have: $C_n^k = \frac{n!}{k!(n-k)!}$. With n=12 stations and k=2, we get 28 possible combinations between 2 stations (1-2, 1-3, 1-4…) Next, the distance between all the possible paths is calculated using the hypot function as follows:

```
Possible_paths=nchoosek(l:Number_of_Stations,2);
%Possible Combinations of 2 selected stations
display (Possible_paths)
%Distances calculation
Euclidean_distance -
hypot(Stations_latitutde(Possible_paths(:,D) -
Stations_latitutde(Possible_paths(:,2)),
Stations_longitude(Possible_paths(:,D) -
Stations_longitude(Possible_paths(:,2)));
len = length(Euclidean_distance);
        display(Euclidean_distance)
```

The code fragment below shows the setting of the constraints and the use of the intlinprog function.

```
Aeq=spones(1:length!idxs));

beq=Number_of_Stations;

Aeq=[Aeq;spaUoc(Number_of_Stations,length)idxs), Number_of_Stations*(Nuniber_of_Station12-1))];

for ii = 1:Number_of_Stations
        whichldxs = (idxs == ii);
        % find the trips that include stop ii
        whichldxs = sparse(sum(whichldxs,2));
        % include trips where ii is at either end
        Aeq(ii+1,:) = whichldxs';
        % makes sure to include the constraint
matrix
End

beq = [beq; 2*ones(Number_of_Stations,l)];
intcon = l:len;
lb - zeros(len,l);
ub = ones(len,l);

Distance_Limit=40;
%This is my inequality constraint
opts = optimoptions('intlinprog'',Display','off');

        [x_tsp,costopt,exitflag,output]
intlinprog(Euclidean_distance,intcon,Euclidean_distance,Distance_Limit,Aeq,beq,lb,ub,opts);
```

Moving forward, there is another very important point to consider while programming which are the subtours eliminations. When I run the program the first time, the result I got was a set of non-linked tours. In order to avoid this, it is necessary to add a part in the code to eliminate the subtours. You can see in the appendix in the code part, the section where the subtours are eliminated.

Since I used longitude and latitude as coordinates. The calculated distance is Euclidean. In order for me to know the distance in Kms to compare it with the optimal tour length generated by the other tools, I simple multiplied the decision variables vector with the distance vector. The fragment code below shows this step (since the distance vector is long it doesn't totally appear):

```
%Calculation of total distance in Kms
'irst_cluster_distances=[9.198,11.25,13.996,8.392,10.506,8.316,10.158,2.921,6.278,14.818,2.177,3.727,2.441.811,15.8..]
Optimal_route=First_cluster_distances*x_tsp;
display(Optimal_route)

display(x_tsp)
:itle('Solution for 8 stops with subtours eliminated');
lold off

lisp(output.absolutegap)
```

IV. RESULTS AND CONCLUSION

The total length of the optimal route generated and with the following route in order:

Looking at the solutions for routing the vehicle in Eastern Macedonia I can create a complete tour 1 → 2 → 4 → 5 → 7 → 6 → 3 → 1 whose length is 535 km. Since this method does not always give the optimal value of the tour, I reviewed the network again with the help of a software tool and found a better tour, such as 1 → 4 → 6 → 7 → 5 → 3 → 2 → 1. The total distance of this tour is 461 km, compared to the previous one of 535 km, or a difference of 48 km.

Looking at the solutions for routing the vehicle in Western Macedonia I can create a complete tour 1 → 13 → 12 → 14 → 10 → 9 → 8 → 11 → 1 whose length is 479 km. Since this method does not always give the optimal tour value, I reviewed the network again and found a better tour, such as 1 → 13 → 12 → 14 → 10 → 9 → 11 → 8 → 1. The total distance of this tour is 456 km, compared to the previous one of 479 km, or a difference of 23 km.

REFERENCES

[1] Daneshzand F, The Vehicle-Routing Problem, Elsevier, 2011,127.
[2] Christofides N, The vehicle routing problem, Numdam, 56.
[3] Sahbi B I, Legras F, Coppin G, Synthèse du problem de routage de véhicule,telecom Bretagne, 7
[4] Mais H R, Wahiba R C, Bloch C, Classification de problemes de tournées de vehicules, 2008.