



Simulated Annealing Algorithm for Multi Depot Two-Echelon Capacitated Vehicle Routing Problem

Surendra Reddy Kancharla^{1*}, Gitakrishnan Ramadurai²

¹Ph.D. Candidate, Department of Civil Engineering, Indian Institute of Technology Madras, Chennai-600036, India

²Associate Professor, Department of Civil Engineering, Indian Institute of Technology Madras, Chennai-600036, India

Abstract

Although urban freight transport is a significant contributor to the development of a country, it has adverse effects on the environment and the quality of life in urban areas. To reduce these adverse effects, we can deploy sustainable city logistic strategies. Scheduling and routing of vehicles is a crucial decision in city logistic strategies. Hence, in this paper, we solve the Multi-Depot Two Echelon Capacitated Vehicle Routing Problem (MD2E-CVRP), which is a variant of Vehicle Routing Problem (VRP) with heterogeneous fleets at both levels. Since VRP is NP-hard, we have proposed a Simulated Annealing (SA) based heuristic solution algorithm and have tested it on the standard 2E-CVRP and MD2E-CVRP instances. The results obtained from SA have a good solution quality at just one-fiftieth of computational time using CPLEX and was found to be faster than Adaptive Large Neighborhood Search (ALNS) with only a marginal drop in solution quality.

Keywords: Two-Echelon Capacitated Vehicle Routing Problem, Simulated Annealing, Mixed-Integer Linear Programming

1. Introduction

Urban freight transport is a major contributor to the development of a country. However, it is also a significant contributor to environmental pollution (Kijewska, Konicki and Iwan, 2016) that is severely affecting the quality of life in urban areas. Due to the rapid degradation of air quality in several megacities of the world, the local governments are bringing in restrictions on the movement of heavy-duty freight vehicles in these cities. These restrictions lead to a steep increase in the use of light-duty freight vehicles, especially in the cities of developing countries. In recent years, several studies (Crainic, Ricciardi and Storchi, 2004; Anderson, Allen and Browne, 2005; Muñuzuri *et al.*, 2005; Danielis, Rotaris and Marcucci, 2010; Van Duin, Tavasszy and Quak, 2013; van Schagen, Goh and de Souza, 2014) focused on solutions to improve the condition of cities through sustainable city logistics strategies.

Urban consolidation is one of the most popular city logistics strategies. It helps in rationalizing goods in consolidation centers which in turn helps in better utilization of

* Corresponding author: Surendra Reddy Kancharla (surendrareddy.kancharla@gmail.com)

freight vehicles. Better utilization of freight vehicles can lead to a reduction of truck traffic and thus reduce congestion, especially in city centers. Urban consolidation centers (Browne, Woodburn and Allen, 2007; Allen *et al.*, 2012; Daniela *et al.*, 2014; Kin *et al.*, 2016) and collaborative logistics (Ballot & Fontane 2010; Gonzalez-Feliu & Salanova 2012; Thompson & Hassall 2012; de Souza *et al.* 2014) are well studied and gaining ground in practice as well. A key decision in the above city logistic strategies is the scheduling and routing of vehicles. The nature of city logistics operations requires us to consider multiple depots and vehicle types.

Two of the main distribution strategies used in freight transport are direct shipping and multi-echelon shipping. In direct shipping, products are shipped directly to customers, whereas in multi-echelon shipping the products are routed through different intermediate locations, called satellites (or transshipment centers). Two-echelon distribution is the most common among the multi-echelon distributions. In two-echelon distribution, the delivery of goods to the satellites are done through large trucks and then smaller trucks deliver to the customers from satellites. The problem of routing vehicles in the two-echelon distribution system is called as Two-Echelon Capacitated Vehicle Routing Problem (2E-CVRP). The traditional 2E-CVRP problem is described as follows: each customer has to be visited exactly once and visiting a satellite is not mandatory, vehicles must carry loads less than or equal to their capacity, and a fixed number of vehicles are available at each satellite and depot.

Jacobsen and Madsen (1980) introduced the two-echelon routing problem, but the term 2E-CVRP was coined much later by Feliu *et al.* (2007). The formulation proposed by Feliu *et al.* (2007) was later strengthened using a family of inequalities in Perboli *et al.* (2010). The strengthened model performed better than the old one, and it has reduced the optimality gap in several instances and solved seven new instances to optimality. However, the formulation proposed by Feliu *et al.* (2007) and Perboli *et al.* (2011) provides loose upper bound for cases with more than two satellites. Jepsen *et al.* (2013) overcome this limitation with a flow-based formulation derived from the formulations of the Capacitated Vehicle Routing Problem (CVRP) and Split Delivery Vehicle Routing Problem (SDVRP). Recent study by Kancharla & Ramadurai (2019) proposed a new formulation relaxing the following assumptions: i) heterogeneous fleets with varying fleet sizes at each of the depot and satellite, ii) multi-depot in the first echelon, iii) allow vehicles to end at any satellite (depot) simultaneously conserving the total flow from each of these satellites (depots), iv) allows split-deliveries from multiple depots in the first echelon. Split-deliveries from multiple depots may not be commonly followed in real-world. In our formulation, we assume that split-deliveries does not happen in the first echelon.

In 2E-CVRP, VRP is solved in two echelons. Since VRP is NP-hard, solving this problem for real-world cases is computationally expensive. Hence, the heuristics solution is generally adopted. There are many heuristic solutions proposed for the 2E-CVRP. Few examples include, math-based heuristics (Perboli, Tadei and Vigo, 2011), hybrid ant colony optimization (Meihua *et al.*, 2011), Adaptive Large Neighborhood Search (ALNS) (Kancharla and Ramadurai, 2019), variable neighborhood search (Schwengerer, Pirkwieser and R. Raidl, 2012), Greedy Randomized Adaptive Search Procedure (GRASP) (Nguyen, Prins and Prodhon, 2012). All these heuristics decompose

the 2E-CVRP into either a CVRP or an MDCVRP and then solve the decomposed problem sequentially. Solutions from heuristics will be sub-optimal for the real-world instances; however, the tradeoff is between accuracy and the runtime of the algorithm. In real-world, the urban freight operators may have to take operational decisions within minutes. Hence, they may prefer algorithms which provide reasonably good solutions in relatively quick time over algorithms with better solutions but with a higher runtime.

The main contribution of this paper is a Simulated Annealing (SA) algorithm for the two-echelon routing problem considering multiple depots and heterogeneous fleets with the relaxation of assumptions mentioned above. We have evaluated the performance of SA on several 2E-CVRP and MD2E-CVRP instances and the results showed that the performance of SA was comparable to existing algorithms. We have also compared the performance of SA with Adaptive Large Neighborhood Search (ALNS) to evaluate the trade-off between runtime and solution quality of these algorithms.

The rest of the paper is organized as follows. Section 2 introduces the MD2E-CVRP and provides a mathematical formulation. Section 3 introduces the SA based heuristic for MD2E-CVRP. Test instances are introduced, and computational results are discussed in section 4, followed by conclusions in section 5.

2. Mathematical formulation

This problem is defined on a graph $G = (V, E)$ where $V = \{T, S, C\}$ is the set of vertices and $E = \{(i, j) : i, j \in V \text{ and } i \neq j\}$ is the set of edges. The set of vertices has three subsets where T is the set of depots, S is the set of satellites, and C is the set of customers. The set of edges has two subsets where the first subset has edges between the depots and satellites as well as edges between satellites and the second subset has edges between satellites and customers and between customers. The sets H and M are related to the types of vehicles used in the first and second echelons. Each edge (i, j) has a positive cost ξ_{ij} associated with it in the first echelon, similarly ξ_{kl} for edge (k, l) in the second echelon. Each customer has positive freight demand (d_i) associated with them. The total number of vehicles of type t with capacity ζ_h at any depot is denoted by γ_t^h , similarly total number of vehicles of type m with capacity ζ_m at each satellite is denoted by δ_s^m . Four decision variables are used in formulating this problem they are: $x1_{ij}^{th}$ is a binary variable equal to 1 iff the first echelon edge $(i \in S \cup T, j \in S \cup T)$ is used by vehicle $h \in H$ starting from depot $t \in T$, $x2_{kl}^{sm}$ is another binary variable equal to 1 iff the second echelon edge $(k \in S \cup C, l \in S \cup C)$ is used by vehicle $m \in M$ starting from satellite $s \in S$, $Q1_{ij}^{th}$ is the total freight flow through edge $(i \in S \cup T, j \in S \cup T)$ by vehicle type $h \in H$ starting from depot $t \in T$ in the first echelon, and finally, $Q2_{kl}^{sm}$ is the freight flow through edge $(k \in S \cup C, l \in S \cup C)$ by vehicle type $m \in M$ starting from satellite $s \in S$ in the second echelon. The mathematical model of the MD2E-CVRP is given below:

Objective function:

$$\min \sum_{t \in T} \sum_{h \in H} \sum_{\substack{i, j \in T \cup S \\ i \neq j}} \xi_{ij} x1_{ij}^{th} + \sum_{s \in S} \sum_{m \in M} \sum_{\substack{k, l \in S \cup C \\ k \neq l}} \xi_{kl} x2_{kl}^{sm} \quad (1)$$

Subject to:

$$\sum_{t \in T} \sum_{h \in H} \sum_{i \in \{t\} \cup S} x1_{ij}^{th} \leq 1, \quad \forall j \in S, i \neq j \quad (2)$$

$$\sum_{s \in S} \sum_{m \in M} \sum_{k \in \{s\} \cup C} x2_{kl}^{sm} = 1, \quad \forall l \in C, k \neq l \quad (3)$$

$$\sum_{j \in S} x1_{tj}^{th} \leq \gamma_t^h, \quad \forall t \in T, h \in H \quad (4)$$

$$\sum_{l \in C} x2_{sl}^{sm} \leq \delta_s^m, \quad \forall s \in S, m \in M \quad (5)$$

$$\sum_{j \in S} x1_{pj}^{ph} = \sum_{t \in T} \sum_{j \in S} x1_{jp}^{th}, \quad \forall p \in T, h \in H \quad (6)$$

$$\sum_{l \in C} x2_{pl}^{pm} = \sum_{s \in S} \sum_{l \in C} x2_{lp}^{sm}, \quad \forall p \in S, m \in M \quad (7)$$

$$\sum_{t \in T} \sum_{h \in H} \sum_{\substack{i \in T \cup S \\ i \neq j}} Q1_{ij}^{th} - \sum_{t \in T} \sum_{h \in H} \sum_{\substack{i \in S \cup T \\ i \neq j}} Q1_{ji}^{th} = \sum_{m \in M} \sum_{l \in C} Q2_{jl}^{jm}, \quad \forall j \in S \quad (8)$$

$$\sum_{s \in S} \sum_{m \in M} \sum_{\substack{k \in C \cup S \\ k \neq l}} Q2_{kl}^{sm} - \sum_{s \in S} \sum_{m \in M} \sum_{\substack{k \in S \cup C \\ k \neq l}} Q2_{lk}^{sm} = d_l, \quad \forall l \in C \quad (9)$$

$$\sum_{t \in T} \sum_{h \in H} \sum_{j \in S} Q1_{jp}^{th} = 0, \quad \forall p \in T \quad (10)$$

$$\sum_{s \in S} \sum_{m \in M} \sum_{k \in C} Q2_{kp}^{sm} = 0, \quad \forall p \in S \quad (11)$$

$$Q1_{tj}^{th} \leq \zeta_h x1_{tj}^{th}, \quad \forall j \in S, t \in T, h \in H \quad (12)$$

$$Q2_{sl}^{sm} \leq \zeta_m x2_{sl}^{sm}, \quad \forall l \in C, s \in S, m \in M \quad (13)$$

$$\sum_{i \in T \cup S} \sum_{t \in T} \sum_{h \in H} x1_{ij}^{th} \leq \sum_{l \in C \cup S} \sum_{m \in M} \sum_{\substack{k \in \{s\} \cup C \\ k \neq l}} x2_{kl}^{jm}, \quad \forall j \in S, i \neq j \quad (14)$$

$$x1_{ij}^{th} \in \{0,1\}, \quad \forall t \in T, h \in H, (i,j) \in S \cup T, i \neq j \quad (15)$$

$$x2_{kl}^{sm} \in \{0,1\}, \quad \forall s \in S, m \in M, (k,l) \in S \cup C, k \neq l \quad (16)$$

$$Q1_{ij}^{th} \geq 0, \quad \forall t \in T, h \in H, (i,j) \in S \cup T, i \neq j \quad (17)$$

$$Q2_{kl}^{sm} \geq 0, \quad \forall s \in S, m \in M, (k,l) \in S \cup C, k \neq l \quad (18)$$

The objectivefunction (1) minimizes the total distance traveled in both levels. Constraints (2)ensure a satellite is visited at most once, while constraints (3) ensure that each customer is visited exactly once. Constraints (4) limit the number of vehicles used from a depot for each vehicle type h to the maximum available vehicles at that depot.

Similarly, constraints (5) limit the number of the vehicles for a satellite. Constraints (6) and (7) allows vehicles from one depot/satellite to end at a different depot/satellite and at the same time conserves the total number of vehicles at each depot and satellite respectively. Constraints (8) and (9) ensure demand satisfaction at each satellite and customer respectively. Constraints (10) and (11) avoid residual loads in the final leg of any route. Constraints (12) and (13) impose vehicle capacity constraints at each level. Constraints (14) allow a first level route to connect to a satellite only if second level routes are starting from that satellite. Constraints (15)-(18) specify the domains of the variables.

3. Simulated Annealing Algorithm

Simulated Annealing (SA)(Kirkpatrick, Gelatt and Vecchi, 1983) is a local search meta-heuristic, which simulates the process of annealing used in metallurgy. SA improves the initial solution while slowly cooling the temperature. It accepts a new solution in the following two cases: i) if it is an improvement over the present, or ii) if it is worse, then with a probability determined by the Boltzmann function. This procedure of accepting a worse solution will prevent the algorithm from getting trapped at a local optimum during the initial stages of the search.

A separation strategy was applied to MD2E-CVRP to divide the problem into two MDCVRPs. We have developed a SA algorithm for an MDCVRP which was applied to the two sub-problems of MD2E-CVRP sequentially. First, the second level problem was solved to get the demands of each satellite based on the customers connected to that satellite. The first level problem was then solved using the satellite demand data obtained from the second echelon. A new neighborhood of the present solution was obtained at each iteration using three techniques: namely insertion, swap, and reversion. These techniques were selected randomly to escape from any local optimum. Insertion process selects two nodes randomly, and then the first node was inserted after the second node, whereas in the swap process, two nodes were selected randomly and swapped. Finally, in reversion, two nodes were selected, and the path between these nodes passing through all the nodes that are in between these two was reversed. The algorithm was allowed to find any solution (can be feasible or infeasible). However, we apply a heavy penalty for solutions that violate the constraints. The algorithm stores the first echelon solutions that consist of satellite demands and locations. We can retrieve a solution from the memory if the second echelon solution results in the same satellite demands and locations, thus helping in reducing runtime. The algorithm will stop if there was no improvement in solution for 150 consecutive temperature decrements or when the temperature falls below a final temperature. We have tested the algorithm on the 2E-CVRP instances and the MD2E-CVRP instances.

4. Computational Tests

In this section, we evaluate the computational efficiency and solution quality of the algorithm on different 2E- CVRP and MD2E-CVRP instances. All tests were performed on a PC with Intel Xeon E5-2430 @ 2.2 GHz with 8 GB RAM. The MILP model was coded in GAMS 23.3 and solved using CPLEX 12.1 solver. SA algorithm was written

in the Python programming language and tested on the same PC. The instances are described in the next subsection, followed by computational results.

4.1 Instance Sets

We have tested the model on 2E-CVRP instances introduced by Perboli et al. (2011) (set-2, set-3, and set-4) and eight MD2E-CVRP instances introduced by Kancharla & Ramadurai (2019). In the MD2E-CVRP instances, two types of vehicles are available in each echelon, and the number of vehicles available at each depot and satellite can be different. Hence, there can be a mix of vehicles for each depot and each satellite.

4.2 Computational Result

All the instances were run using CPLEX with a time limit of 10,000s. CPLEX was able to find optimal solutions for all the 22 customer instances, and for the 33 customer instances with an average gap of less than 5% compared to the best bound (obtained by CPLEX after linear relaxation).

Performance of SA depends on the following parameters - iterations at a particular temperature (T_{iter}), initial temperature (T_0), final temperature (T_f), temperature reduction rate (α), and the Boltzmann constant (k). To identify the ideal parameters for the present algorithm, we have tested three different values of T_{iter} , T_0 , and α making a total of 27 combinations on all the 22 node instances. Table 1 shows the results for these different combinations. We rank each combination in reverse order (the best solution gets a rank of 27, and the worst solution gets 1) based on the solution quality obtained at the end of the run and if two or more combinations have same solution quality then ranking is done based on the runtime. Summing up the ranks of a combination in all the instances will give the total score shown in column two. The columns three and four have the average value of the solution obtained and the average time taken for all the instances for that particular combination. The parameter values that gave the best results is the combination (C22) with the following values: $T_{iter} = 800$, $T_0 = 50$, and $\alpha = 0.95$. T_f and k are not varied, and the values taken are 0.001 and 1/9 respectively.

We have tested SA with the parameters mentioned above on the 2E-CVRP instances and the results reported are the best of three runs with the corresponding runtime. In Table 2 first column presents name of the instance, second column presents the solution obtained by CPLEX, third column presents the solution obtained by SA, fourth column presents gap between the solution obtained by CPLEX and SA, and finally, fifth column presents the run time for the SA. Overall, SA algorithm was able to find solutions with an optimality gap of less than 1% compared to the solution obtained by CPLEX, and it took an average time of 93s for set-2 and 115s for set-3 instances. The algorithm found eight new solutions, four each in set-2 and set-3 respectively. CPLEX was not able to find any feasible solution within a time limit of 10,000s on set-4 instances. Hence, we have compared SA with the Best-Known Solutions (BKS) from 2E-CVRP literature (Table 3), and the average optimality gap was found to be 4.53%. Figure 1 compares the performance of SA with ALNS(Kancharla and Ramadurai, 2019). From the figure, it is clear that runtime has a polynomial increase with instance size in the case of ALNS, whereas the increase is almost linear for SA. Since the increase in runtime is linear for SA, it can be used to solve large size real-world instances in a relatively short time

compared to ALNS. However, solution quality (refer Figure 2) of SA deteriorates with increase in instance size when compared to ALNS.

Table 1: Evaluation of SA parameter combination

| Combination (T_{iter}, α, T_0) | Total score | Average solution | Average time (s) |
|---|--------------------|-------------------------|-------------------------|
| C22 (800, 0.95, 50) | 253 | 448 | 75 |
| C20 (800, 0.90, 100) | 221 | 449 | 73 |
| C19 (800, 0.9, 50) | 206 | 449 | 64 |
| C23 (800, 0.95, 100) | 201 | 449 | 94 |
| C24 (800, 0.98, 30) | 196 | 449 | 84 |
| C26 (800, 0.98, 100) | 192 | 449 | 129 |
| C25 (800, 0.98, 50) | 186 | 449 | 99 |
| C13 (500, 0.95, 50) | 182 | 448 | 148 |
| C18 (800, 0.90, 30) | 179 | 449 | 63 |
| C10 (500, 0.90, 50) | 172 | 449 | 133 |
| C15 (500, 0.98, 30) | 172 | 449 | 135 |
| C11 (500, 0.90, 100) | 168 | 449 | 146 |
| C14 (500, 0.95, 100) | 168 | 448 | 167 |
| C16 (500, 0.98, 50) | 167 | 448 | 170 |
| C09 (500, 0.90, 30) | 155 | 451 | 121 |
| C21 (800, 0.95, 30) | 150 | 452 | 68 |
| C02 (100, 0.90, 100) | 136 | 449 | 650 |
| C01 (100, 0.90, 50) | 132 | 449 | 674 |
| C17 (500, 0.98, 100) | 125 | 450 | 200 |
| C00 (100, 0.90, 30) | 122 | 449 | 658 |
| C12 (500, 0.95, 30) | 116 | 451 | 126 |
| C03 (100, 0.95, 30) | 114 | 449 | 698 |
| C04 (100, 0.95, 50) | 105 | 448 | 683 |
| C05 (100, 0.95, 100) | 100 | 448 | 778 |
| C06 (100, 0.98, 30) | 88 | 449 | 705 |

We have also tested SA on MD2E-CVRP instances introduced by (Kancharla and Ramadurai, 2019), and the algorithm was able to find solutions with an average optimality gap of 2.34% (Table 4). Results show that runtime is increasing with the increase in satellites and depots even for the same number of customers. The algorithm improved solutions in two instances and the overall performance of SA is better on larger instances than on smaller instances compared to CPLEX.

Table 2: SA on set-2 and set-3 Instances

| Instance | Model | SA | Gap | Time (s) |
|-----------------|--------------|-----------|--------------|-----------------|
| Set-2 | | | | |
| E22-K4-S06-17 | 416.85 | 416.85 | 0.00 | 63 |
| E22-K4-S08-14 | 384.96 | 384.96 | 0.00 | 65 |
| E22-K4-S09-19 | 445.01 | 445.01 | 0.00 | 67 |
| E22-K4-S10-14 | 366.00 | 371.50 | 1.48 | 65 |
| E22-K4-S11-12 | 422.97 | 422.97 | 0.00 | 54 |
| E22-K4-S12-16 | 377.23 | 377.23 | 0.00 | 58 |
| E33-K4-S01-09 | 675.35 | 725.73 | 6.94 | 111 |
| E33-K4-S02-13 | 714.54 | 701.05 | -1.89 | 148 |
| E33-K4-S03-17 | 706.23 | 695.81 | -1.48 | 111 |
| E33-K4-S04-05 | 810.31 | 810.31 | 0.00 | 152 |
| E33-K4-S07-25 | 743.00 | 741.15 | -0.25 | 112 |
| E33-K4-S14-22 | 852.00 | 828.20 | -2.79 | 110 |
| Mean | | | 0.17 | 93 |
| Set-3 | | | | |
| E22-K4-S13-14 | 519.20 | 520.15 | 0.18 | 78 |
| E22-K4-S13-16 | 515.10 | 515.11 | 0.00 | 87 |
| E22-K4-S13-17 | 495.71 | 495.71 | 0.00 | 64 |
| E22-K4-S14-19 | 473.54 | 473.54 | 0.00 | 89 |
| E22-K4-S17-19 | 501.48 | 508.83 | 1.44 | 74 |
| E22-K4-S19-21 | 512.74 | 512.74 | 0.00 | 73 |
| E33-K4-S16-22 | 660.16 | 650.81 | -1.42 | 103 |
| E33-K4-S16-24 | 664.46 | 654.09 | -1.56 | 181 |
| E33-K4-S19-26 | 676.08 | 668.98 | -1.05 | 115 |
| E33-K4-S22-26 | 678.82 | 689.42 | 1.54 | 185 |
| E33-K4-S24-28 | 668.25 | 672.03 | 0.56 | 197 |
| E33-K4-S25-28 | 647.33 | 633.03 | -2.21 | 129 |
| Mean | | | -0.21 | 114.58 |

Table 3: SA on set-4 Instances

| Instance | BKS | SA | Gap | Time (s) |
|-----------------|------------|-----------|-------------|-----------------|
| Instance-50-19 | 1546 | 1575 | 1.84 | 176 |
| Instance-50-20 | 1272 | 1279 | 0.55 | 153 |
| Instance-50-21 | 1577 | 1616 | 2.4 | 124 |
| Instance-50-22 | 1281 | 1364 | 6.11 | 72 |
| Instance-50-23 | 1652 | 1717 | 3.79 | 66 |
| Instance-50-24 | 1282 | 1290 | 0.62 | 73 |
| Instance-50-25 | 1440 | 1546 | 6.84 | 202 |
| Instance-50-26 | 1167 | 1310 | 10.91 | 155 |
| Instance-50-27 | 1447 | 1504 | 3.79 | 165 |
| Instance-50-28 | 1210 | 1296 | 6.65 | 288 |
| Instance-50-29 | 1561 | 1569 | 0.51 | 220 |
| Instance-50-30 | 1211 | 1278 | 5.21 | 196 |
| Instance-50-31 | 1440 | 1513 | 4.83 | 206 |
| Instance-50-32 | 1199 | 1206 | 0.58 | 359 |
| Instance-50-33 | 1478 | 1549 | 4.61 | 204 |
| Instance-50-34 | 1233 | 1356 | 9.09 | 236 |
| Instance-50-35 | 1570 | 1598 | 1.75 | 208 |
| Instance-50-36 | 1228 | 1345 | 8.7 | 271 |
| Instance-50-37 | 1528 | 1644 | 7.06 | 211 |
| Instance-50-38 | 1163 | 1267 | 8.23 | 271 |
| Instance-50-39 | 1520 | 1569 | 3.14 | 279 |
| Instance-50-40 | 1165 | 1232 | 5.43 | 295 |
| Instance-50-41 | 1652 | 1715 | 3.68 | 425 |
| Instance-50-42 | 1190 | 1194 | 0.34 | 218 |
| Instance-50-43 | 1408 | 1422 | 0.98 | 330 |
| Instance-50-44 | 1035 | 1040 | 0.48 | 257 |
| Instance-50-45 | 1406 | 1489 | 5.57 | 397 |
| Instance-50-46 | 1058 | 1068 | 0.94 | 364 |
| Instance-50-47 | 1564 | 1590 | 1.64 | 230 |
| Instance-50-48 | 1074 | 1227 | 12.48 | 295 |
| Instance-50-49 | 1435 | 1548 | 7.29 | 221 |
| Instance-50-50 | 1065 | 1121 | 5 | 268 |
| Instance-50-51 | 1387 | 1405 | 1.28 | 396 |
| Instance-50-52 | 1103 | 1109 | 0.54 | 255 |
| Instance-50-53 | 1545 | 1671 | 7.54 | 234 |
| Instance-50-54 | 1113 | 1277 | 12.87 | 371 |
| Mean | | | 4.53 | 241 |

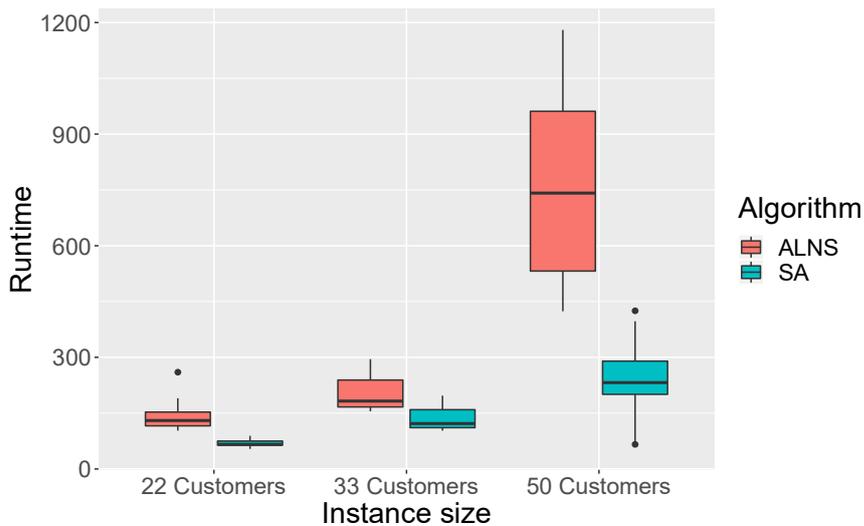


Figure 1: Scalability of SA algorithm compared to ALNS algorithm

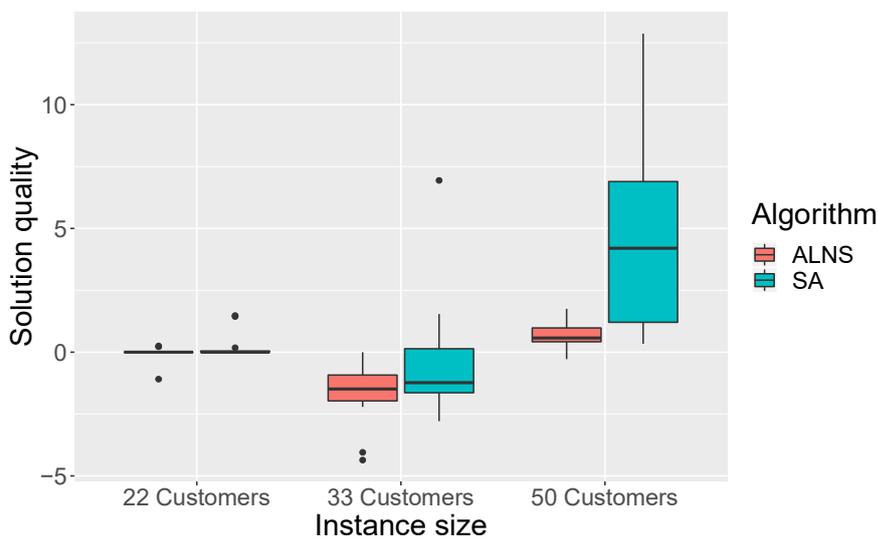


Figure 2: Solution quality of SA algorithm compared to ALNS algorithm

Table 4: SA on new instances

| Instance | CPLEX | SA | Gap | Time(s) |
|-----------|--------|--------|-------------|---------------|
| C16-S4-D2 | 136.51 | 136.51 | 0.00 | 210 |
| C18-S4-D2 | 175.69 | 194.69 | 9.76 | 226 |
| C24-S6-D2 | 220.62 | 241.32 | 8.58 | 590 |
| C28-S5-D2 | 226.29 | 228.53 | 0.98 | 488 |
| C28-S6-D3 | 210.79 | 214.22 | 1.6 | 525 |
| C30-S3-D1 | 275.61 | 285.93 | 3.61 | 417 |
| C30-S6-D3 | 295.53 | 281.05 | -5.15 | 617 |
| C36-S4-D2 | 223.63 | 222.21 | -0.64 | 660 |
| Mean | | | 2.34 | 466.62 |

5. Conclusion

We have introduced a new mixed-integer linear programming model for the Two-Echelon Capacitated Vehicle Routing Problem (2E-CVRP) that considers multiple depots and heterogeneous vehicle fleet at both levels. The MILP was solved on small to medium-sized instances using CPLEX solver. We have worked with three sets of 2E-CVRP instances, set 2, 3, and 4, and eight MD2E-CVRP instances from literature. CPLEX was able to find optimal solutions in few cases and upper bounds in the remaining cases.

We have also proposed a Simulated Annealing heuristic which not only has a comparable performance (gap < 2.4%) with significantly lower runtime for MD2E-CVRP cases, but also has competitive performance, less than 1% for up to 33 nodes and under 5% for 50 node instances, in the 2E-CVRP cases when compared to existing algorithms. Our algorithm has improved on eight solutions in set 2 and 3 instances, and two solutions in MD2E-CVRP instances compared to CPLEX. SA is approximately 60% faster than ALNS. However, the solution quality is approximately 4% worse compared to ALNS. This problem can be extended to consider electric vehicles in the second echelon, satellite synchronization, and time windows.

References

- Allen, J. *et al.* (2012) ‘The Role of Urban Consolidation Centres in Sustainable Freight Transport’, *Transport Reviews*, 32(4), pp. 473–490. doi: 10.1080/01441647.2012.688074.
- Anderson, S., Allen, J. and Browne, M. (2005) ‘Urban logistics - How can it meet policy makers’ sustainability objectives?’, *Journal of Transport Geography*, 13(1 SPEC. ISS.), pp. 71–81. doi: 10.1016/j.jtrangeo.2004.11.002.
- Ballot, E. and Fontane, F. (2010) ‘Reducing transportation CO2 emissions through pooling of supply networks: perspectives from a case study in French retail chains’, *Production Planning and Control*, 21(6), pp. 640–650. doi: 10.1080/09537287.2010.489276.
- Browne, M., Woodburn, A. G. and Allen, J. (2007) ‘Evaluating the potential for urban consolidation centres’, *European Transport - Trasporti Europei*, 35, pp. 46–63.
- Crainic, T. G., Ricciardi, N. and Storchi, G. (2004) ‘Advanced freight transportation systems for congested urban areas’, *Transportation Research Part C: Emerging Technologies*, 12, pp. 119–137. doi: 10.1016/j.trc.2004.07.002.
- Daniela, P. *et al.* (2014) ‘Reduced Urban Traffic and Emissions within Urban Consolidation Centre Schemes: The Case of Bristol’, in *17th Meeting of the EURO Working Group on Transportation*. Elsevier B.V., pp. 508–517. doi: 10.1016/j.trpro.2014.10.032.
- Danielis, R., Rotaris, L. and Marcucci, E. (2010) ‘Urban freight policies and distribution channels’, *European Transport - Trasporti Europei*, 46, pp. 114–146.
- Van Duin, J. H. R., Tavasszy, L. a. and Quak, H. J. (2013) ‘Towards E(lectric)-urban freight: First promising steps in the electric vehicle revolution’, *European Transport - Trasporti Europei*, (54), pp. 1–19. Available at: <http://hdl.handle.net/10077/8875>.
- Feliu, J. G. *et al.* (2007) ‘The Two-Echelon Capacitated Vehicle Routing Problem’, *Proceedings of the 22nd European Conference on Operational Research, Prague*,

- pp. 1–40.
- Gonzalez-Feliu, J. and Salanova, J. (2012) ‘Defining and Evaluating Collaborative Urban Freight Transportation Systems’, *Procedia - Social and Behavioral Sciences*, 39, pp. 172–183. doi: 10.1016/j.sbspro.2012.03.099.
- Jacobsen, S. K. and Madsen, O. B. G. (1980) ‘A comparative study of heuristics for a two-level routing-location problem’, *European Journal of Operational Research*, 5(6), pp. 378–387. doi: 10.1016/0377-2217(80)90124-1.
- Jepsen, M., Spoorendonk, S. and Ropke, S. (2013) ‘A Branch-and-Cut Algorithm for the Symmetric Two-Echelon Capacitated Vehicle Routing Problem’, *Transportation Science*, 47(1), pp. 23–37. doi: 10.1287/trsc.1110.0399.
- Kancharla, S. R. and Ramadurai, G. (2019) ‘Multi-depot Two-Echelon Fuel Minimizing Routing Problem with Heterogeneous Fleets: Model and Heuristic’, *Networks and Spatial Economics*. *Networks and Spatial Economics*, 19(3), pp. 969–1005. doi: 10.1007/s11067-018-9437-7.
- Kijewska, K., Konicki, W. and Iwan, S. (2016) ‘Freight Transport Pollution Propagation at Urban Areas Based on Szczecin Example’, *Transportation Research Procedia*, 14, pp. 1543–1552. doi: 10.1016/j.trpro.2016.05.119.
- Kin, B. *et al.* (2016) ‘Is there life after subsidy for an urban consolidation centre? An investigation of the total costs and benefits of a privately- initiated concept’, *Transportation Research Procedia*. Elsevier B.V., 12(June 2015), pp. 357–369. doi: 10.1016/j.trpro.2016.02.072.
- Kirkpatrick, S., Gelatt, C. . D. and Vecchi, M. . P. (1983) ‘Optimization by Simulated Annealing’, *Science*, 220(4598), pp. 671–680. doi: 10.1126/science.220.4598.671.
- Meihua, W. *et al.* (2011) ‘Hybrid Ant Colony Optimization Algorithm for Two Echelon Vehicle Routing Problem’, *Procedia Engineering*, 15(2010), pp. 3361–3365. doi: <http://dx.doi.org/10.1016/j.proeng.2011.08.630>.
- Muñuzuri, J. *et al.* (2005) ‘Solutions applicable by local administrations for urban logistics improvement’, *Cities*, 22(1), pp. 15–28. doi: 10.1016/j.cities.2004.10.003.
- Nguyen, V.-P. P., Prins, C. and Prodhon, C. (2012) ‘Solving the two-echelon location routing problem by a GRASP reinforced by a learning process and path relinking’, *European Journal of Operational Research*. Elsevier B.V., 216(1), pp. 113–126. doi: 10.1016/j.ejor.2011.07.030.
- Perboli, G., Tadei, R. and Tadei, R. (2010) ‘New families of valid inequalities for the two-echelon vehicle routing problem’, *Electronic Notes in Discrete Mathematics*. Elsevier B.V., 36(C), pp. 639–646. doi: 10.1016/j.endm.2010.05.081.
- Perboli, G., Tadei, R. and Vigo, D. (2011) ‘The Two-Echelon Capacitated Vehicle Routing Problem: Models and Math-Based Heuristics’, *Transportation Science*, 45(3), pp. 364–380. doi: 10.1287/trsc.1110.0368.
- van Schagen, J., Goh, M. and de Souza, R. (2014) ‘Collaboration in urban logistics: motivations and barriers’, *International Journal of Urban Sciences*, 18(2), pp. 278–290. doi: 10.1080/12265934.2014.917983.
- Schwengerer, M., Pirkwieser, S. and R. Raidl, G. (2012) ‘A Variable Neighborhood Search Approach for the Two-Echelon Location-Routing Problem’, in *Evolutionary Computation in Combinatorial Optimization: 12th European Conference, EvoCOP 2012*. Springer Berlin Heidelberg, pp. 13–24. doi: 10.1007/978-3-642-29124-1_2.
- de Souza, R. *et al.* (2014) ‘Collaborative Urban Logistics – Synchronizing the Last Mile a Singapore Research Perspective’, *Procedia - Social and Behavioral Sciences*. Elsevier B.V., 125, pp. 422–431. doi: 10.1016/j.sbspro.2014.01.1485.

Thompson, R. G. and Hassall, K. P. (2012) ‘A Collaborative Urban Distribution Network’, *Procedia - Social and Behavioral Sciences*, 39, pp. 230–240. doi: 10.1016/j.sbspro.2012.03.104.

Acknowledgements

The authors thank the Centre of Excellence in Urban Transport sponsored by the Ministry of Urban Development, Government of India and Shakthi Sustainable Energy Foundation for supporting this study.