THE HETEROGENEOUS VEHICLE ROUTING PROBLEM WITH SIMULTANEOUS PICKUP AND DELIVERY: A HYBRID HEURISTIC APPROACH BASED ON SIMULATED annealing

BarisKececi1*, FulyaAltiparmak2 and ImdatKara3
1Department of Industrial Engineering
Baskent University, Ankara, Turkey
bkececi@baskent.edu.tr

2Department of Industrial Engineering
Gazi University, Ankara, Turkey
fulyaal@gazi.edu.tr

3Department of Industrial Engineering
Baskent University, Ankara, Turkey
ikara@baskent.edu.tr

ABSTRACT

Vehicle Routing Problem (VRP) is one of the most important industrial engineering problems which may be encountered on transportation, distribution, logistics, scheduling, and etc. In real life logistics the requirements arising from customers and goods may force to use different vehicles. Besides, companies do care more about the management of reverse flow of goods because of their economic benefits and legal, environmental liabilities. In this paper, a variant of VRP, called Heterogeneous Vehicle Routing Problem with Simultaneous Pickup and Delivery (HVRPSPD), is considered. The HVRPSPD can be defined as determining the routes in such a way that pickup and delivery demands of each customer must be performed with same vehicle and the vehicle types on each route while minimizing the total cost. We propose a mathematical model for the problem. Since the HVRPSPD is an NP-hard problem, the proposed mathematical model can be used to find the optimal solution for the small-size problems. Hence we propose a hybrid heuristic based on the Simulated Annealing (SA) and Local Search (LS) to solve medium and large-size HVRPSPD. A series of experiments is performed to evaluate the performance of heuristic. Results show that the proposed heuristic is computationally efficient to find good quality solutions.

Keywords: Heterogeneous fleet, Simultaneous pickup and delivery, Vehicle routing problem, Mixed integer programming formulation, Heuristics, Simulated annealing algorithm

1 INTRODUCTION

The conceptual origin of Logistics goes back to the military operations in the ancient Roman civilization. Logistics can be defined broadly as the process of planning, implementing, and controlling the efficient, cost effective flow and storage of raw materials, in-process inventory, finished goods and related information from point of origin to point of consumption for the purpose of meeting customer requirements [1].

With the increasing population and scarce resources in the world, the importance of the logistics and distribution systems for the economy of countries, sectors and companies, can be seen in spectacular reports. Kearny [2] states that the annual distribution costs in United
States in 1983 were $650 million (approximately 0.02% of gross domestic product) and the transportation cost constitutes the 22.5% of the containable costs in a production activity. In another study Toth and Vigo [3] report that the use of computerized procedures for the distribution process planning produces substantial savings (generally from 5% to 20%) in the global transportation costs.

The Vehicle Routing Problem (VRP) which has a major importance in the field of transportation, distribution and logistics was first defined and modeled by Dantzig and Ramser [4] towards the end of 1950s. After the Dantzig and Ramser’s study, various models and algorithms have been proposed in the literature to obtain exact and approximate solutions for different types of VRP. The interest to the VRP is because it is one of the most important problems in the operational level logistics and also it is in the class of NP-hard problems. The VRP can be defined as the problem of determining the minimum cost routes, in which the vehicles in a fleet should follow, in order to satisfy customer requirements under some operational restrictions. There are various types of the VRP according to the considered operational constraints in the literature. One of the variants is the Heterogeneous Vehicle Routing Problem (HVRP) and the other is the VRP with Simultaneous Pickup and Delivery (VRPSPD).

In the classical VRP, vehicles in a fleet are considered identical, in other words homogeneous. However, in real life logistics, the vehicles in a fleet may have different properties such as fixed cost (purchasing or rental) of the vehicles, unit variable (transportation) cost between any two customers, vehicle capacities, etc. Besides, according to the customer and/or freight needs, different type of vehicles may be required. That is why to reduce the cost of logistics, the identification of distribution routes as well as the selection of vehicle fleet (i.e. how many vehicle should be bought/rent of each type and which type of vehicle should follow which route) are gaining importance for the companies. This situation particularly requires the consideration of the strategic investment decision in the absence of a current vehicle fleet. There are basically two types of HVRP examined in the literature. The first one in which has unlimited number of vehicles of each type was first proposed by Golden et al [5]. In this problem the optimal fleet of vehicles is determined. This problem was originally named in several ways in different studies such as, “The Fleet Size and Mix VRP by Golden et al [5], The Vehicle Fleet Mix by Salhi and Rand [6], The Fleet Size and Composition VRP by Gheysens et al [7]”. The second basic type of HVRP was first studied by Taillard [8] and there is a limited number of vehicles of each type in a fleet. This case is more realistic and was named in several different ways such as, “The VRP with a Heterogeneous Fleet of Vehicles by Taillard [8], The Heterogeneous Fixed Fleet VRP by Tarantilis et al [9]”. In addition, it is possible to classify the HVRPs depending on whether the fixed and transportation costs are considered or not and whether the fleet size is limited or not etc. Baldacci et al [10] give a classification for this problem. We refer the interested readers to the paper of Hoff et al. [11] for an extensive review about this problem and its variants.

Another basic assumption in the classical VRP is that the customers either demand or supply goods. Hence, the vehicles are considered either distribute or collect goods on a route. In the VRPSPD, each customer demands and supplies certain amount of goods at the same time. In case of the customers both demand and supply goods, major economic benefits can be obtained when both activities are performed by a vehicle on the same route rather than by vehicles on different routes. The applications of the VRPSPD can be encountered in the distribution system of grocery store chains, blood banks, etc. Reverse logistics is also another area in which the planning of vehicle routes takes the form of VRPSPD. Companies are facing more often with the management of reverse flow of the products, work in process and/or raw materials. Also there are increasing environmental and social responsibilities and some legal obligations that make the Reverse Logistics Management attractive and mandatory for the companies in addition to its economic return. Particularly on the environmental and economic issues such as collection, disposal and assessment of waste,
recycling, re-manufacturing and evaluation of used products; the applications of Reverse Logistics force the companies to use their distribution and logistics network in a most efficient way. We refer the interested readers to the papers of Berbeglia et al. [12] and Parragh et al. [13] for extensive review about this problem and its variants.

The need for new scientific challenges and an industrial demand for more powerful and versatile routing tools has shifted the focus of VRP research to more complex, general, and larger size variants [11]. Because of the increasing importance of the HVRP and VRPSPD in practical and in scientific researches, in this paper we consider a variant of the VRP called the HVRP with simultaneous pickup and delivery (HVRPSPD). The HVRPSPD includes more real-world aspects of routing problems than the classical VRP by taking into account the heterogeneous vehicle fleet and the simultaneous distribution and collection of goods. Despite the fact that the HVRP and VRPSPD are two important problems in the literature and practice, the HVRPSPD has received little attention from researchers so far. Rieck and Zimmermann [14] address a variant of the VRP faced by less-than-truckload carriers in Europe. The problem includes heterogeneous vehicles, time windows, simultaneous delivery and pick-up at customer locations, and multiple use of vehicles. They present a vehicle routing model which integrates the real-life VRP and the assignment problem of vehicles to loading bays at the depot. The proposed savings-based solution heuristic combines a multi-start and a local search procedure. Cetin and Gencer [15] consider the VRPSPD with time windows constraints and heterogeneous fleet. They propose a mixed integer programming formulation for the problem based on the model developed by Dethloff [16] for the VRPSPD.

In this paper, we propose a mixed integer programming (MIP) formulation, which is arc-based formulation, for the HVRPSPD. Since the problem is in the class of NP-hard problems, the proposed formulation can be used to obtain optimal solutions for small-sized problems. Hence, we propose a hybrid heuristic algorithm based on simulated annealing (SA) and local search (LS) (called SA_LS) to solve the medium and large-size HVRPSPDs [17, 18]. We investigate the performance of the SA_LS on a set of instances derived from the literature and compare it with the proposed arc-based MIP formulation in terms of the solution quality and computation time. The paper is organized as follows: Problem definition is given in Section 2. The detailed description of the proposed algorithm is given in Section 3. Section 4 reports computational results and conclusions follow in Section 5.

2 PROBLEM DEFINITION

The HVRPSPD can be defined mathematically as in the following. Let \( G = (N, A) \) be a complete directed graph where \( N = \{0, \ldots, n\} \) is the set of nodes and \( A = \{(i, j) : i, j \in N, i \neq j\} \) is the set of arcs, respectively. \( 0 \) indicates the depot node while the remaining are the customer nodes in \( N \). The fleet is composed by \( b \) different types of vehicles, with \( B = \{1, \ldots, b\} \). For each \( k \in B \) there are \( T_k \) available vehicles, each with capacity \( Q_k \) and fixed cost \( f_k \). Each arc \( (i, j) \in A \) is associated a nonnegative cost \( c_{ij} = \theta_k l_{ij} \) where \( l_{ij} \) is the distance between the nodes \( (i, j) \) with \( l_{ij} = l_{ji} \) for each \( i, j \in N \) triangular inequality holds (i.e. \( l_{ij} + l_{jk} \geq l_{ik} \)) and \( \theta_k \) is the dependent (variable) cost per distance unit of vehicle \( k \in B \). Each customer \( i \in N \) has delivery \( (d_i) \) and pickup \( (p_i) \) demands, with \( 0 \leq d_i, p_i \leq Q_k, \forall k \in B \) and \( d_0 = p_0 = 0 \). The problem consists in finding the minimum cost feasible routes and determining the type of vehicle on each route such that only one type of vehicle must be used on each route and each customer must be visited by exactly one type of vehicle, each route must begin and end at the depot and the total load on vehicles must not exceed the vehicle capacity.

Based on the above definitions, the decision variables of the proposed MIP formulation are given as follows: \( x_{ijk} = 1 \) if a vehicle of type \( k \) travels directly from node \( i \) to node \( j \); \( z_{ij} = \) the total remaining delivery load of vehicle just after it gives the delivery demand of node \( i \), if the vehicle travels directly on arc \( (i, j) \), otherwise 0; \( t_{ij} = \) the total load picked up by vehicle, if the vehicle travels directly on arc \( (i, j) \), after it takes the pickup demand of node
\[ \min z = \sum_{i\in N} \sum_{j\in N} \sum_{k\in B} c_{ijk} x_{ijk} + \sum_{k\in B} f_k y_k \]  

Subject to

1. \[ \sum_{j\in N\setminus\{0\}} \sum_{k\in B} x_{0jk} \leq m \]
2. \[ \sum_{i\in N\setminus\{0\}} \sum_{k\in B} x_{i0k} \leq m \]
3. \[ \sum_{i\in N} \sum_{k\in B} x_{ijk} = 1, \ j \in N\{0\} \]
4. \[ \sum_{i\in N} x_{ijk} = \sum_{i\in N} x_{jik}, \ i \in N\{0\}, k \in B \]
5. \[ z_{ij} + t_{ij} \leq \sum_{k\in B} Q_k x_{ijk}, \ i \in N, j \in N, i \neq j \]
6. \[ \sum_{i\in N} z_{ji} - \sum_{j\in N} z_{ij} = d_i, \ i \in N \]
7. \[ \sum_{k\in B} d_i x_{ijk} \leq z_{ij} \leq \sum_{k\in B} (Q_k - d_i) x_{ijk}, \ i \in N, j \in N, i \neq j \]
8. \[ \sum_{i\in N} t_{ij} - \sum_{j\in N} t_{ji} = p_i, \ i \in N\{0\} \]
9. \[ \sum_{k\in B} p_i x_{ijk} \leq t_{ij} \leq \sum_{k\in B} (Q_k - p_j) x_{ijk}, \ i \in N, j \in N, i \neq j \]
10. \[ z_{i0} = 0, \ i \in N\{0\} \]
11. \[ t_{0j} = 0, \ j \in N\{0\} \]
12. \[ \sum_{k\in B} y_k \leq m \]
13. \[ y_k \leq T_k, \ k \in B \]
14. \[ \sum_{j\in N\{0\}} x_{0jk} = y_k, \ k \in B \]
15. \[ y_k \geq 0 \text{ and integer, } k \in B \]
16. \[ m \geq 0 \]
17. \[ z_{ij}, t_{ij} \geq 0, \ i \in N, j \in N \]
18. \[ x_{ijk} \in \{0,1\}, \ i \in N, j \in N, k \in B \]

In this formulation, the objective function (1) minimizes the total transportation cost and the total vehicle utilization cost. The constraints (2) and (3) satisfy at most \( m \) vehicles leave and return back to the depot, respectively. The constraint (4) yields that any node is visited by exactly one type of vehicle and with the constraint (5) it is guaranteed that the same type of vehicle enters and leaves at any node. The constraint (6) prevents the vehicle capacity to be exceeded on any node in a feasible solution. Furthermore, it enforces the auxiliary variables to be zero in case they are not in the solution. The constraint (7) ensures that the auxiliary variables, related with the delivery load, take decreasing values on a feasible vehicle tour and similarly the constraint (9) ensures that the auxiliary variables, related with the pick-up load, take increasing values on a feasible vehicle tour. (7) and (9) avoid the sub-tours together. The constraints (8) and (10) are bounding constraints of delivery and pickup loads, respectively. They strengthen the model and give tighter formulation. The equalities (11) and (12) initially give zero value to the related variables since a vehicle starts and ends its tour with empty load, respectively. The constraint (13) provides that the total number of vehicles of each type must be equal to at most \( m \) in the fleet. The constraint (14) restricts that the number of vehicle of each type in the fleet must
be less than or equal to the available number of vehicle of each type. Finally, the constraint (15) ensures that the number of arcs leaving the depot of vehicle type \( k \), should be equal to the number of vehicles of type \( k \) in the fleet. The constraints (16), (17), (18) and (19) are the non-negativity and integrality constraints. The arc-based formulation has \( O(n^2b) \) number of 0-1 integer decision variables, \( O(2n^2) \) number of continues decision variables, \( O(b) \) number of integer decision variables and \( O(3n^2) \) number of constraints.

3 PROPOSED HYBRID SIMULATED ANNEALING ALGORITHM

Since the HVRPSPD is an NP-hard problem, the proposed formulation is not directly applicable to obtain optimal solutions for medium- and large-size problems. Hence, heuristics are necessary to quickly obtain solutions for these problems. In this paper, we propose a hybrid heuristic approach based on Simulated Annealing (SA) and Local Search (LS), called SA_LS, to solve the medium- and large-size HVRPSPDs. In SA_LS, while SA is implemented to search good vehicle routes in the solution space, LS is used to improve the best vehicle routes found during the search process of SA.

The proposed SA_LS algorithm starts with an initial solution \( S \). A new solution \( S_{new} \) is generated in the neighborhood of the current solution \( S \) by using a moving strategy which is randomly selected within a group of strategies (called Type-1 moving strategies). If the new solution \( S_{new} \) is better than the current solution \( S \), i.e. \( f(S_{new}) < f(S) \), then the search process continues with a simple LS by using a moving strategy, which is each time randomly selected within a group of strategies (called Type-2 moving strategies), and updates the new solution \( S_{new} \). LS continues until no improvement is obtained for five successive iterations. At the end of the LS, four moving strategies (called Type-3 moving strategies) are implemented in a predetermined order to the \( S_{new} \) and \( S_{new} \) is updated. If the new solution is better than the current solution then it is accepted as the current solution, otherwise it is accepted with the probability of \( \exp(-\Delta s/T_{it}) \) as the current solution where \( T_{it} \) is the current temperature. \( \Delta s \) is a relative percent deviation of quality of the new solution from the current solution and calculated by \( [f(S_{new}) - f(S_{cur})]/f(S_{cur})]*100 \). In each iteration of the SA_LS, the temperature is reduced according to the geometric cooling schedule, \( T_{it} = 0.95T_{it-1} \). The SA_LS stops whenever the temperature reaches a final temperature \( (T_f) \) or the best solution is not improved for \( n \) successive iterations.

This section describes the basic structure of the SA_LS considering solution representation, generating the initial solution and moving strategies.

3.1 Representation

Representation is one of the important issues that affect the performance of meta-heuristics. In general, different problems have different data structures or representations. In SA_LS, we utilize a matrix to represent solutions for the HVRPSPD. In this matrix, each row corresponds to a tour of vehicle where the first element in the row shows the vehicle type and the remaining indicate the customers to be visited sequentially in the tour. An illustrative example for a representation is given in Figure 1. In this example, an HVRPSPD instance has 3 different vehicles and 10 customers. As seen from Figure 1, the solution consists of three tours. While the vehicle type 3 visits customers 7-1-8 sequentially in the first tour, the vehicle types 1 and 3 visit customers 2-9 and 4-5-3-10-6 in the second and third tour, respectively.
3.2 Generating the Initial Solution

An initial solution is used to start the search mechanism. It can be generated randomly or using heuristic algorithms developed for the problem. Since heuristic solutions are closer to the optimal one than the random solution, it is generally preferred to obtain a heuristic solution to save computation time of the search algorithm. In this study, we propose a heuristic approach to obtain an initial solution for the HVRPSPD. This approach based on the giant tour and its partition is a kind of “route first-cluster second” approach which is well known in the VRP literature. The similar approach was first proposed by Beasley [26] for the VRP. Golden et al [27] used this method to solve the HVRP. Also, it has been used in several papers [28-30] for solving different type of VRP variants.

The proposed approach to generate the initial solution for the HVRPSPD constitutes of three main steps: generating giant tour, its partition and feasibility check. These steps are given as follows:

In the first step, a giant tour is generated by solving TSP in which the pickup and delivery demands of customers and capacity restrictions of vehicles in the problem are omitted. We used Concorde TSP Solver\(^\dagger\) to optimally solve the TSP, i.e. to obtain an ordered set of customers, \(E = \{E_1, E_2, ..., E_{|N| - 1}\}\) over the given network.

In the second step, the giant tour along \(E\) is partitioned into routes. An acyclic auxiliary graph \(H = (\hat{E}, F)\) is built to partition the giant tour. In this graph, \(F = \{(i, j, k): i < j, i, j \in N, k \in B\}\) is the set of arcs and \(\hat{E} = \{0, 1, 2, ..., |N| - 1\}\) is the set of nodes which contains a dummy node 0 and the nodes 1 to \(|N| - 1\) for the customers \(E_1\) to \(E_{|N| - 1}\). The arc \((i, j, k)\) in the graph represents the vehicle of type \(k\) leaves the depot, visits the customers \(E_{i+1}\) to \(E_j\) and returns back to the depot. The weight \(\hat{c}_{ijk}\) of the arc \((i, j, k)\) is calculated as follows:

\[
\hat{c}_{ijk} = \begin{cases} 
f_k + \theta_k(l_{0,E_{i+1}} + \sum_{r=i+1}^{j-1} l_{E_r,E_{r+1}} + l_{E_j,0}), & i < j, \max(\sum_{r=i+1}^{j-1} d_{E_r}, \sum_{r=i+1}^{j-1} p_{E_r}) \leq Q_k \\
M, & \text{otherwise}
\end{cases}
\]

An optimal partitioning of \(E\) is obtained by a minimum-cost path from node 0 to node \(|N| - 1\) in \(H\) where at most \(T_k\) number of arc \((i, j, k)\) must be selected since there is availability restriction for vehicle of type \(k\). This problem is called Shortest Path Problem with Resource Constraints [31] in the literature. Although this problem is an NP-hard problem, it can be solved quickly in practice using dynamic programming methods [29]. Thus we use the following MIP formulation to optimally partition \(E\) into routes. In this formulation the binary decision variable \(\hat{x}_{ijk} = 1\) iff the arc \((i, j, k)\) is in the solution, otherwise 0.

\[^\dagger\] Concorde TSP Solver can be reached in the source: \url{http://www.math.uwaterloo.ca/tsp/concorde/index.html}.
\[
\min \hat{z} = \sum_{i \in N} \sum_{j \in N} \sum_{k \in B} \hat{x}_{ijk} \hat{x}_{ijk} 
\]  
\text{(21)}

Subject to
\[
\sum_{j \in N \setminus \{0\}} \sum_{k \in B} \hat{x}_{0jk} = 1 
\]  
\text{(22)}

\[
\sum_{i \in N \setminus \{|N| - 1\}} \sum_{k \in B} \hat{x}_{ijk} = 1 
\]  
\text{(23)}

\[
\sum_{i \in N} \sum_{k \in B} \hat{x}_{ijk} = \sum_{i \in N} \sum_{k \in B} \hat{x}_{ijk}, \ j \in N \setminus \{0, |N| - 1\} 
\]  
\text{(24)}

\[
\sum_{i \in N} \sum_{j \in N} \hat{x}_{ijk} \leq T_k, \ k \in B 
\]  
\text{(25)}

\[
\hat{x}_{ijk} \in \{0, 1\}, \ i \in N, j \in N, k \in B 
\]  
\text{(26)}

In this formulation, the objective function (21) minimizes the total arc weights. The constraints (22) and (23) satisfy that exactly one arc leaves the origin and enters the destination, respectively. The constraint (24) yields exactly one arc enters and leaves the intermediate nodes. The constraint (25) guarantees that at most \(T_k\) number of arc \((i, j, k)\) is selected in the optimal solution. The constraint (26) defines the integrality conditions of the decision variables.

In the third step, after partitioning the giant tour into the routes, every route is checked whether the maximum load along the route exceeds the vehicle capacity. If a violation is observed in any route in terms of the vehicle capacity, the route is repaired by considering it as a VRPSPD with one vehicle. As known, the VRPSPD with one vehicle is a special case of the HVRPSPD. Thus, this problem is easily solved by the MIP formulation of the HVRPSPD considering following settings \(N = N', B = \{k^*\}, m = 1\) where \(N'\) is the set of nodes which includes the depot and the nodes in the route to be repaired and \(k^*\) is the vehicle type assigned to the route.

### 3.3 Moving Strategies

The proposed SA_LS algorithm uses seven inter-route and four intra-route moving strategies to search the neighbors of a current solution. These strategies are explained in this section in detail. Inter-route moving strategies are;

**Shift(1,0):** A customer \(i\) from route \(r_1\) is transferred to route \(r_2\).

**Shift(2,0):** Two adjacent customers \(i\) and \(j\) from route \(r_1\) are transferred to route \(r_2\).

**Swap(1,1):** Customer \(i\) from a route \(r_1\) and a customer \(j\) from a route \(r_2\) are exchanged.

**Swap(2,1):** Two adjacent customers, \(i\) and \(j\), from a route \(r_1\) are exchanged by a customer \(k\) from a route \(r_2\).

**Swap(2,2):** Two adjacent customers, \(i\) and \(j\), from a route \(r_1\) are exchanged by another two adjacent customers \(k\) and \(l\), belonging to a route \(r_2\).

**Cross:** The arcs between customers \(i\) and \(j\) in route \(r_1\) and between \(k\) and \(l\) in route \(r_2\) are removed. Then two arcs connecting customers \(i\) and \(l\) and also customers \(k\) and \(j\) are inserted.

**K – Shift:** A subset of consecutive customers from a route \(r_1\) is transferred to the end of a route \(r_2\).

Intra-route strategies are;

**Or – opt:** One, two or three adjacent customers from a route are removed and inserted in another position of the same route.

**2 – opt:** Two nonadjacent arcs are deleted and another two are added in such a way that a new route is generated.
Exchange: The positions of two customers $i$ and $j$ in a route are changed.

Reverse: The route direction is reversed if there is a chance to reduce the maximum load along the route.

The SA_LS implements the best improvement strategy when searching a solution space by a moving strategy to select a neighbor of the current solution. Moreover, it accepts only the feasible moves which do not violate the maximum load constraints. We conducted several tests on the inter-route moving strategies to determine which of them are more effective on the performance of the LS. This analysis showed that the inter-route moving strategies $\text{Shift}(1,0)$, $\text{Shift}(2,0)$ and $K-\text{Shift}$ (classified as Type-2) are more effective than $\text{Swap}(1,1)$, $\text{Swap}(2,1)$, $\text{Swap}(2,2)$ and $\text{Cross}$ (classified as Type-1) on the performance of the LS. Thus, it is decided that the Type-1 strategies are implemented to generate the best neighbor of the current solution in the SA_LS while the Type-2 strategies are used to improve solution by the LS procedure of the SA_LS. Moreover, based on our another preliminary analysis, the intra-route moving strategies (classified Type-3) are used in order of $\text{Exchange}$, $2-\text{opt}$, $\text{Reverse}$ and $Or-\text{opt}$ at the end of each LS implementation of the SA_LS.

### 4 COMPUTATIONAL RESULTS

In order to investigate the performance of SA_LS, the results of the SA_LS are compared with the upper bounds obtained by the MIP formulation on a set of test instances. Two problem sets for the HVRP are used to generate the HVRPSPD test instances. The first HVRP test set was derived by Taillard’s [8] from the VRP test problems of Golden et al [27]. This set includes four instances with 50 customers, two instances with 75 customers and two instances with 100 customers. The second HVRP test set was derived by Liu and Shen [32] from the Solomon’s [33] VRP test problems. This set consists of eighty instances with 100 customers. A HVRPSPD test instances can be easily obtained from a HVRP instance by using a demand separation approach.

In this study, we utilize two demand separation approaches to generate the delivery and pickup demands of customers in each HVRPSPD test instance. The first strategy was proposed by Salhi and Nagy [34]. In the first approach a ratio $r_i = \min\{x_i/y_i; y_i/x_i\}$ is calculated and the original demands were split into the pickup and delivery demands according to this ratio. For example, let $q_i$ be the original demand of the customer $i$. Then the delivery demand is $d_i = r_i q_i$ and the pickup demand $p_i = (1 - r_i) q_i$. We call this type of problems as Type X. Similarly, another problem type which is briefly referred to as Type Y is obtained by shifting each demand of the customer to the next one’s demand. In this study, we also proposed another demand separation approach. This approach splits the original demands into the pickup and delivery demands according to the “Golden Ratio”. In mathematics two quantities are in the Golden Ratio if the ratio of the sum of the quantities to the larger quantity is equal to the ratio of the larger quantity to the smaller one. For instance when we divide a line segment $|AB|$ into two parts according to the Golden Ratio, this segment should be divided by a point of $C$ such that the ratio of the bigger part $|CB|$ to the smaller part $|AC|$ is equal to the ratio of the whole line segment $|AB|$ to the bigger part $|CB|$; i.e. $\frac{|CB|}{|AC|} = \frac{|AB|}{|CB|} = \phi$. $\phi$ is an irrational number like $\pi$ or $e$ and it is equal to $\frac{1+\sqrt{5}}{2}$ in fractional and is equal to 1.6180339887... in decimal as well. According to these definitions, the original demand $q_i$ of the customer $i$ is split in to the pickup and delivery demands as $d_i = \left\lceil \frac{2q_i}{1+\sqrt{5}} \right\rceil$, $p_i = q_i - d_i$ in case $i$ is odd and as $p_i = \left\lfloor \frac{2q_i}{1+\sqrt{5}} \right\rfloor$, $d_i = q_i - p_i$ in case $i$ is even. We call this type of problems as Type W and similarly Type Z of problems are generated with shifting each demand of the customer to the next customer as it is explained previously. As a result, 520 (5x4x26) small and medium-sized instances are generated by using the first 20, 25, 30, 35 and 40 customers of the problems in each problem set which are generated by using 26 original HVRP test problems and 4 different separation procedures (X, Y, W, and Z).
The OPL language and CPLEX® 12.3 solver engine are used in coding and solving the MIP formulations of HVRPSPD, respectively. The proposed SA_LS algorithm interacting with the CPLEX Concert Technology is coded in C++ programming language in Visual Studio 2008 compiler and all experiments are performed on a computer with Intel Core i5 750 CPU @2,67 @2,67 Ghz processor and 2 GB RAM. Based on our preliminary experiments, the initial temperature in the SA_LS algorithm is taken as 380 in which an inferior solution (inferior by 40% relative to the current solution) is accepted with a probability of 0.90, the final temperature is taken as 0.15 such that a solution which is inferior by 1% relative to current solution is accepted with a probability of 0.001 and the LS procedure within the SA_LS algorithm is used with the probability of 0.2. Each instance is run 5 times by the proposed algorithm with different random seeds and the computation time of MIP formulation is limited with 2 CPU hours.

In the comparison of the results of the SA_LS and the MIP formulation, following performance measures are used: percentage gap and computation time. The percentage gap is calculated as $100 - (\frac{(UB - LB)}{LB})$ where LB is the LP relaxation bound of the MIP formulation obtained within two hours and upper bound (UB) is the optimal/best solution obtained by the MIP formulation within two hours / the proposed SA_LS. Also, we utilize additional two indicators to represent the performance of the SA_LS. These are average improvement of the initial solution and the number of problems solved optimally. Table 1 presents computational results for the SA_LS and the MIP formulation on HVRPSPD instances obtained by using two demand separation approaches. In the table, the first two columns show the number of customers in a HVRPSPD instance and the demand separation strategy, respectively. Subsequent four columns show the average value of percentage gap, the average improvement of the initial solution, the average computation time and the number of problems solved optimally with the SA_LS, respectively. Finally, last two columns stand for average percentage gap of upper bound obtained by MIP formulation and average computation time of the MIP formulation. As seen from table, good quality solutions are obtained by the proposed algorithm in a very short computation time. The average computation time of the SA_LS is 5.87 seconds while the MIP formulation needs 2493 seconds on average to solve HVRPSPD instances. While the average percentage gap of the SA_LS is 8.19%, this value reduces to 7.24% for MIP formulation. However, the performance of the MIP formulation quickly degenerates for the instances with 35 and 40 customers. The SA_LS algorithm optimally solves the 45 of 198 instances which are optimally solved by MIP formulation. Also, it improves upper bounds obtained by MIP formulation for 216 instances. These results show that the SA_LS is superior to MIP formulation. Moreover, the SA_LS can improve the initial solution 5.20% in average and also this improvement is between 2.80% and 11.70%.

5 CONCLUSION

In this study, we consider the heterogeneous vehicle routing problem with simultaneous pickup and delivery, HVRPSPD, and propose a MIP formulation and a hybrid heuristic approach to solve the problem. The proposed heuristic approach is based on SA and LS, called SA_LS. We analyze the performance of the proposed algorithm on a set of instances adopted from the literature. Computational results indicate that the good quality solutions (8.19% in average) are obtained in a reasonable computation time (approximately 6 sec.). Further research can be performed on the investigation of the proposed algorithm on the large size test instances. Moreover, the proposed algorithm can be used to obtain an initial solution for any exact algorithm (i.e. branch and bound, branch and cut, column generation) to shorten the optimization process of exact algorithm.
Table 1: Computational Results of the Proposed SA_LS Algorithm

<table>
<thead>
<tr>
<th>n</th>
<th>Type</th>
<th>SA_LS</th>
<th>MIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>X</td>
<td>4.50</td>
<td>5.50</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>4.40</td>
<td>5.30</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>4.00</td>
<td>7.10</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>6.30</td>
<td>3.30</td>
</tr>
<tr>
<td></td>
<td>Avg. (Sum)</td>
<td>4.80</td>
<td>5.30</td>
</tr>
<tr>
<td>25</td>
<td>X</td>
<td>5.10</td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>5.00</td>
<td>3.50</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>6.10</td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>8.10</td>
<td>3.20</td>
</tr>
<tr>
<td></td>
<td>Avg. (Sum)</td>
<td>6.08</td>
<td>3.58</td>
</tr>
<tr>
<td>30</td>
<td>X</td>
<td>8.75</td>
<td>4.30</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>11.20</td>
<td>2.80</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>6.55</td>
<td>6.30</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>6.18</td>
<td>4.90</td>
</tr>
<tr>
<td></td>
<td>Avg. (Sum)</td>
<td>8.17</td>
<td>4.58</td>
</tr>
<tr>
<td>35</td>
<td>X</td>
<td>8.32</td>
<td>11.70</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>9.29</td>
<td>5.72</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>7.97</td>
<td>3.99</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>9.88</td>
<td>3.67</td>
</tr>
<tr>
<td></td>
<td>Avg. (Sum)</td>
<td>8.87</td>
<td>6.27</td>
</tr>
<tr>
<td>40</td>
<td>X</td>
<td>16.60</td>
<td>5.44</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>18.10</td>
<td>8.10</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>9.08</td>
<td>4.93</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>8.36</td>
<td>6.66</td>
</tr>
<tr>
<td></td>
<td>Avg. (Sum)</td>
<td>13.04</td>
<td>6.28</td>
</tr>
<tr>
<td></td>
<td>Avg. (Sum)</td>
<td>8.19</td>
<td>5.20</td>
</tr>
</tbody>
</table>

6 REFERENCES


