International Journal of Industrial Engineering Computations 12 (2021) 221–234

Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

Designing optimal route for the distribution chain of a rural LPG delivery system

Purusotham Singamsetty[a] and Jayanth Kumar Thenepalle[b]*

[a]*Department of Mathematics, School of Advanced Sciences, VIT, Vellore-632014, Tamil Nadu, India*
[b]*Department of Science and Humanities, Sreenivasa Institute of Technology and Management Studies, Chittoor, Andhra Pradesh, India*

C H R O N I C L E

Article history:
Received January 9 2020
Received in Revised Format October 7 2020
Accepted November 23 2020
Available online November, 23 2020

Keywords:
Truncated vehicle routing problem
LPG delivery
Simultaneous pickup and delivery
Vehicle routing problem
Lexi-search algorithm

A B S T R A C T

A practical distribution system that arises in the context of delivering liquefied petroleum gas (LPG) through cylinders is considered in this study. To meet all the challenging constraints, the model is explicitly considered as a simultaneous pickup and delivery single commodity truncated vehicle routing problem with the homogeneous fleet of vehicles. The aim of this problem is to find the optimal routes for the set of vehicles locating at the distributing agency (DA), which offers simultaneous pickup and delivery operations over single commodity (i.e. LPG cylinders) to a fixed subset (need not serve all delivery centers) of delivery centers at rural level. The model is designed using zero-one integer linear programming. For proper treatment of the present model, an exact Lexi-search algorithm (LSA) has been developed. A comparative study is performed between the LSA and existing results for the relaxed version of the present model. Further, the efficiency of the LSA is tested through numerical experiments over small and medium CVRP benchmark test instances. The extensive computational results have shown that the LSA is productive and revealed that the real solutions have more consistent than the integral solutions in the presence of truncation constraint.

1. Introduction

The physical flow is the key activity in logistics and distribution systems, which involves the distribution of supplies from manufacturer to the consumers using a fleet of vehicles through distributing agencies/centers. The literature has termed this problem as the vehicle routing problem (VRP), introduced by Dantzig and Ramser (1959). The VRP looks for a collection of vehicle routes that begin and finish at the central depot, serve all the customers that (i) guarantee that a consumer is served by exactly one vehicle; the overall demand served by a vehicle does not surpass its capacity, and (ii) optimizes the total traversal cost/distance by these vehicles. Due to its wide applicability, the VRP has been extended to many practical variants that depend on the vehicle types, type of the delivered goods and quality of facility needed etc. A recent detailed extensive study on VRP, its variants, formulation, and solution techniques can be found in Sharma et al. (2018). One variant of the classical VRP is the vehicle routing problem with simultaneous pickup and delivery (VRPSPD), in which, the consumers require not only the supply of goods but also the instantaneous pickup of goods from them. A general assumption is that all supplied goods start from the depot and all pickup goods must be moved to the depot. Reverse logistics is a practice of moving the commodities/goods from the point of usage to the point of origin/manufacturer for refilling, recycling/refurbishing or waste disposal. For instance, in soft drinks distribution, the vehicle delivers the filled bottles to the retail shops and collects the utilized bottles from them in order to deposit at the industry. The same distribution strategy is also followed by the petroleum industries/distributing agencies in the context of distributing LPG gas through cylinders. Effective distribution of LPG cylinders to the delivery centers/customers at the rural areas that undertake by any distributing agency (DA) is still a challenging problem due to practical scenarios. Therefore, the present study concerns the distribution system of LPG cylinders

* Corresponding author
E-mail: jayanth.maths@gmail.com (J. K. Thenepalle)

2021 Growing Science Ltd.
doi: 10.5267/j.ijiec.2020.11.001

in rural areas and its effective treatment. Here, the fleet of vehicles start with filled LPG cylinders, which are to be delivered to the delivery centers and simultaneously the same vehicle, has to collect the similar amount of the empty cylinders in order to deposit at the DA. Since the LPG distribution is to be done through metallic cylinders that have less damage and more durability and the empty cylinders can be recovered to the maximum extent. Several researchers have addressed distribution problems based on the nature of the supplied products, such as fresh milk (Tarantilis & Kiranoudis, 2001), fresh meat (Tarantilis & Kiranoudis, 2002), soft drinks (Privé et al. 2006), oil tankers (Ahmed et al. 2012), free newspaper (Archetti et al. 2013), and confectionaries (İnanlı et al. 2015).

Reviewing the literature of VRPSPD, the VRPSPD was first studied by Min (1989). Since then, it has relatively gained much attention and developed several solution techniques in past few decades. To mention, an insertion based heuristic approach (Dethloff, 2001), Tabu search (TS) (Montané & Galvao, 2006), a hybrid insertion-based metaheuristic algorithm(Chen & Wu, 2006), a hybrid tabu search together with a variable neighborhood technique (Bianchessi & Righini, 2007), a reactive tabu search (TS) (Wassan et al. 2008), Particle swarm optimization (PSO) (Ai & Kachitvichyanukul, 2009), a guided local search together with tabu search (Zachariadis et al. 2009), a parallel heuristic approach integrated with variable neighborhood descent method (Subramanian et al. 2010), Genetic algorithm (GA) (Tasan & Gen, 2012). Besides to this, some of the exact algorithms namely, branch-and-cut algorithm and branch-cut-and-price algorithm were also developed (Subramanian et al. 2011; Subramanian et al. 2013) and shown that these methods are capable of solving the instances of sizes up to 100 customers. Due to its wide applicability, the constrained VRPSPD has also received great attention. To mention, The VRPSPD with time constraints has been studied, modelled as a mixed integer programming by Polat et al. (2015) and proposed perturbation based neighborhood search (PVNS), a heuristic approach that depends on the perturbation mechanism, variable neighborhood search (VNS), and the savings algorithm. The VRPSPD with time windows is studied by Wang et al. (2015) and developed a parallel-simulated annealing (PSA) for efficient solutions. To solve VRPSPD, Kalayci & Kaya (2016) adapted inter and intra-route neighborhood structures from Polat et al. (2015) and developed a hybrid metaheuristic algorithm called Ant colony system algorithm (ACS) based variable neighborhood search.

From the extensive literature, it is note that most of the works on VRPSPD deals with an assumption that the DA has to serve all the delivery centers. However, there are certain situations in which the DA will be able to serve partial (out of all) delivery centers due to practical constraints (for instance, less availability of cylinders at DA). In such case, the DA can either stop providing the service to all or serve partial delivery centers. Instead of stop providing the service to all, serving atleast few delivery centers helps DA in many aspects. By understanding the practical utility, several researchers have studied truncated version of Travelling Salesman Problem (TSP) (Gensch, 1978; Volgenant & Jonker, 1987; Bhavani & Murthy, 2006; Giardini & Kalmar-Nagy, 2011). Here, the truncated version of TSP represents the classical TSP with an assumption that the salesman need not visit all the cities, instead it is enough to cover a subset of cities. The study of Stetsyuk (2016) have experimentally shown that finding the $k-$node $(n \leq k)$ shortest route is more challenging than determining the optimal Hamiltonian route of length n. Recently, Thenepalle & Singamsetty (2018) studied the bicriteria version of truncated TSP and solved using Lexi-search algorithm as well as Tabu search approach. The study has shown that the solution produced by TS may not be reasonably good on the truncation aspect, as the initial solution does not include all the nodes in the sub-tour and difficult to perform the swapping with uncovered nodes in the sub-tour. Hence, high-level solution procedures are needed to solve a truncated version of TSP related problems. In addition to the TSP, distinct combinatorial problems involving truncation constraint such as *k*-cardinality assignment problem (Belik & Jörnsten, 2016), *k*-cardinality minimum spanning tree problem (Katagiri et al., 2012; Kumar & Purusotham, 2018) have been studied. Further, Xu et al. (2018) addressed the practical utility of the truncation constraint in the context of underground mining. This motivates to adapt the truncation constraint in the context of real time distribution systems.

With this motivation, in this study, a truncated version of VRPSPD that simultaneously performs pickup and delivery tasks using a homogeneous fixed fleet of vehicles over a single commodity (LPG Cylinders) in the rural areas is considered. Although, the proposed model physically appears VRP with simultaneous pickup and delivery tasks, but due to the assumption that the amount of filled LPG cylinders delivered to the delivery centers is same as the quantity of empty cylinders collected back from them, the truncated version of VRPSPD can be viewed as a capacitated truncated vehicle routing problem (CTVRP). The CTVRP is closely related to team orienteering problem (TOP), an extensively studied problem in the family of VRP with profits (Keshtkaran et al. 2018). It is interesting to note that a truncated version of CVRP has not been addressed in the literature. An exact Lexi-search algorithm (LSA) is presented to solve the CTVRP. Numerical experiments are carried out on CVRP benchmark instances in order to check the effectiveness of the proposed LSA. The proposed VRP model can be seen as a combination of two problems, that are (i) capacitated vehicle routing problem (CVRP) (ii) capacitated truncated vehicle routing problem (CTVRP). The arbitrary solutions of the CVRP and CTVRP are demonstrated in Fig. 1 and Fig. 2, respectively. Figure 1 represents an arbitrary solution of CVRP to an example in which three vehicles are positioned at DA offered services to 10 delivery centers (including DA). Similarly, Fig. 2 depicts an arbitrary solution of the CTVRP in which 3 trucks based at DA offered delivery services to 8 (out of 10) delivery centers including DA. It is note that in Fig. 2, both the pickup and delivery operations are involved.

The paper is organized as follows: The problem statement and its mathematical model is presented in Section 2. In Section 3, the preliminaries and proposed LSA are described. A numerical example is illustrated in Section 4. Experimental results are summarized in Section 5. Finally, conclusions are drawn in Section 6.

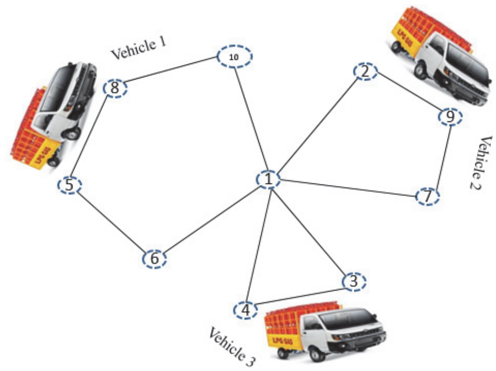

Fig. 1. Illustration of CVRP

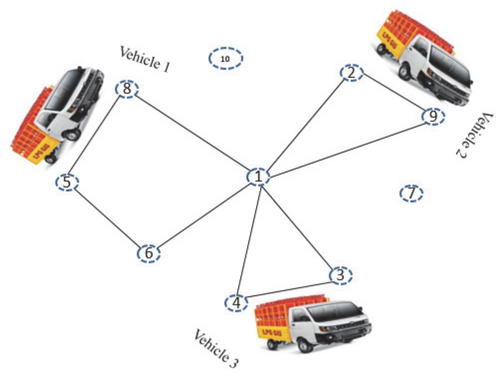

Fig. 2. Illustration of CTVRP

2. Problem description and mathematical model

Consider a logistics network $G(N, E)$, where $N = \{1, 2, ..., n\}$ be the set of delivery center/customer nodes including a unique distribution agency (DA) (say $\delta, \delta \in N$) and E denotes the edge set $(E = \{(i, j) / i, j \in N\})$. Let $K = \{1, 2, ..., m\}$ be a set of m homogeneous vehicles with capacity (say CV) positioned at the DA to serve the delivery centers. A non-negative symmetric travel distance/cost c_{ij} (in units) is associated with each edge $(i, j) \in E$. Each node other than DA needs certain amount of filled LPG cylinders to be supplied from the DA and simultaneously, collects the delivered amount of empty cylinders at each delivery center in order to deposit at the DA. The fleet of vehicles start from the DA and need not serve all the delivery centers (say n), instead, an arbitrary subset with fixed cardinality (n_0) of delivery centers are to be served without violating the vehicle capacity. Each delivery center is served at most once by single vehicle. The aim of the problem is to establish the route plan for a fleet of vehicles such that the overall routing distance is minimized. This problem is referred to as *capacitated truncated vehicle routing problem* (CTVRP), which allows both pickups and deliveries simultaneously. The model CTVRP formulated as a 0-1 integer linear program (0-1 ILP). Note that, in this study the distance, capacity of a vehicle, demand values are represented as arbitrary units. The assumptions, which are used to formulate CTVRP are given below, follows the mathematical model.

Assumptions

- The data such as the pickup and delivery demands, number of vehicles, vehicle capacities and the distance between the delivery centers are predetermined.
- No split pickups, partial deliveries, and transshipments are allowed.
- No on the spot (instantaneous) pickups and deliveries are considered.
- Bulk availability at the DA to serve the specified number of delivery centers.
- All vehicles start with filled cylinders, serves filled cylinders to a desired number of delivery centers, collects back the empty cylinders and returns to DA.
- Filled cylinders delivered first and empty cylinders picked up next.
- No time window is considered for service at any delivery center.
- No restriction on overall distance (in units) travelled by any vehicle on any route.
- No restriction on the number of delivery centers served by any vehicle.
- No delivery quantity (d_j) and pickup quantity (p_j) at each delivery center of a route exceed the vehicle's capacity, i.e. $d_j \le CV$, $p_j \le CV \ \forall j \in N$.
- As the model concerns the homogeneous fleet of vehicles, once the optimal route plan is obtained, any vehicle can be designated to any route.

Under these assumptions, the model described above formulated as 0-1 ILP as follows:

$$\min Z = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} c_{ij}\, x_{ij}^{k} \tag{1}$$

subject to the constraints

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m} x_{ij}^{k} = n_0 + m - 1 \qquad (2)$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m} x_{\delta j}^{k} = \sum_{i=1}^{n}\sum_{k=1}^{m} x_{i\delta}^{k} = m \qquad (3)$$

$$\sum_{i=1}^{n}\sum_{k=1}^{m} x_{ij}^{k} \le 1, \forall j \in N / \{\delta\} \qquad (4)$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m} x_{ij}^{k} \le 1, \forall i \in N / \{\delta\} \qquad (5)$$

$$\sum_{i=1}^{n} x_{iq}^{k} - \sum_{j=1}^{n} x_{qj}^{k} = 0, \forall k \in K,\ q \in N,\ i \ne q \qquad (6)$$

$$d_j \sum_{i=1}^{n} x_{ij}^{k} = d_j, \forall j \in N, k \in K \qquad (7)$$

$$\sum_{i=1}^{n} d_i \sum_{j=1}^{n} x_{ij}^{k} \le CV, \forall k \in K \qquad (8)$$

$$p_j \sum_{i=1}^{n} x_{ij}^{k} = p_j, \forall j \in N, k \in K \qquad (9)$$

$$\sum_{i=1}^{n} p_i \sum_{j=1}^{n} x_{ij}^{k} \le CV, \forall k \in K \qquad (10)$$

$+$ Subtour elimination constraints (11)

$$x_{ij}^{k} \in \{0,1\}, \forall\ i, j \in N,\ k \in K,\ i \ne j \qquad (12)$$

In the mathematical model, (1) refers to the objective function that aims to minimize the overall distance travelled. Constraints (2) guarantee that a feasible schedule consists of $m + n_0 - 1$ arcs that form m routes and serves exactly n_0 delivery centers including central depot. Constraint (3) ensures that the set of vehicles departs from the depot is equals to the set of vehicles returns to the depot. Constraint set (4-5) represent that a vehicle enters the delivery center and departs from it at most once, respectively, but these constraints does not guarantee the continuity of the trip. To do so, constraint (6) has been used that guarantees that each delivery center is to be visited at and departed by the same vehicle. Constraint set (7) does not allow the split deliveries. Constraint set (8) represent that the delivery loads of a route do not surpass the vehicle's capacity. In fact, the present model involves pickup operations; this is not reflected in the constraint set (7-8). Although $d_j = p_j \ \forall j = 1, 2, ..., n$, for better understanding, we introduce the constraint set (9-10), in which the partial pickups are not allowed and the pickup loads of a route does not exceed the vehicle's capacity. Since the constraint (6) does not control the appearance of the illegal/ sub tours (i.e. For instance, 3-5-6-3 indicate an illegal tour, as it does not include the DA), to eliminate all such tours, a constraint (11) called subtour elimination constraint is considered. Finally, (12) enforces the binary constraints on the decision variables. Without loss of generality, if $n_0 \to n$, then the model CTVRP reduces to CVRP. Therefore, the constraint (2) turns into the constraint (13).

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m} x_{ij}^{k} = m + n - 1. \qquad (13)$$

3. Preliminaries of LSA

Obviously, solving combinatorial optimization problems (COP) using exact methods is highly expensive. Thus, the interest from the researchers has been growing in developing meta-heuristic approaches to tackle COP that may provide best or near-optimal solutions. In fact, effective decisions can be made through exact solutions. In this aspect, developing exact search methods have also been receiving much attention. The lexicographic search algorithm is one such method, which employs the concept of construction of words. The entire search process for an optimal schedule can be done in a systematic manner that is analogous to the search of a required word in a dictionary; thus, the name is given as "Lexi-search".

The main difficulty of solving COP using implicit enumeration methods is (i) verifying the feasibility (ii) setting effective bounds. There is a difficulty in testing the feasibility for a few problems. To overcome this, a pattern recognition technique based Lexi-search approach (Murthy, 1976) has been developed and stated as follows:

"*A unique pattern is connected with each solution of a problem. Partial pattern represents a partial solution. An alphabet-table characterizes with the assistance of which the words, representing the pattern are listed in a lexicographic or dictionary order. During the search for an optimal word, when a partial word is considered, first bounds are determined and then the partial words for which the value is less than the trail value are checked for the feasibility**"**.*

The basic concepts associated with Lexi-search algorithm (LSA) are described below.

3.1. Pattern

A two-dimensional matrix X related to the solution is called a pattern. The values of the pattern X calculated using the formula (14). Where, $V(X)$ defines the value of the objective function and gives the overall distance represented by the solution X . Here, the terms pattern and solution refers the same.

$$V(X)=\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m}c_{ij}\,x_{ij}^{k} \tag{14}$$

3.2. Alphabet table

Typically, there exists n^2 ordered pairs for any two dimensional distance matrix $C=[c_{ij}]$ of order n . Since the model concerns about the symmetric distance matrices, either upper or lower triangular matrix is enough to generate the alphabet table. It is known that the upper or lower triangular matrices includes the principal diagonal elements also, but in the present model those elements are neglected as they forms self-loops. Therefore, for any symmetric distance matrix of order n consists of $(n^2-n)/2$ ordered pairs and only those are considered to generate the alphabet table. All these ordered pairs are to be organized in an increasing manner subject to their distances and are labelled from 1 to $(n^2-n)/2$. Let $SN=\{1,2,...,(n^2-n)/2\}$, a set of $(n^2-n)/2$ indices and C^* be an array of distance elements defined in such a way that if $a_1,a_2\in SN$ and $a_1<a_2$ then $C^*(a_1)\le C^*(a_2)$. The arrays CC,R and CL represents the cumulative distance, row and column (i.e. indices) of the distance element in C , respectively. For understanding, if $a_k\in SN$ then $C^*(a_k)=C(R(a_k),CL(a_k))$ be the distance in the position of $(R(a_k),CL(a_k))$ from the distance matrix, and $CC(a_k)=\sum_{i=1}^{k}C^*(a_i)$. The *alphabet table* can be formed by augmenting the arrays SN,C^*,CC,R and CL together. Let $L_r=(a_1,a_2,a_3,...,a_r),\ a_i\in SN$, be an ordered sequence of r indices from SN . The pattern represented by the ordered pairs is independent of a_i in the sequence given by L_r . For uniqueness, the indices in L_r are arranged in ascending order, such that $a_i<a_{i+1},\ i=1,2,3,...,r-1$. The ordered sequence of r indices L_r from SN is called a word of length r . A word L_r is a sensible word if $a_i<a_{i+1},\ i=1,2,3,...,r-1$; otherwise, it is non-sensible. L_r is said to be a feasible word if it represents a feasible pattern; otherwise, it is infeasible. Any one of the indices from SN can take up the prime position in the partial word L_r . In this model, the set of m vehicles wishes to serve exactly $n_0(\le n)$ delivery centers that also includes DA. Therefore, for a feasible solution needs $m+n_0-1$ arcs which forms m closed routes and serves exactly n_0 delivery centers including DA. If $r<m+n_0-1$, then L_r is said to be a *partial/incomplete word*, whereas $r=m+n_0-1$ represents a *full length feasible word* or simply a *word*.

3.3. Setting effective bounds

Let $VT=9999$ (a sufficiently large value) be the trial solution and is considered as an upper bound. The lower bound (LB) of a word L_r is evaluated using $LB(L_r)=V(L_r)+CC(a_r+B-r)-CC(L_r)$ where $B=m+n_0-1$. The value of the word $V(L_r)$ can be determined recursively using $V(L_r)=V(L_{r-1})+C^*(L_r)$ with $V(L_0)=0$.

3.4. Lexi-search algorithm

The systematic procedure of Lexi-search algorithm (LSA) is described as follows:

Step 1: *Initialize*: distance matrix $C=[c_{ij}]$, set $VT=9999$ (trial value) and required parameters such as m,n,n_0, and CV .

Step 2: **Construction of alphabet table:**
Generate the alphabet table by sorting the elements of the distance matrix along with the necessary arrays of indices as described in Section 3.2. Return to Step 3.

Step 3: **Computation of bounds:**

a. Initially, the algorithm begins with a partial word of length 1, i.e. $L_r=(a_r)=1$, where $r=1$.
b. Compute the lower bound (LB) of a partial word (L_r) as discussed in Section 3.3.

c. If $LB < VT$, then go to Step 4, otherwise, drop the partial word L_r, discard the word with L_r as leader, since it does not provide optimal solution and thus, reject all the partial words of order r that succeeds L_r, go to Step 6.

Step 4: **Feasibility checking:**

a. If the partial word L_r holds the feasibility criteria, such as balancing the size of the truncation parameter (n_0) (constraint (2)), assuring m tours and preserving the continuity between the customers (constraints (3-6)), vehicle capacity restrictions (constraint set (7-10)), preventing formation of sub tours (constraint (11)), and satisfying the binary variable (constraint (12)), then it is feasible otherwise, it is infeasible.

b. If L_r is feasible, then accept it and continue for the next partial word of order $r+1$ and move to Step 5.

c. If L_r is infeasible, then consider the next partial word of order r by considering a new letter that succeeds a_r in its r^{th} place, go to Step 3b.

Step 5: **Concatenation:**

a. If L_r is a full-length feasible word (i.e. $r = m + n_0 - 1$), then $VT = LB(L_r)$, record VT and L_r and for further improvement go to Step 7.

b. If L_r is a partial feasible word, then it can be concatenated by using $L_{r+1} = L_r * (a_{r+1})$ where * specifies the string operation and go to Step 3b.

Step 6: If all the words of order r are exhausted and the length of the word L_r is 1, then go to Step 8. Otherwise, go to Step 7.

Step 7: **Backtracking:**

a. In order to explore the solution space, backtracking is performed by assuming current VT as the upper bound and continues the search with the next letter of the partial word of order $r-1$ and go to Step 3b.

b. Repeat the Steps 3b to 7, eliminate the feasible/infeasible solutions which are not useful to achieve the optimal solution. Continue this process until VT has no further improvement and go to Step 8.

Step 8: Stop.

Finally, at the end of the search, VT provide the optimal solution, the latest complete word L_r give the position of the letters and one can find the optimal schedule for connectivity of given delivery centers with the help of L_r.

4. Numerical example of CTVRP

A benchmark instance *P-n21-k2* of size 21 delivery centers (i.e. 20 delivery centers and one DA/ central depot) with two homogeneous vehicles of capacity 160 units, is considered from the CVRP library (http://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-instances/). The location of delivery centers and a central depot on Euclidean plane is shown in Fig. 3, the respective locational coordinate points (X, Y) along with their demand quantities are given in Table 1. Note that pickup and delivery demand values are identical and the first coordinate $(30, 40)$ is assumed as DA $(\text{i.e. } \delta = 1)$. A fleet of vehicles initiated at the central depot All the delivery centers served with, supply filled LPG cylinders to the delivery centers and simultaneously collects empty cylinders from them. The geometric distance (in units) between any two delivery centers say (X_i, Y_i) and (X_j, Y_j) is computed by using the usual Euclidian distance formula $c_{ij} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}$. It is important to note that, as the present model deals with a homogeneous fleet of vehicles, once we arrive with an optimal route plan, any vehicle can be designated to any route without loss of generality.

Table 1
The coordinates of delivery centers and their demands

Index	1	2	3	4	5	6	7	8	9	10	11
X	30	37	49	52	31	52	42	52	57	62	42
Y	40	52	49	64	62	33	41	41	58	42	57
Demand (Number of Cylinders)	0	7	30	16	23	11	19	15	28	8	8
Index	12	13	14	15	16	17	18	19	20	21	
X	27	43	58	58	37	38	61	62	63	45	
Y	68	67	48	27	69	46	33	63	69	35	
Demand (Number of Cylinders)	7	14	6	19	11	12	26	17	6	15	

The distance matrix (C) is generated by collecting all the geometric distances up to four precisions between each pair of locations. The alphabet table is constructed for the distance matrix by arranging the distance elements in ascending order (the process is same as discussed in Section 3.2). First, this benchmark instance is solved using LSA in the absence of truncation $(\text{i.e. } n_0 = n)$ and compared with the available best-known solution. It is seen that the optimal fractional solution found by LSA is 212.7115 units, whereas the available best-known integral solution (i.e. the solution for the distance matrix with integral entities) for this instance is observed as 211 units. The deviation caused between two solutions i.e. 1.7115 units is due to rounding the distance matrix entries to its nearest integral values. However, the final route plan of the optimal fractional solution is same as that of the route plan of the best-known integral solution available at the CVRP library. The optimal route plan may change accordingly with the integral and non-integral values of the distance matrix. It is note that the best-known integral solutions are observed when the geometric distances are rounded to its nearest integral values. This result shows the efficiency of the LSA in producing the optimal solutions. Figure 4 demonstrates the optimal LPG distribution route plan for the two vehicles when the DA is able to serve all the delivery centers. The optimal route plan for the two vehicles is given by Route 1: depot–21–6–15–18–10–14–3–8–7–depot and Route 2: depot–17–2–11–9–19–20–4–13–16–12–5–depot.

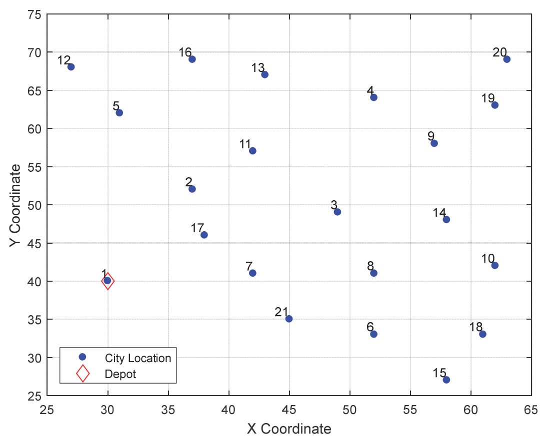

Fig. 3. Location of delivery centers and central depot

Due to scarcity of filled cylinders at DA, it is difficult to serve all the delivery centers. Instead, it can supply to a desired number of delivery centers $(n_0 < \text{n})$ with its maximum availability. To study this case, we considered the same benchmark instance (*P-n21-k2*) with the assumption that the DA is capable to serve any of the 16 delivery centers $(\text{i.e. } n_0 = 16 + DA = 17)$ out of 20 delivery centers. The optimal LPG distribution route plan found by LSA for the two vehicles with 17 delivery centers is shown in Fig. 5. The overall distance travelled by the two vehicles is 173.0046 units and the corresponding route plan is given by Route 1: depot–17–2–11–13–16–12–5–depot and Route 2: depot–21–6–15–18–10–14–3–8–7–depot. If the distance matrix entries are rounded to nearest integral values, then the best found integral solution for the same case is observed as 172 units. However, there is a deviation (i.e. 1.0046 units) between the integral and fractional solutions, the optimal route plans are observed the same. Since the two vehicles are independent and have the same capacity, thus any vehicle can be designated to any route.

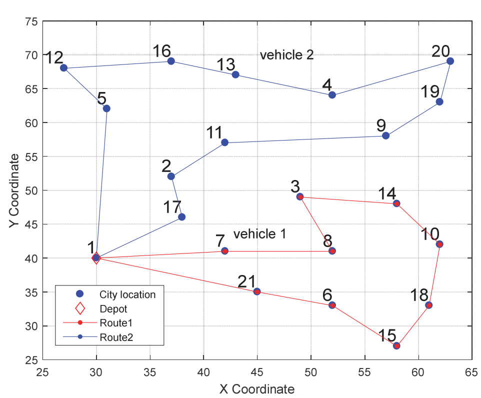

Fig. 4. Optimal LPG distribution route plan of CVRP

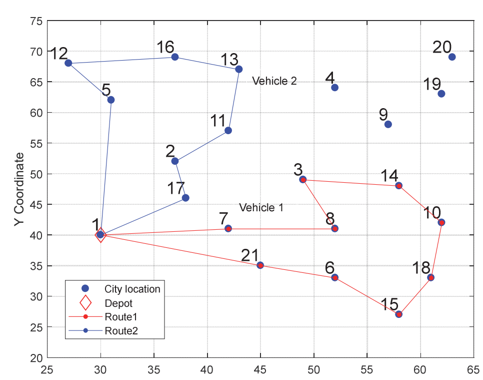

Fig. 5. Optimal LPG distribution route plan of CTVRP

5. Computational analysis

As the benchmark instances of CTVRP are not available, the CVRP benchmark instances are considered for computational experiments. This section provides the numerical experiments of the LSA on CVRP test instances. All computational experiments are carried out by implementing the proposed LSA in Matlab 2017a and tested on Intel(R) Core i3 processor, 2.0 GHz PC with 4 GB of RAM, Windows 10 Operating System.

5.1. Comparison between LSA and best known results of CVRP

To test the LSA, we used five well-known datasets of CVRP (composed of 16 benchmark instances taken from the datasets A, B, E, P, and V) available at http://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-instances/ (accessed 05 July 2020). The considered datasets are symmetric with vehicle capacity constraints and number of available vehicles restrictions. Each instance has the following terminology: the principal term designates the source of the instance and the second term represents the number of delivery centers followed by the number of vehicles required. For better understanding, in the instance *A-n32-k5*, *A* denotes name of the CVRP dataset, *n32* represents the size of the problem i.e. 32 delivery centers (including DA) and the last term *k5* indicates the number of vehicles involved i.e. 4. The comparative study is made between the best-found solutions by LSA and the best-known optimal solutions for 16-benchmark test instances. Note that those best-known optimal solutions are observed from the datasets of CVRP, in which the Euclidean distances (i.e. distances between customers) are rounded to the nearest integral values. For each instance, the best-known integral solution, best integral solution and best fractional solutions found by LSA, error value and their percentages, CPU runtime, and the ratio of demand to capacity are reported in Table 2. The observations are given as follows:

- Most of the earlier studies relied on integral solutions; therefore, both integral and fractional optimal solutions are presented.
- The integral solution found by proposed LSA matched with the best-known integral solution for all 16-test instances. This shows that LSA is potential in obtaining exact solutions.
- The results reported in column 7 represents the optimal solutions when the distance matrix entries are rounded to four precisions. These solutions are not available in the CVRP library.
- To measure the superiority of the solutions, the error value (i.e. error = Opt_Frac − Best known), error percentage between the best-known integral optimal solution and fractional optimal solution (*Opt_Frac*) found by LSA is computed using the formula (15) and those are reported in the columns 8 and 9, respectively.

$$\text{Error}\,\% = \frac{|\,Best\,known - Opt\_Frac\,|}{Opt\_Frac}\,\% \tag{15}$$

- The error percent values vary from 0 % to 0.85%. This shows that fractional optimal solutions found by LSA are more accurate than the known integral optimal solutions. However, there is a deviation between the best-known integral solution and best-found fractional optimal solution, their optimal route plans for the set of vehicles remains the same for considered 16 instances, but this cannot be true for all the cases.
- The CPU runtime required to obtain the optimal solutions are given in column 10, it is ranging from 10 seconds to 1652 seconds. Although the runtimes seem to be little expensive, but are practically considerable from the solution quality viewpoint.
- It is interesting to note that the developed LSA relies on the data structure. For instance, the LSA solved the problem E-n23-k3 within 60 seconds, whereas the problem E-n22-k4 takes 676 seconds to obtain the optimal solution. This show that apart from the problem size, the structure of the data also influence the efficiency of the LSA. To visualize the same, a scatter diagram is plotted between the size of the instance and CPU runtime, which is shown in Fig. 6. In Fig. 6, the scatter plot with correlation of +0.7859 clearly indicates that there is moderate linear correlation between the size of the instance and CPU runtime.
- The last column shows that all problem instances have tight vehicle capacity constraints in which the ratios of demand to the capacity determined using (16) below are near to 1, except the E-n13-k4 and E-n23-k3 instances, which are equivalent to 0.75.

$$\text{Ratio of demand to capacity} = \sum_{i=1}^{n} d_i / (m \times CV) \tag{16}$$

Here, d_i is the delivery demand at the delivery center i, m is the number of vehicles, and CV is the vehicle capacity.

- The overall results show the developed LSA is productive in getting optimal solutions within considerable CPU runtimes.

Table 2
Comparison of LSA results with known integral optimum for CVRP instances

Instance	*n*	*m*	CV	Best known	Opt_Int	Opt_Frac	Error	Error %	CPU time (s)	RDC
A-n32-k5	32	5	100	784	784	787.8082	3.8082	0.48	1124	0.82
A-n36-k5	36	5	100	799	799	802.1318	3.1318	0.39	1652	0.88
B-n31-k5	31	5	100	672	672	676.7583	4.7583	0.70	896	0.82
B-n34-k5	34	5	100	788	788	790.8051	2.8051	0.35	924	0.91
E-n13-k4	13	4	6000	247	247	247	0	0	10	0.75
E-n22-k4	22	4	6000	375	375	375.2798	0.2798	0.07	676	0.94
E-n23-k3	23	3	4500	569	569	568.5625	0.4375	0.07	60	0.75
E-n33-k4	33	4	8000	835	835	838.7212	3.7212	0.44	1126	0.92
P-n19-k2	19	2	160	212	212	212.6569	0.6569	0.30	285	0.97
P-n20-k2	20	2	160	216	216	217.4156	1.4156	0.65	440	0.97
P-n21-k2	21	2	160	211	211	212.7115	1.7115	0.80	326	0.93
P-n22-k2	22	2	160	216	216	217.8522	1.8522	0.85	482	0.96
P-n23-k8	23	8	40	529	529	531.1739	2.1739	0.40	1508	0.98
gr-n17-k3	17	3	6	2685	2685	2685	0	0	164	0.88
gr-n21-k3	21	3	7	3704	3704	3704	0	0	424	0.95
gr-n24-k4	24	4	7	2053	2053	2053	0	0	512	0.82

n – number of city locations; *m* – number of vehicles; CV – capacity of a vehicle; Best known – best known integral optimum solution; Opt_Int – optimal

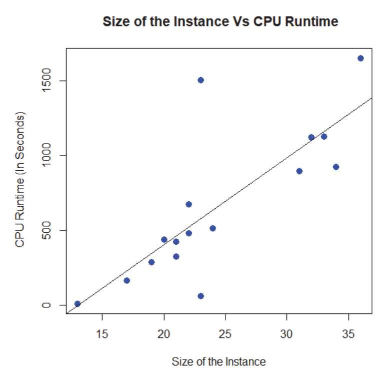

Fig. 6. Scatter plot with correlation +0.7859

5.2. *Computational results of LSA for CVRP instances*

To measure the efficiency of the LSA, a set of 33 small and medium-sized CVRP benchmark instances, which are taken from CVRP data sets (i.e. B, E, P, and V) for computational experiments. Each instance is tested using LSA for both integral and real (up to 4 precisions) distance matrix entries with respect to distinct values of n_0, m and CV. Note that for all the instances, the DA is assumed as 1 (i.e. $\delta = 1$). For each instance, the results namely, optimal integral solution, optimal fractional solution found by LSA, error percentage, CPU runtime, and the ratio of demand to capacity are summarized in Table 3. Furthermore, for each case, the optimal fractional solution corresponding route plan of each vehicle, delivered load and distance travelled by each vehicle can be found in Table 4. The observations are given as follows:

- Concerning Table 3, as most of the existing studies dealt with integral solutions, we present both integral and fractional optimal solutions in columns 7 and 8.
- Note that the results reported in column 8 represent the optimal solutions when the distance matrix entries are rounded to four precisions.
- To measure the superiority between the integral optimal solution (*Opt_Int.*) and fractional optimal solution (*Opt_Frac.*) found by LSA, error percentage (Error %) is computed using the formula (17) and those results are reported in column 9. It can be seen that the error percent values vary from 0 % to 8.37%, which are low.

$$\text{Error}\,\% = \frac{|\,Opt\_Frac - Opt\_Int.|}{Opt\_Frac.}\% \qquad (17)$$

- Infact, the error percent values are low, but it is interesting to note, among 26 problem instances, for 5 problem instances (i.e. 12, 18, 19, 20 and 21), the route plans for the integral and fractional optimal solutions are not same, those route plans are demonstrated in Table 5. In Table 5, it is seen that all the five instances have small deviation between the integral and fractional optimal solutions and it is varied from 1.5687 to 3.7455. It is evident that a small deviation can cause the changes in their optimal route plans. It is clearly observed that the fractional optimal solutions are more consistent than the integral optimal solutions.

- The CPU time required to achieve the optimal solutions are reported in column 10, it is ranging from 3 seconds to 892 seconds. The results reveal that apart from the structure of a given distance matrix, the truncation parameter (n_0) also affects the performance of the LSA. For instance, the LSA solves the test instance P-n21-k2 within 326 seconds, in which the two vehicles have to serve all the 21 delivery centers. It solves the same instance with same vehicle capacity restrictions within 542 seconds, where the two vehicles were asked to serve only 16 (out of 21) delivery centers. Similarly, under the same conditions, the LSA uses 498 seconds when the two vehicles were asked to serve only 18 (out of 21) delivery centers.
- A scatter plot overlying with a regression line is constructed between the size of instance and the CPU runtime, shown in Fig. 7 and it clearly indicates that there is a sensible positive linear correlation with +0.6009.
- The last column in Table 3 reports the values of the ratio of demand to capacity (RDC) determined using the formula (18). It is seen that most of the problem instances have more than 60 % vehicle capacity constraints.

$$\text{Ratio of demand to capacity} = \textit{delivered demand} \,/\, (m \times CV) \quad (18)$$

Here, delivered demand is the demand at which the delivery center i involved in the route plan, m is the number of vehicles, and CV is the vehicle capacity.

- To measure the effectiveness of RDC over the proposed LSA performance, ten distinct cases of the instance E-n23-k3 has been tested by giving distinct vehicle capacity restrictions and n_0 values. The experimental results can be found from problem instances 15 to 24 in Table 3.
- For the instances 15 to 17, the RDC values varies from 0.35 to 0.43, shows that the instances have loose vehicle capacity restrictions and their CPU runtime varies from 100 seconds to 192 seconds. For the instances 18 to 24, the RDC values vary from 0.76 to 0.95, shows that the instances have tight vehicle capacity restrictions as it and their CPU runtimes ranging from 192 seconds to 820 seconds.
- It is evident that when the problem instances have tight vehicle capacity restrictions, the LSA takes more CPU time to find the optimal solution.

Table 3
Summary computations of CTVRP over benchmark instances

SN	Instance	n	m	CV	n_0	*Opt_Int.*	*Opt_Frac.*	Error %	CPU time (s)	RDC
1	E-n13-k4	13	4	6000	9	177	177	0	3	0.49
2	E-n13-k4	13	4	6000	11	210	210	0	5	0.65
3	E-n13-k4	13	4	4000	11	281	281	0	25	0.91
4	E-n13-k4	13	4	6000	12	230	230	0	10	0.70
5	P-n19-k2	19	2	160	14	149	149.8755	0.58	95	0.71
6	P-n19-k2	19	2	160	16	169	170.3475	0.79	124	0.78
7	P-n19-k2	19	3	140	17	202	203.2165	0.59	262	0.68
8	P-n19-k2	19	2	150	15	160	161.5190	0.94	120	0.81
9	P-n20-k2	20	2	160	16	172	174.0778	1.20	228	0.71
10	P-n20-k2	20	3	140	16	193	194.0436	0.54	380	0.57
11	P-n20-k2	20	3	150	18	215	216.3241	0.61	282	0.63
12	P-n21-k2	21	2	160	16	163	164.5687	0.95	546	0.67
13	P-n21-k2	21	2	160	18	183	184.6479	0.89	498	0.77
14	P-n21-k2	21	3	140	16	181	182.4342	0.79	892	0.48
15	E-n23-k3	23	3	4500	18	353	352.3474	0.38	180	0.35
16	E-n23-k3	23	3	4500	20	402	401.5577	0.39	192	0.43
17	E-n23-k3	23	4	3500	20	445	444.3862	0.31	100	0.41
18	E-n23-k3	23	2	2300	14	277	277.1798	0.06	684	0.78
19	E-n23-k3	23	2	2300	15	298	298.2223	0.07	764	0.83
20	E-n23-k3	23	2	1300	12	252	252.2003	0.07	326	0.89
21	E-n23-k3	23	2	1300	11	232	231.8760	0.05	192	0.76
22	E-n23-k3	23	2	2500	16	317	318.5849	0.49	692	0.93
23	E-n23-k3	23	2	2500	17	335	335.9531	0.28	820	0.95
24	E-n23-k3	23	2	2800	19	380	378.4125	0.41	420	0.93
25	ulysses-n16-k3	16	3	5	12	41	44.7455	8.37	80	0.73
26	ulysses-n16-k3	16	3	5	14	52	54.4813	4.55	228	0.86
27	ulysses-n22-k4	22	4	6	16	41	43.8364	6.47	326	0.62
28	ulysses-n22-k4	22	3	6	14	30	32.0641	6.44	192	0.72
29	gr-n17-k3	17	3	6	14	1796	1796	0	234	0.72
30	gr-n17-k3	17	3	5	15	1962	1962	0	196	0.93
31	gr-n17-k3	17	3	6	16	2199	2199	0	312	0.83
32	gr-n21-k3	21	3	7	17	2511	2511	0	382	0.76
33	gr-n21-k3	21	3	7	19	3018	3018	0	440	0.85

n – number of city locations; m – number of vehicles; CV – capacity (number of cylinders) of a vehicle, n_0 – number of delivery centers to be served; *Opt_Int.* – optimal integral solution found by LSA; *Opt_Frac.* – optimal fractional solution found by LSA; Error % – error percentage of solutions caused due to rounding the distance matrix entries; CPU time (s) – CPU time (in seconds) required to provide the optimal solution; RDC– ratio of demand to capacity.

Table 4
The LPG distribution route plan corresponding to the optimal fractional solution of CTVRP

SN	Route of each Vehicle	Delivered Load	Distance Travelled
1	1 – 2 – 1	1200	18
	1 – 4 – 6 – 3 – 1	4900	54
	1 – 12 – 11 – 7 – 1	4700	69
	1 – 13 – 1	1100	36
2	1 – 2 – 1	1200	18
	1 – 9 – 6 – 3 – 1	5300	59
	1 – 10 – 13 – 4 – 1	4700	69
	1 – 12 – 11 – 7 – 1	4400	64
3	1 – 5 – 8 –7 – 1	4000	95
	1 – 11 – 10 – 1	3400	82
	1 – 12 – 3 – 1	3400	49
	1 – 13 – 4 – 2 – 1	3800	55
4	1 – 2 – 1	1200	18
	1 – 9 – 6 – 4 – 1	5100	75
	1 – 10 – 13 – 8 – 3 – 1	4700	69
	1 – 12 – 11 – 7 – 1	5800	68
5	1 – 5 – 12 – 15 – 13 – 11 – 2 – 1	82	76.6216
	1 – 19 – 6 – 14 – 16 – 8 – 3 – 7 – 1	147	73.2538
6	1 – 5 – 12 – 15 – 13 – 4 – 18 – 17 – 9 – 11 – 2 – 1	149	116.3288
	1 – 19 – 6 – 8 – 3 – 7 – 1	102	54.0188
7	1 – 3 – 8 – 10 – 16 – 14 – 6 – 19 – 1	126	98.9021
	1 – 5 – 12 – 15 – 13 – 4 – 9 – 11 – 2 –1	130	80.2312
	1 – 7 – 1	31	24.0832
8	1 – 19 – 6 – 14 – 16 – 8 – 3 – 7 – 1	98	88.2651
	1 – 5 – 12 – 15 – 13 – 4 – 11 – 2 –1	147	73.2538
9	1 – 7 – 8 – 14 – 10 – 17 – 15 – 6 – 20 – 1	98	88.2651
	1 – 5 – 12 – 16 – 13 – 4 – 11 – 2 – 1	131	85.8126
10	1 – 3 – 8 – 14 – 10 – 17 – 15 – 6 – 20 – 1	82	76.6216
	1 – 5 – 12 – 16 – 13 – 11 – 2 – 1	130	93.3388
	1 –7 – 1	31	24.0832
11	1 – 3 – 8 – 14 – 10 – 17 – 15 – 6 – 20 – 1	126	98.9021
	1 – 5 – 12 – 16 – 13 – 4 – 9 – 11 – 2 – 1	130	93.3388
	1 –7 – 1	31	24.0832
12	1 – 7 – 21 – 6 – 15 – 18 – 10 – 14 – 3 – 17 – 1	70	87.9470
	1 – 5 – 12 – 16 – 13 – 11 – 2 – 1	146	76.6216
13	1 – 7 – 8 – 3 – 14 – 10 – 18 – 15 – 6 – 21 – 1	149	94.1925
	1 – 5 – 12 – 16 – 13 – 4 – 11 – 2 –17 – 1	98	90.4555
14	1 – 5 – 12 – 16 – 13 – 11 – 2 – 1	119	85.8126
	1 – 21 – 6 – 15 – 18 – 10 – 14 – 8 – 7– 1	70	76.6216
	1 – 17 – 1	12	20
15	1– 8 – 9 – 5 – 6 – 10 – 14 – 12 – 7 – 2 – 3 – 4 – 17 – 16 – 18 – 15 – 1	4319	294.1598
	1 –13– 1	300	14.1421
	1 – 19 – 1	120	44.0454
16	1– 8 – 9 – 5 – 6 – 10 – 14 – 12 – 1	2875	142.3276
	1 –13– 1	300	14.1421
	1 – 19 – 20 – 21 – 18 – 15 – 16 – 17 – 4 – 3 – 2 – 7 – 1	2664	245.0879
17	1–7 – 2 – 3 – 4 – 17 – 16 – 15 – 18 – 21 – 20 – 19 – 1	2664	245.0879
	1 –12– 1	225	44.0454
	1 – 13 – 1	300	14.1421
	1 – 8 – 9 – 5 – 6 – 10 – 14 – 1	2650	141.1108
18	1 – 8 – 9 – 10 – 14 – 12 – 1	2075	107.8646
	1 – 15 – 16 – 17 – 4 – 3 – 2 – 7 – 13 – 1	1494	169.3152
19	1 – 8 – 9 – 10 – 14 – 12 – 1	2075	107.8646
	1 – 15 – 18 – 16 – 17 – 4 – 3 – 2 – 7 – 13 – 1	1744	190.3577
20	1 – 8 – 14 – 12 – 13 – 1	1125	83.3339
	1 – 15 – 16 – 17 – 4 – 3 – 2 – 7 – 1	1194	168.8664
21	1 – 14 – 12 – 13 – 1	775	63.0096
	1 – 15 – 16 – 17 – 4 – 3 – 2 – 7 – 1	1194	168.8664
22	1 – 7 – 2 – 3 – 4 – 17 – 16 – 15 – 21 – 20 – 19 – 1	2414	227.9074
	1 – 8 – 10 – 14 – 12 – 13 – 1	2225	90.6775
23	1 – 7 – 2 – 3 – 4 – 17 –16 – 15 – 21 – 20 – 19 – 1	2414	227.9074
	1 – 8 – 9 – 10 – 14 – 12 – 13 – 1	2375	108.0457
24	1 – 13 – 12 – 7 – 2 – 3 – 4 – 17 – 16 – 15 – 18 – 21 – 19 – 1	2589	237.3018
	1 – 14 – 10 – 6 – 5 – 9 – 8 – 1	2650	141.1107
25	1 – 3 – 2 – 4 – 8 – 1	4	14.8825
	1 – 13 – 12 – 16 – 1	3	11.3897
	1 – 15 – 6 – 7 – 14 – 1	4	18.4733
26	1 – 3 – 2 – 4 – 8 – 1	4	14.8825
	1 – 10 – 7 – 6 – 5 – 15 – 1	5	27.3145
	1 – 14 – 13 – 12 – 16 – 1	4	12.2843
27	1 – 8 –1	1	1.4406
	1 – 15 – 6 – 7 – 19 – 20 – 21 – 1	6	20.2185
	1 – 14 – 13 – 12 – 16 – 1	4	12.2843
	1 – 18 – 4 – 17 – 22 – 1	4	9.8931
28	1 – 8 –1	1	1.4406
	1 – 14 – 13 – 12 – 20 – 21 – 16 – 1	6	15.3824
	1 – 22 – 18 – 4 – 17 – 2 – 3 – 1	6	15.2411
29	1 – 7 – 17 – 13 – 1	3	234
	1 – 14 – 15 – 11 – 3 – 6 – 8 – 1	6	798
	1 – 16 – 12 – 9 – 4 – 1	4	764
30	1 – 3 – 11 – 5 – 15 – 14 – 1	5	895
	1 – 7 – 8 – 6 – 17 – 13 – 1	5	303
	1 – 16 – 12 – 9 – 4 – 1	4	764
31	1 – 3 – 11 – 5 – 10 – 15 – 14 – 1	6	1132
	1 – 7 – 8 – 6 – 17 – 13 – 1	5	303
	1 – 16 – 12 – 9 – 4 – 1	4	764
32	1 – 4 – 11 – 12 –1	3	312
	1 – 16 – 9 – 5 – 6 – 8 – 7 – 1	6	1030
	1 – 19 – 17 – 10 – 18 – 15 – 21 – 20 – 1	7	1169
33	1 – 7 – 8 – 6 – 12 –1	4	334
	1 – 16 – 5 – 9 – 3 – 2 – 15 – 21 – 1	7	1726
	1 – 19 – 17 – 10 – 18 – 20 – 11 – 4 – 1	7	958

Table 5
Instances with distinct route plans for the optimal integral and fractional solutions

Instance	m	CV	n_0	Opt_Int	Opt_Frac	Dev.		Route plans
P-n21-k2	2	160	16	163	164.5687	1.5687	Int_Rte	1 – 21 – 6 – 15 – 18 – 10 – 14 – 8 – 7 – 1
								1 – 5 – 12 – 16 – 13 – 11 – 2 – 17 – 1
							Frac_Rte	1 –7 –21– 6– 15– 18– 10– 14 – 3 – 17 – 1
								1 – 5 – 12 – 16 – 13 – 11 – 2 – 1
ulysses-n16-k3	3	5	12	41	44.7455	3.7455	Int_Rte	1– 4 – 2 – 3 – 1
								1 – 12 – 13 – 16 – 1
								1 – 8 –15 – 6 – 7 – 14 – 1
							Frac_Rte	1 – 3 – 2 – 4 – 8 – 1
								1 – 13 – 12 – 16 – 1
								1 – 15 – 6 – 7 – 14 – 1
ulysses-n16-k3	3	5	14	52	54.4813	2.4813	Int_Rte	1 – 3 – 2 – 4 – 8 – 1
								1 – 10 – 7 – 6 – 5 – 15 – 1
								1 – 14 – 12 – 13 – 16 – 1
							Frac_Rte	1 – 3 – 2 – 4 – 8 – 1
								1 – 10 – 7 – 6 – 5 – 15 – 1
								1 – 14 – 13 – 12 – 16 – 1
ulysses-n22-k4	4	6	16	41	43.8364	2.8364	Int_Rte	1 – 8 – 1
								1 – 19 – 10 – 20 – 21 – 16 – 1
								1 – 14 – 12 – 13 – 1
								1 – 3 – 2 – 17 – 18 – 4 – 22 –1
							Frac_Rte	1 – 8 – 1
								1 – 15 – 6 – 7 – 19 – 20 – 21 – 1
								1 – 14 – 13 – 12 – 16 – 1
								1 – 18 – 4 – 17 – 22 – 1
ulysses-n22-k4	3	6	14	30	32.0641	2.0641	Int_Rte	1 – 8 – 1
								1 – 3 – 2 – 17 – 18 – 4 – 22 – 1
								1 – 19 – 20 – 21 –12 – 13 – 16 – 1
							Frac_Rte	1 – 8 – 1
								1 – 22 – 18 – 4 – 17 – 2 – 3 – 1
								1 – 14 – 13 – 12 – 20 – 21 – 16 – 1

m – number of vehicles; CV – capacity of a vehicle; n_0 – number delivery centers required to serve; Opt_Int –optimal integral solution found by LSA; Opt_Frac – optimal fractional solution found by LSA; Dev. – difference between the optimal integral and optimal fractional solutions; Int_Rte – optimal integral solution corresponding route plan; Frac_Rte – optimal fractional solution corresponding route plan

5.3. Descriptive statistical analysis of LSA for CTVRP over random instances

In addition, to measure the performance of the proposed LSA, the computational experiments have been extended to the random instances. Initially, we define the size of node set (n), distance matrix (C), truncation parameter (n_0), number of vehicles (m), capacity of vehicles (CV), delivery demands (d_i)'s for different sizes of data sets as input parameters. The distance matrices have been generated randomly at distinct ranges using the command *randi* (r, n) in MATLAB, where r refers to the range of the matrix entries and n indicate the size of the random instance. All the test instances generated are symmetric in nature. The computational analyses carried out on a class of eight data sets varying from 10 to 50 generated at random. Overall, 80 random instances were tested. The input data sets for distance matrices of sizes 10, 15 and 20 have been generated over the range [1, 50], the sizes of 25, 30, and 35 were generated over the range [1, 100] and the sizes of 40 and 50 over the range [1, 200]. The delivery demands for each data set is considered unit demands. The descriptive statistical results of CPU runtime of LSA tested on random instances is summarized in Table 6. From Table 6, it is seen that the mean CPU runtime varies from 0.50 seconds to 904.80 seconds for solving the problem instances of sizes from 10 to 50, respectively. However, the CPU runtimes seem to be expensive, it is quite considerable from solution quality aspect. It is also seen that the standard deviation values vary from 0.20 to 64.75, which indicates that CPU runtimes are widely spread out to its mean CPU runtime. It is evident to claim that the proposed LSA relies on the structure of the data instance.

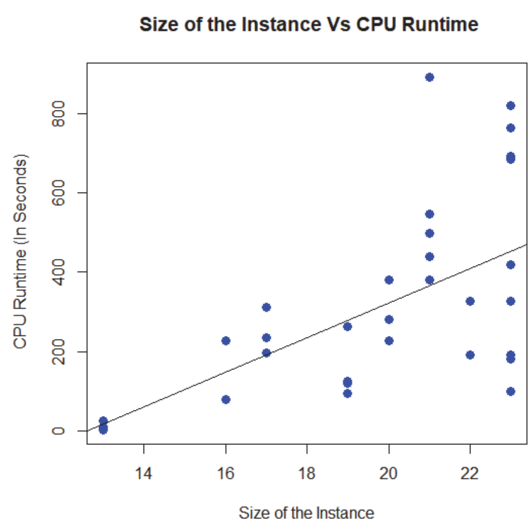

Fig. 7. Scatter plot with correlation +0.6009

Table 6
Descriptive statistical results of CPU runtime of LSA on random instances

SN	n	n_0	m	CV	*NPT*	LSA			
						Min.	Max.	Avg.	SD
1	10	8	2	4	10	0.26	0.96	0.50	0.20
2	15	10	2	5	10	2.54	8.04	5.65	2.12
3	20	14	3	5	10	48.02	77.52	58.81	9.74
4	25	18	3	6	10	302.54	410.43	363.74	34.84
5	30	24	4	7	10	380.40	504.76	434.52	47.03
6	35	28	5	6	10	410.43	594.26	502.36	64.75
7	40	30	5	7	10	610.39	794.98	695.22	58.87
8	50	40	6	7	10	846.01	994.26	904.80	49.96

SN– serial number; n – size of instance; n_0 – number of delivery centers required to serve including the DA; m – number of vehicles; CV – capacity of a vehicle; *NPT* – number of problems tried; Min.– minimum CPU runtime required to get the optimal solution; Max.– maximum CPU runtime required to get the optimal solution; Avg.– mean CPU runtime required to get the optimal solution; SD – standard deviation of CPU time for each of the data set

6. Conclusion

A typical model that arise in the context of distribution of LPG cylinders in rural areas referred to as *capacitated truncated vehicle routing problem* (CTVRP) is studied. The model has been formulated using zero-one integer linear programming problem. An exact Lexi-search algorithm (LSA) is developed to find the optimal solutions. A small sized benchmark instance has been illustrated through the proposed algorithm. The observations are given as follows:

- The comparative results and error analysis shown that the capability of proposed LSA in producing exact solutions.
- The extensive computational results reveal that LSA works efficiently for solving small and medium size test instances.
- The results reported in this study showed that fractional optimal solutions have more quality than the integral optimal solutions and these results will be used as a reference to the further investigations.
- Apart from the problem dimension and structure of the given distance matrix, the truncation parameter also judge the performance of the proposed LSA.
- The higher instances can be likely solved as well using the proposed LSA.
- By controlling the LSA search space, one can obtain the effective solutions in reasonable time, but that does not guarantees optimal solution.
- The model CTVRP finds wide applications in transportation, logistics distribution allied fields, etc.

References

Ahmed, N., Das, S., & Purusotham, S. (2012). The oil tankers dispatching problem. *Opsearch*, *49*(4), 366-385.

Ai, T. J., & Kachitvichyanukul, V. (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, *36*(5), 1693-1702.

Archetti, C., Doerner, K. F., & Tricoire, F. (2013). A heuristic algorithm for the free newspaper delivery problem. *European Journal of Operational Research*, *230*(2), 245-257.

Belik, I., & Jörnsten, K. (2016). A new Semi-Lagrangean Relaxation for the k-cardinality assignment problem. *Journal of Information and Optimization Sciences*, *37*(1), 75-100.

Bhavani, V., & Murthy, M. S. (2006). Truncated M-travelling salesmen problem. *Opsearch*, *43*(2), 152-177.

Bianchessi, N., & Righini, G. (2007). Heuristic algorithms for the vehicle routing problem with simultaneous pick-up and delivery. *Computers & Operations Research*, *34*(2), 578-594.

Chen, J. F., & Wu, T. H. (2006). Vehicle routing problem with simultaneous deliveries and pickups. *Journal of the Operational Research Society*, *57*(5), 579-587.

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

Dethloff, J. (2001). Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up. *OR-Spektrum*, *23*(1), 79-96.

Gensch, D. H. (1978). An industrial application of the traveling salesman's subtour problem. *AIIE Transactions*, *10*(4), 362-370.

Giardini, G., & Kalmar-Nagy, T. (2011). Genetic algorithm for combinatorial path planning: the subtour problem. *Mathematical Problems in Engineering*, *2011*.

İnanlı, A., Ünsal, B., & Türsel Eliiyi, D. (2015). Route optimization for the distribution network of a confectionary chain. *Promet-Traffic &Transportation*, *27*(6), 497-503.

Kalayci, C. B., & Kaya, C. (2016). An ant colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery. *Expert Systems with Applications*, *66*, 163-175.

Katagiri, H., Hayashida, T., Nishizaki, I., & Guo, Q. (2012). A hybrid algorithm based on tabu search and ant colony optimization for k-minimum spanning tree problems. *Expert Systems with Applications*, *39*(5), 5681-5686.

Keshtkaran, M., Ziarati, K., Bettinelli, A., & Vigo, D. (2016). Enhanced exact solution methods for the team orienteering problem. *International Journal of Production Research*, *54*(2), 591-601.

Kumar, T., & Purusotham, S. (2018). The degree constrained k-cardinality minimum spanning tree problem: a lexi-search algorithm. *Decision Science Letters*, *7*(3), 301-310.

Min, H. (1989). The multiple vehicle routing problem with simultaneous delivery and pick-up points. *Transportation Research Part A: General*, *23*(5), 377-386.

Montané, F. A. T., & Galvao, R. D. (2006). A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. *Computers & Operations Research*, *33*(3), 595-619.

Murthy, M. S. (1976). A bulk transportation problem. *Opsearch*, *13*(3-4), 143-155.

Polat, O., Kalayci, C. B., Kulak, O., & Günther, H. O. (2015). A perturbation based variable neighborhood search heuristic for solving the vehicle routing problem with simultaneous pickup and delivery with time limit. *European Journal of Operational Research*, *242*(2), 369-382.

Privé, J., Renaud, J., Boctor, F., & Laporte, G. (2006). Solving a vehicle-routing problem arising in soft-drink distribution. *Journal of the Operational Research Society*, *57*(9), 1045-1052.

Sharma, S. K., Routroy, S., & Yadav, U. (2018). Vehicle routing problem: recent literature review of its variants. *International Journal of Operational Research*, *33*(1), 1-31.

Stetsyuk, P. I. (2016). Problem statements for k-node shortest path and k-node shortest cycle in a complete graph. *Cybernetics and Systems Analysis*, *52*(1), 71-75.

Subramanian, A., Drummond, L. M. D. A., Bentes, C., Ochi, L. S., & Farias, R. (2010). A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, *37*(11), 1899-1911.

Subramanian, A., Uchoa, E., Pessoa, A. A., & Ochi, L. S. (2011). Branch-and-cut with lazy separation for the vehicle routing problem with simultaneous pickup and delivery. *Operations Research Letters*, *39*(5), 338-341.

Subramanian, A., Uchoa, E., Pessoa, A. A., & Ochi, L. S. (2013). Branch-cut-and-price for the vehicle routing problem with simultaneous pickup and delivery. *Optimization Letters*, *7*(7), 1569-1581.

Tarantilis, C. D., & Kiranoudis, C. T. (2001). A meta-heuristic algorithm for the efficient distribution of perishable foods. *Journal of food Engineering*, *50*(1), 1-9.

Tarantilis, C. D., & Kiranoudis, C. T. (2002). Distribution of fresh meat. *Journal of Food Engineering*, *51*(1), 85-91.

Tasan, A. S., & Gen, M. (2012). A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries. *Computers & Industrial Engineering*, *62*(3), 755-761.

Thenepalle, J. K., & Singamsetty, P. (2018). Bi-criteria travelling salesman subtour problem with time threshold. *The European Physical Journal Plus*, *133*(3), 1-15.

Volgenant, T., & Jonker, R. (1987). On some generalizations of the travelling-salesman problem. *Journal of the Operational Research Society*, *38*(11), 1073-1079.

Wang, C., Mu, D., Zhao, F., & Sutherland, J. W. (2015). A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup–delivery and time windows. *Computers & Industrial Engineering*, *83*, 111-122.

Wassan, N. A., Wassan, A. H., & Nagy, G. (2008). A reactive tabu search algorithm for the vehicle routing problem with simultaneous pickups and deliveries. *Journal of Combinatorial Optimization*, *15*(4), 368-386.

Xu, H., Li, Q., Wang, J., Luo, G., Zhu, C., & Sun, W. (2018). An optimization routing algorithm for green communication in underground mines. *Sensors*, *18*(6), 1950.

Zachariadis, E. E., Tarantilis, C. D., & Kiranoudis, C. T. (2009). A hybrid metaheuristic algorithm for the vehicle routing problem with simultaneous delivery and pick-up service. *Expert Systems with Applications*, *36*(2), 1070-1081.