**A Heuristic for Solving Open Vehicle Routing Problem with Capacity Constraints**

V.Bapi Rajua, Govada Rambabu b, Kandukuri Narayana Raoc

*aDept. of Mechanical Engg,V.R. Siddhartha Engineering College,Vijayawada– 520007, AP,India*

*bDept. of Mechanical Engg,Andhra university,Visakhapatnam –530003, AP,India*

*cDept. of Mechanical Engg,Government Polytechnic College, AP,India*

\*Corresponding author Email: rajuvpublications@gmail.com

**Abstract:** In this paper a new heuristic is developed to solve an open vehicle routing problem (OVRP). This heuristic considers minimizing the number of vehicles as the primary objective and minimizing the total distance travelled as secondary objective. The new heuristic is tested on standard test instances found in literature. The results are then compared with the eleven other methods. This new heuristic is found to be quite effective compared to other methods if both speed of computation and closeness to the best value are considered important. The output from the heuristic can further be improved by meta-heuristics if necessary.

***Keywords*:** OVRP, Minimum no of Vehicles, heuristic

1. **Introduction**

Transportation is one of the important cost in logistics cost. Capacitated vehicle routing problems (CVRP) is one of the interesting optimization problem. Open Vehicle routing problem (OVRP) is a special type of vehicle routing problem in which the vehicle is not required to return the depot. Minimizing number of vehicles and minimizing total distance are two main objectives which are considered by many researchers in OVRP. Many a times it is seen hiring/maintaining a new vehicle is costlier than using existing vehicles for longer distances. Hence most methods use hierarchical objectives, which consider minimizing the number of vehicles used as primary objective and minimizing the total distance as secondary objective. Most of the time these two objectives are conflicting meaning when the number of vehicles is less the distance travelled is more and vice versa. Hence an attempt should be made to first get a solution satisfying the primary objective of minimizing the number of vehicles and later this initial solution can be improvised to reduce the total distance travelled without increasing the number of vehicles.

1. **Literature Survey**

Vehicle Routing Problem (VRP) was first proposed by Dantzig and Ramser in 1959, and has been proved to be an NP-complete problem1. Since OVRP is a special case of VRP where vehicles do not return to depot it is also an NP-hard problem. Exact methods of solving OVRP are not practical as the computational time increases exponentially with larger number of customers. Hence heuristics and metaheuristics are used to get the solution. Sariklis and Powell2 proposed a two-phase heuristic which first assigns customers to clusters and then builds a Hamiltonian path for each cluster, Tarantilis et al.3 described a population-based heuristic, while Tarantilis et al.4,5 presented threshold accepting metaheuristics, Brandão6 Fu et al. 7,8 proposed tabu search heuristics. Pisinger and Ropke9 presented an adaptive large neighbourhood search heuristic which follows a destruct-and-repair paradigm. Li et al.10 described a record-to-record travel heuristic. V.Bapi Raju et al.11 developed a heuristic for VRP problem which minimizes the number of vehicles even when the capacity ratio (total demand/total capacity) is high.

1. **Methodology**

This method is based on two phase method of solving VRP. In the first phase the customers are clustered by modified sweep method and then in the second phase the routes are determined using modified travelling salesman problem (TSP) for each cluster. This method is based on two new propositions. First one is that sweeping should start from a node whose angular distance from next consecutive node, considering depot as origin, is very large (as shown in Fig 1). This would ensure routes are densely packed and would reduce the total distance travelled. Second preposition is that vehicles can be loaded to their capacity if some of the customers can be skipped during sweeping. This would result in less number of vehicles especially when the capacity ratio is close to 1. Based on the above discussion 2 separate algorithms are developed. For a given problem first Algorithm 1 is used to obtain a solution which can handle problems with less capacity ratio. If this does not result in least number of vehicles then algorithm 2 is used which can solve problems with high capacity ratio. A modified travelling salesman problem is used which considers the distance from the last vehicle to the depot to be zero (because this is an OVRP). The flow chart of the method can be seen in Fig 2.

Fig 1.tif

Fig. 1. Angle between two successive customers

No

Yes

Cluster Using Algorithm 2

Calculate Minimum Number of Vehicles Kmin

Number of rotues= Kmin

Cluster Using Algorithm 1

m Clusters

No

Yes

Fianl Solution

C=C+1

C=1

Add Depot to cluster

Using modified TSP solve the problem

C=m

Fig. 2. Flow Chart showing the methodology

**Algorithm 1**

1. Calculate minimum number of vehicles Kmin, using the formula

Kmin = ((Total Demand)/( Capacity of each vehicle)) rounded up to the nearest integer.

1. Locate the depot as the centre. Compute the polar coordinates of each customer with respect to the depot. Sort all customer w.r.t to polar angle. Calculate the angular distances between the successive nodes w.r.t depot as shown in Fig 1. Identify the two successive nodes which form the maximum angular distance. Let these nodes be N1 and N2.
2. Starting from customer N1 sweeping is done by increasing polar angle in the clock wise direction. Clusters are formed using the standard sweep. The formation of route for each cluster is done by using modified TSP and the distance travelled for each route is calculated. The total distance (dist 1) for the solution is obtained by summing up the solutions for each route. Starting from customer N2 sweeping is done by increasing polar angle in the anti clock wise direction. The total distance (dist 2) for the solution is obtained by using the same procedure as above.
3. Best solution from the above two (i.e. solution corresponding to the minimum of dist1 and dist2) is selected as the final solution.

**Algorithm 2**

1. Calculation of Kmin and starting points of sweeping(N1,N2) are determined as explained in Algorithm 1
2. Starting from customer N1 sweeping is done by increasing polar angle in the clock wise direction. Assigning of customers is continued until constraints are violated. If the current vehicle is not having minimum specified percent of capacity (example 95%) swap the last customer who was added to the current cluster with a nearest unrouted customer who would meet the specified capacity level criteria. If no customer is found then the swapping is tried with the last but one customer. This process is repeated until a suitable customer is found who would meet the specified capacity level criteria. The minimum specified percent of capacity is dependent on the capacity ratio. Higher the capacity ratio higher will be the specified percent capacity. The customer who was removed becomes the first customer in the next route. This process would ensure that vehicles are nearly loaded to their capacity resulting in less number of vehicles. This process is repeated until all customers are covered. If this solution does not result in least number of vehicles this solution is ignored. Otherwise the solution for is calculated and dist1 is determined as explained in algorithm1
3. Starting from customer N2 sweeping is done by increasing polar angle in the anti clock wise direction. A second solution is obtained by using a similar step as above to get total distance dist2. If both the solutions result in least number of vehicles then the solution corresponding to the least distance is the best solution. Otherwise the solution with the least number of vehicles is the best solution.
4. **Experimental tests**

Seven test instances from Christofides et.al12 work have been selected. The new heuristic is run on the test instances and the results are tabulated in table 1. The results from other papers mentioned in literature survey are also presented. The new algorithms are implemented on Matlab. The experiments have been done on a PC (Intel Core i3-3470 CPU @ 3.20 GHzCPU, 4GB RAM) with Windows 7 OS.

Table. 1. Comparison of results from various methods

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **1 -CFRS [2]** | | **2- TSAK[5]** | | **3- TSAN[6]** | |
| Problem | n | Kmin | Distance | Time(s) | Distance | Time(s) | Distance | Time(s) |
| C1 | 50 | 5 | 488.204 | 0.22 | 416.1 | 88.8 | 438.2 | 1.7 |
| C2 | 75 | 10 | 795.334 | 0.16 | 574.5 | 167.5 | 584.7 | 4.9 |
| C3 | 100 | 8 | 815.042 | 0.94 | 641.6 | 325.3 | 643.4 | 12.3 |
| C4 | 150 | 12 | 1034.139 | 0.88 | 740.8 | 870.2 | 767.4 | 33.2 |
| C5 | 199 | 16 | 1349.709 | 2.2 | 953.4 | 1415 | 1010.9 | 116.9 |
| C11 | 120 | 7 | 828.254 | 1.54 | 683.4 | 696 | 713.3 | 15.7 |
| C12 | 100 | 10 | 882.265 | 0.76 | 535.1 | 233.6 | 543.2 | 7.8 |
| **Total** | | | **6192.947** | **6.7** | **4544.9** | **3796.4** | **4701.1** | **192.5** |
|  |  |  |  | |  | |  | |
|  |  |  | **4-BR[3]** | | **5-BATA[4]** | | **6-LBTA[5]** | |
| Problem | n | Kmin | Distance | Time(s) | Distance | Time(s) | Distance | Time(s) |
| C1 | 50 | 5 | (6)412.96 | 7.2 | (6)412.96 | 38.62 | (6)412.96 | 28.75 |
| C2 | 75 | 10 | (11)564.06 | 25.8 | (11)564.06 | 68.89 | (11)564.06 | 61.21 |
| C3 | 100 | 8 | (9)641.77 | 28.8 | 642.42 | 56.54 | (9)639.57 | 53.78 |
| C4 | 150 | 12 | 735.47 | 75 | 736.89 | 81.69 | 733.68 | 84.13 |
| C5 | 199 | 16 | (17)877.13 | 226 | 879.37 | 98.13 | (17)870.26 | 96.47 |
| C11 | 120 | 7 | (10)679.38 | 29.4 | (9)679.60 | 37.67 | (10)678.54 | 25.36 |
| C12 | 100 | 10 | 534.24 | 14.4 | 534.24 | 84.54 | 534.24 | 64.59 |
| **Total** | | | **-** | **406.6** | **-** | **466.08** | **-** | **414.29** |
|  |  |  |  |  |  |  |  |  |
|  |  |  | **7-TSF[7]** | | **8-TSR[8]** | | **9-ALNS 25K[9]** | |
| Problem | n | Kmin | Distance | Time(s) | Distance | Time(s) | Distance | Time(s) |
| C1 | 50 | 5 | 408.5 | 170 | 413.3 | 65 | 416.06 | 120 |
| C2 | 75 | 10 | 587.8 | 202 | 570.6 | 197.8 | 567.14 | 360 |
| C3 | 100 | 8 | 644.3 | 720 | 617 | 367.6 | 641.76 | 850 |
| C4 | 150 | 12 | 734.5 | 1610 | 741.1 | 1094 | 733.13 | 1790 |
| C5 | 199 | 16 | (17)878.0 | 2061 | (17)886.6 | 1279.9 | 897.93 | 1240 |
| C11 | 120 | 7 | 753.8 | 736 | 716.5 | 88.9 | 682.12 | 730 |
| C12 | 100 | 10 | 549.9 | 413 | 534.8 | 30.9 | 534.24 | 800 |
| **Total** | | |  | **5912** |  | **3124.1** | **4472.38** | **5890** |
|  |  |  |  |  |  |  |  |  |
|  |  |  | **10- ALNS 50K[9]** | | **11- ORTR[10]** | | **(New method developed by the author)** | |
| Problem | n | Kmin | Distance | Time(s) | Distance | Time(s) | Distance | Time(s) |
| C1 | 50 | 5 | 416.06 | 230 | 416.06 | 6.2 | 436.83032 | 4.221 |
| C2 | 75 | 10 | 567.14 | 530 | 567.14 | 31.3 | 666.96628 | 5.276 |
| C3 | 100 | 8 | 641.76 | 1280 | 639.74 | 39.5 | 674.21606 | 9.521 |
| C4 | 150 | 12 | 733.13 | 2790 | 733.13 | 128.6 | 811.3622 | 14.19 |
| C5 | 199 | 16 | 896.08 | 2370 | 924.96 | 380.6 | 955.37176 | 17.91 |
| C11 | 120 | 7 | 682.12 | 1410 | 682.54 | 121.6 | 854.52124 | 14.28 |
| C12 | 100 | 10 | 534.24 | 1180 | 534.24 | 32.9 | 604.88708 | 8.348 |
| **Total** | | | **4470.53** | **9790** | **4497.81** | **740.7** | **5004.155** | **73.746** |

1. **Results and discussion**

The total distance travelled and the time taken for all the test instances is calculated for each of the methods and is presented in the table 1. The solutions from the methods 4,5,6,7,8 yield higher number of vehicles then minimum required. Hence total distances for the solutions, corresponding to these methods, is not calculated. They are removed from further comparison. Method 1 is fastest taking about 7 seconds but the solution is much inferior. The total distance travelled is approximately 39% more than the best (least distance). Methods 2,9,10,11 provide reasonably good solutions (distance travelled is not more than 2% than the best) but they time taken by these methods is very large (not less than 740 seconds). Method 3 provides solution which is approximately 5% higher than the best but the time taken is 28 times more than best (which is 7 sec by method 1).They newly developed heuristic provides the solution which is only 12% higher distance than the best and time taken is nearly 11% higher than the best. Hence this algorithm can used to get an approximate solution somewhat quickly. Further this solution can be used as a initial solution for other methods for improving the solution.

1. **Conclusion**

A new heuristic for solving OVRP is developed. The main objective of this heuristic is to minimize the number of vehicles. The secondary objective is to minimize the total distance travelled. This heuristic provides a reasonably good solution using a short amount of processing time. The output from this heuristic is compared with eleven other methods from literature. The comparison of the outputs shows that the new heuristic can be quite useful if both parameters, the speed of computation and closeness to the best value, are important. The output from this heuristic can be further improved bymetaheuristics if necessary. This heuristic can be used get a rough idea about the final solution of any OVRP problem within a short interval of time.

**References**

[1] G. B. Dantzig and J. H. Ramser, :The Truck Dispatching Problem, Management Science, Informs, vol. 6, no.l, pp. 80-91, (1959)

[2] Sariklis, D., Powell, S.: A heuristic method for the open vehicle routing problem.Journal of the Operational Research Society 51, 564-573 (2000)

[3] Tarantilis, C.D., Diakoulaki, D., Kiranoudis, C.T.: Combination of geographical information system and efficient routing algorithms for real life distribution operations. European Journal of the Operational Research 152, 437-453 (2004)

[4] Tarantilis, C.D., Ioannou, G., Kiranoudis, C.T., Prastacos, G.P.: A threshold accepting approach to the open vehicle routing problem. RAIRO Operations Research 38, 345-360 (2004)

[5] Tarantilis, C.D., Ioannou, G., Kiranoudis, C.T., Prastacos, G.P.: Solving the open vehicle routing problem via a single parameter metaheuristic algorithm. Journal of the Operational Research Society 56, 588-596 (2005)

[6] Brandão, J.: A tabu search algorithm for the open vehicle routing problem. European Journal of Operational Research 157, 552-564 (2004)

[7] Fu, Z., Eglese, R., Li, L.Y.O.: A new tabu search heuristic for the open vehicle routing problem. Journal of the Operational Research Society 56, 267-274 (2005)

[8] Fu, Z., Eglese, R., Li, L.Y.O.: Corrigendum: A new tabu search heuristic for the open vehicle routing problem. Journal of the Operational Research Society 57, 1018 (2006)

[9] Pisinger, D., Ropke, S.: A general heuristic for vehicle routing problems. Computers & Operations Research 34, 2403{2435 (2007)

[10] Li, F., Golden, B., & Wasil, E., The open vehicle routing problem: Algorithms, large-scale test problems, and computational results. Computers & operations research, 34(10), 2918-2930.(2007)

[11] V.Bapi Raju, Govada Rambabu, Kandukuri Narayana Rao, A Heuristic for Solving CVRP, ICRDME 2019, in press

[12] Christofides, N., Mingozzi, A., Toth, P.: The vehicle routing problem. In: Christofides, N., Mingozzi, A., Toth, P., Sandi, C. (eds.), Combinatorial Optimization, pp.313-38. Wiley, Chichester (1979)