A New Heuristic for Solving Vehicle Routing Problem with Capacity Constraints

Vangipurapu Bapi Raju[1,*], Govada Rambabu[2], Kandukuri Narayana Rao[3]

[1]Department of Mechanical Engineering, V.R. Siddhartha Engineering College, Vijayawada-520007, India
[2]Department of Mechanical Engineering Andhra University, Visakhapatnam-530003, India
[3]Department of Mechanical Engineering, Government Polytechnic College, Vijayawada-520008, India

*Corresponding author email: rajuvpublications@gmail.com, Tel.: +91 9703469728

ABSTRACT

In this paper, a heuristic method for solving capacitated vehicle routing problem (CVRP), is developed. The new method proposed uses a modified sweep algorithm that produces a solution with the least number of vehicles, in a relatively short amount of time. This objective is achieved by loading the vehicles nearly to their full capacity, by skipping some of the customers if necessary. This method is tested on standard Augerat benchmark instances. The results are compared with the results from other similar methods. The results indicate the modified sweep algorithm provides a better solution in terms of the number of vehicles required without much increase in total distance traveled. The complexity of the algorithm is only o(n), and hence the algorithms produces the results in a relatively short time. The output from this method can be further improved by using Meta heuristics like Genetic algorithm.

Keywords: CVRP, heuristic, modified sweep algorithm, skipping customer, minimum vehicles.

1. INTRODUCTION

Transportation is one of the important costs of logistics. Capacitated vehicle routing problems (CVRP) is one of the interesting optimization problems. Minimizing the number of vehicles and minimizing total distance are two main objectives which are considered by many researchers in CVRP. Many times it is seen hiring/maintaining a new vehicle is costlier than using existing vehicles for longer distances. Hence most heuristics use hierarchical objectives, which consider minimizing the number of vehicles used as the primary objective and minimizing the total distance as the secondary objective. That is because a solution with less number of vehicles is treated as a better one than the solution with more number of vehicles even when the total distance traveled in the latter case is less. Most of the time these two objectives are conflicting meaning when the number of vehicles is less the distance traveled 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 traveled without increasing the number of vehicles.

2. LITERATURE REVIEW

Vehicle Routing Problem (VRP) was first proposed by Dantzig and Ramser [1] and has been proved to be an NP-hard problem. Its purpose is to design the least costly routes for a fleet to serve geographically scattered customers. According to different applications and restrictions, many extended VRP types were classified. The Capacitated Vehicle Routing Problem (CVRP) is known as the basic extension, in which the total demand of any vehicle cannot exceed a preset capacity value. Since VRP is a NP-hard problem CVRP is also a NP-hard problem. Since the first VRP was presented, many algorithms have been proposed for solving either the classical VRP or its variants. At present, those algorithms can be divided into three main groups: exact algorithms, heuristics, and metaheuristics. Toth and Vigo [2] have given a detailed description of these methods. An exact algorithm is an algorithm that solves a problem to optimality. This category includes branch and bound approach, cutting planes, network flow, dynamic programming approach and so on. Laporte [3] and Liong et.al [4] gave an overview of these methods. Exact methods are suitable for small instances only as complexity increases rapidly with an increase in the number of customers, making these methods unsuitable in those cases. So, numerous studies have concentrated on developing heuristics to obtain near-optimal solutions. Classical heuristics refer to use the experience based inductive reasoning and the experimental analysis to solve a problem. They include mathematical programming method, improvement or exchanges methods, saving or insertion methods, cluster first route second and route first cluster second and so on. Classical heuristics for CVRP have been surveyed by G. Laporte and Semet [5] and Laporte et al. [6] The classical heuristic approaches can find one feasible solution quickly, but this feasible solution may have a large disparity compared with the best solution. In recent years, based on biology, physics, and artificial intelligence, metaheuristics were developed. Metaheuristics have been applied for various fields due to its efficient optimization performance. In most works of literature printed, most of the CVRP are tackled by metaheuristics, such as Tabu Search, Simulated Annealing (SA), Immune Algorithm, Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization etc. Although metaheuristics are effective and efficient they consume a lot of CPU time to arrive at the solution. For example, a typical genetic algorithm runs for more than 100 iterations to solve even medium sized vehicle routing problem consuming a lot of computational time in the process. Secondly, metaheuristics often suffer from parameter optimization. A thorough knowledge of the problem structure or a lengthy trial-and-error process is needed to select the parameter set carefully. The best parameter set is usually re-determined for each of the problem instances considering the application area, size or input data of the problem. Diaz and Laguna [7] state that during the development of a heuristic of CVRP about 10% of the total time is dedicated to designing and testing of a new heuristic is spent for development, and the remaining 90% is consumed in the tuning of parameters. Lastly, the quality of initial solution can impact the performance of the metaheuristics. These problems can be overcome to a certain extent if we have a good initial solution, reduced objectives, simplified problem. Thus there is a need for a heuristic to get a good initial solution. A good heuristic should be able to provide this initial solution in a relatively short amount of time. Also, output from the heuristic should not deviate much from the best value.

Gillett and Miller [8] developed the sweep algorithm that applies to planar instances of VRP. Feasible routes are created by rotating a ray centered at the depot and gradually including customers in a vehicle route until the capacity or route length constraint is attained. A new route is then initiated and the process is repeated until the entire plane has been swept. In each solution, the routes are optimized for by moving and exchanging the customers between the routes. The authors have shown that the processing time increases quadratically as the average number of locations per route increases keeping the total number of customers constant, but when the total number of customers is increased keeping the average number of customers in each route constant the processing time increases only linearly. This indicates that real processing time is consumed only by the improvement phase and not by the formation of routes. Aziz et al [9] developed a Hybrid Heuristic Algorithm for solving CVRP. His method consists of using the sweep algorithm and the Nearest Neighbor algorithm. Shin and Han[10] developed a Centroid-Based Heuristic Algorithm for the Capacitated Vehicle Routing Problem. However, although all these methods provide a reasonable initial solution they may not yield a solution with least numbers of vehicles especially when the tightness (Total demand of all customers/ total capacity of all vehicles) is high. Faulin and Garca [11] highlighted the need of an algorithm to generate a solution with minimum number of routes and they proposed ALGELECT algorithm for this. However, even in their case, the algorithm does not result in having the least number of vehicles always. Veera Senthil and Jayachitra [12] discussed the concept of skipping a customer during a liner sweep and selecting the next customer in order to increase the capacity of the Vehicle.

In this paper, a new heuristic, based upon sweep algorithm, is developed to solve CVRP problems. The output from this heuristic always has the least number of vehicles even when the tightness (total demand /total capacity) is equal to 1. This is achieved by loading the vehicles nearly to their full capacity whenever the tightness is near 1. Note that this algorithm uses at most 2 x n different ways of grouping the customers where n is the number of customers. Hence the complexity increases only linearly with an increase in the number of customers. The complexity of the algorithm is only O(n). Hence the computational time is only minimal even when the number of customers and number of vehicles are large. The new modified sweep is based on two new propositions. First one is that sweeping should start from a node whose angular distance from next consecutive node w.r.t depot is very large. This is shown in Fig. 1 As can be seen from Fig. 1 it makes sense to start sweeping from customer 1 in the anti-clockwise direction or from customer 12 in the clockwise direction. This prevents customer 1 and 12 being in the same vehicle and subsequently increasing the traveled distance. In other words, this would ensure routes are densely packed with customers and would reduce the total distance traveled.

Second proposition is that vehicles can be loaded nearly to their full capacity if some of the customers can be skipped during sweeping. This would result in less number of vehicles. This is explained using Fig. 2 and Fig. 3 The capacity of each vehicle is assumed to be 100. In Fig. 2 normal sweeping is done from customer 1 in the anticlockwise direction and a new vehicle is formed whenever the total demand exceeds the capacity of the vehicle. This results in total of 4 vehicles. Fig. 3 corresponds to modified sweep. Here after adding customers 5, 6, 7 to vehicle 2 customers 8, 9, 10, 11 are skipped (since they result in violating capacity constraint) and customer 12 is added. This would result in only 3 vehicles. Depending upon the tightness of the vehicles, 2 separate algorithms are developed one using the normal sweep and the other using the modified sweep. For a given problem first algorithm 1 which uses normal sweep is used to obtain a solution. If this does not result in the least number of vehicles then algorithm 2 is used.

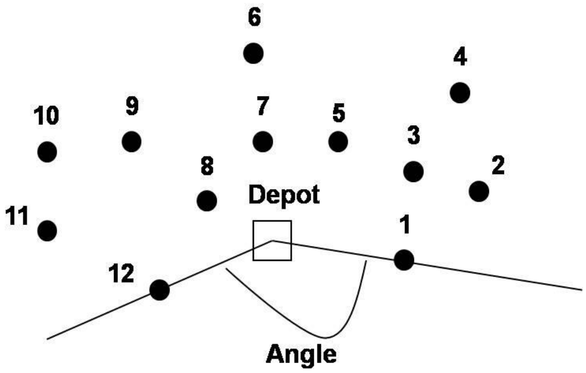

Fig. 1 Angle between successive customers

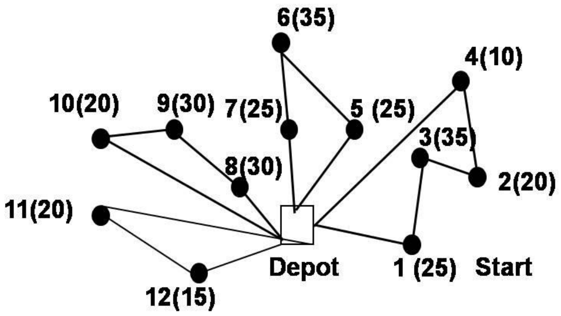

Fig. 2 Solution using a normal sweep (Vehicle Capacity 100).

The model is tested on Augerat benchmark problems [13]. The output is compared with the output of similar methods.

Rest of the document is organized as follows. Modified sweep algorithm is explained in the next section. Results and discussion is the next section. Finally, in the last section, the conclusion of this work is summarized. In the appendix portion solution to some of the problems are given.

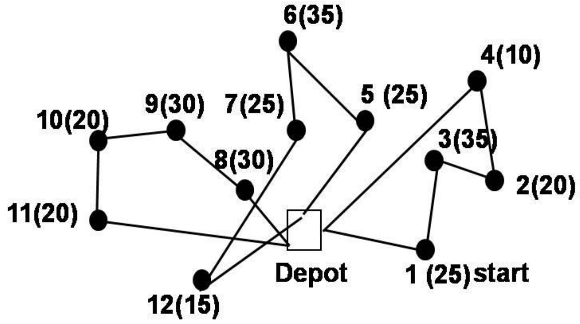

Fig. 3 Solution using a modified sweep (Vehicle Capacity 100)

3. MODIFIED SWEEP ALGORITHM

Two separate algorithms are developed based on the tightness. First algorithm (Algorithm 1) is based on normal sweep algorithm and it does not skip any customers. Second algorithm (Algorithm 2) skips some of the customers so as to load the vehicle to its maximum capacity. This may increase the distance but decreases the number of vehicles required. A solution is tried with algorithm 1 and if this does not result in the least number of vehicles algorithms 2 (which can handle problems with tightness close to 1) is used.

3.1 Algorithm 1

- Calculate the minimum number of vehicles (m) required as follows. K_{min}= ((Total Demand)/(Capacity of each vehicle)) rounded up to the nearest integer.

- Locate the depot as the center. 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 and the depot. Identify the two successive nodes which form the maximum angular distance as shown in Fig 1. Let these nodes be N1 and N2.

- Starting from customer N1 sweeping is done by increasing polar angle in the clockwise direction. As-signing of customers is continued until constraints are violated. Then a new route is formed by resuming the sweep where the last one is left off. This process repeated until all the customers have been assigned to routes. The solution for each route is calculated by using standard travelling salesman problem (TSP) and the distance traveled 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-clockwise direction.. The total distance (dist 2) by using a similar procedure as above.

- Best solution from the above two (i.e. solution corresponding to the minimum of dist1 and dist2) is selected as the final solution.

3.2 Algorithm 2

- Calculation of K_{min} and identification of the two successive nodes between which the angular distance is maximum is done using the method which is explained in algorithm 1. Let these nodes be N1 and N2.

- Starting from customer N1 sweeping is done by increasing polar angle in the clockwise 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 in the current route with a nearest unrouted customer who would meet the specified capacity level. 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. The minimum specified percent of capacity is dependent on the tightness ratio. This would result in the loading of the vehicle to its maximum capacity. A new vehicle is started after this. 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 each route is calculated by using standard travelling salesman problem (TSP) and the distance traveled 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-clockwise direction. A second solution is obtained by using a similar step above to get total distance dist2.

- If both the solutions result in the 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.

Flow chart for the above process is shown in Fig. 4.

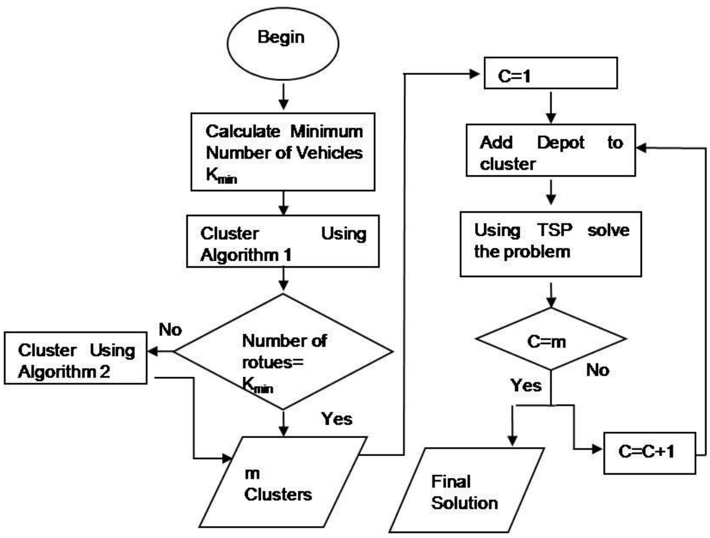

Fig. 4 Flow chart showing solution method

4. EXPERIMENTAL TESTS

Augerat benchmark problems [13] are chosen for testing and the output is compared with the centroid [10], sweep + cluster adjustment [10] and ALGELECT [11] methods. 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.

5. RESULTS AND DISCUSSIONS

The newly developed algorithm is run on the Augerat 'A' series instances. The results are tabulated in Table 1. The outputs from similar other heuristics are also presented in the table.

From the results, it can be seen that the newly developed algorithm always results in the lowest average number of vehicles with only a slight increase in the average distance. Hence this algorithm produces better results than the other methods mentioned. Besides this, the new algorithm produces the results using very little of CPU time thus satisfying the requirements of a good heuristic. Hence the new algorithm can be used to get a reasonably good solution using very little of processing time. The results from some of the outputs is given in the appendix.

Table 1: Comparison of results with other methods from literature

Instance		Best known	Centroid-based 3-phase		Sweep + cluster adjustment		ALGELECT		New algorithm before		
Instance	Tightness (Demand/Capacity)	Distance	Distance	Routes	Distance	Routes	Distance	Routes	Distance	Routes	CPU time
A-n32-k5	0.82	784	881	5	872	5	1032	5	884.4598	5	3.72
A-n33-k5	0.89	661	728	5	788	5	789	5	699.8436	5	2.32
A-n33-k6	0.9	742	770	6	829	7	834	7	751.6514	6	2.07
A-n34-k5	0.92	778	812	5	852	6	835	5	840.5644	5	2.34
A-n36-k5	0.88	799	814	5	884	5	908	5	884.6734	5	2.52
A-n37-k5	0.81	669	756	5	734	5	783	5	757.9387	5	2.8
A-n37-k6	0.95	949	1 027	7	1050	7	1046	6	1231.991	6	2.48
A-n38-k5	0.96	730	819	6	874	6	861	6	910.3944	5	2.73
A-n39-k5	0.95	822	864	5	971	6	990	5	975.605	5	2.86
A-n39-k6	0.88	831	881	6	966	6	861	6	1078.085	6	2.73
A-n44-k6	0.95	937	1 037	7	1092	7	1028	6	1056.838	6	3.04
A-n45-k6	0.99	944	1 040	7	1043	7	1040	7	1141.552	6	3.25
A-n45-k7	0.91	1146	1 288	7	1281	7	1258	7	1303.298	7	3.06
A-n46-k7	0.86	914	992	7	1013	7	1062	7	976.3825	7	3.12
A-n48-k7	0.89	1073	1145	7	1143	7	1162	7	1153.364	7	3.36
A-n53-k7	0.95	1010	1117	8	1116	8	1191	7	1092.612	7	3.87
A-n54-k7	0.96	1167	1209	8	1320	8	1291	7	1319.988	7	4.08
A-n55-k9	0.93	1073	1155	10	1192	9	1191	9	1213.038	9	3.72
A-n60-k9	0.92	1354	1430	9	1 574	10	1503	9	1587.149	9	4.32
A-n61-k9	0.98	1034	1201	11	1184	11	1181	10	1261.932	9	4.49
A-n62-k8	0.92	1288	1470	9	1559	9	1408	8	1409.71	8	4.68
A-n63-k9	0.97	1616	1766	10	1823	10	1745	9	1818.431	9	4.49
A-n63-k10	0.93	1314	1405	11	1523	11	1409	10	1501.159	10	4.37
A-n64-k9	0.94	1401	1587	10	1597	10	1521	9	1639.659	9	4.79
A-n65-k9	0.97	1174	1276	10	1351	10	1357	9	1315.996	9	4.65
A-n69-k9	0.94	1159	1283	10	1254	10	1389	9	1245.483	9	5.09
A-n80-k10	0.94	1763	1883	11	2014	11	1901	10	2068.7	10	6.09
Average		1042	1135	7.67	1181	7.77	1169	7.22	1190	7.07	3.6

6. CONCLUSION

A new heuristic algorithm for the CVRP based on the modified sweep with additional constraints of using the least number of vehicles is developed. This heuristic is used to get a reasonably good solution with least number of vehicles. The output from the heuristic can also be used as an initial solution by meta heuristics like a genetic algorithm to get the best solution. The complexity of the algorithm is only O (n). Therefore this method generates the output in a relatively short time. This is evident from the fact that the average CPU time taken is only 3.6 sec.

APPENDIX

The solution for one of the instance A-n33-K5 is shown in Fig. 5 and the routes are shown in Table 2.

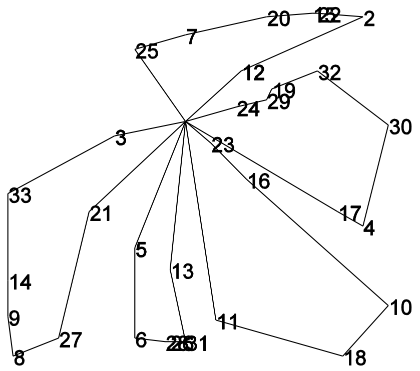

Fig. 5 Solution of the instance A-n33-K5 (Tightness 0.89)

Table 2 Solution of the A-n33-K5 instance

Tightness 0.89	
Vehicle 1	0 3 33 14 9 8 27 21 0
Vehicle 2	0 5 6 28 26 31 13 0
Vehicle 3	0 11 18 10 16 23 0
Vehicle 4	0 17 4 30 32 19 29 24 0
Vehicle 5	0 12 2 22 15 20 7 25 0

The solution for one of the instance A-n45-K6 is shown in Fig. 6 and the routes are shown in Table 3.

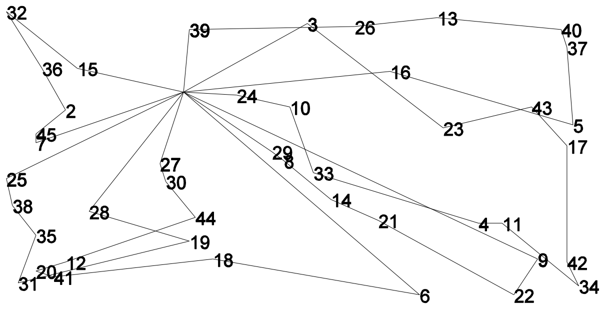

Fig. 6 Solution of the instance A-n45-K6 (Tightness 0.99)

Table 3. The solution of the A-n45-K6 instance

Tightness 0.99	
Vehicle 1	0 7 45 2 36 32 15 0
Vehicle 2	0 25 38 35 31 19 28 0
Vehicle 3	0 6 18 41 20 12 44 30 27 0
Vehicle 4	0 9 22 21 14 8 29 0
Vehicle 5	0 3 23 43 17 42 34 11 4 33 10 24 0
Vehicle 6	0 16 5 37 40 13 26 39 0

REFERENCE

[1] G. B. Dantzig, J. H. Ramser, The truck dispatching problem, *Management Science,* 6 (1), 1959, 80-91.

[2] P. Toth, D. Vigo, *The vehicle routing problem*, SIAM Monograhs on Discrete Mathematics and Applications, 2001.

[3] Laporte, Gilbert, The vehicle routing problem: An overview of exact and approximate algorithms, *European Journal of Operational Research,* 59 (3), 1992, 345-358.

[4] C.Y. Liong, R.I. Wan, O. Khairuddin, Z. Mourad, Vehicle routing problem: models and solutions, *Journal of Quality Measurement and Analysis,* 4 (1), 2008, 205-218.

[5] G. Laporte, F. Semet, *Classical heuristics for the capacitated VRP*, The Vehicle Routing Problem, 2002.

[6] G. Laporte, M. Gendreau, J.Y Potvin, F. Semet, Classical and modern heuristics for the vehicle routing problem, *International Transactions in Operational Research*, 7, 2000, 285-300.

[7] A.D. Belarmino, M. Laguna, Fine-tuning of algorithms using fractional experimental designs and local search, *Operations Research,* 54 (1), 2006, 99-114.

[8] E. Gillet Billy, L.R. Miller, A heuristic algorithm for the vehicle-dispatch problem, *Operations Research*, 22(2), 1974, 340-349.

[9] M.M. AbdElAziz, H.A. El-Ghareeb, M.S.M. Ksasy, Hybrid Heuristic Algorithm for Solving Capacitated Vehicle Routing Problem, *International Journal of Computers & Technology*, 12 (9), 2014, 3844-51

[10] S. Kwangcheol, S. Han, A centroid-based heuristic algorithm for the capacitated vehicle routing problem, *Computing and Informatics,* 30 (4), 2012, 721-732.

[11] J. Faulin, A. García del Valle, Solving the capacitated vehicle routing problem using the ALGELECT electrostatic algorithm, *Journal of the Operational Research Society,* 59 (12), 2008, 1685-1695.

[12] V. Senthil Kumar, R. Jayachitra, Linear Sweep Algorithm for Vehicle Routing Problem with Simultaneous Pickup and Delivery between Two Depots with Several Nodes, *Global Journal of Pure and Applied Mathematics*, 12 (1), 2016, 897-908.

[13] Test data source. \url{http://neo.lcc.uma.es/vrp/vrp-instances/}