A Modified Savings Algorithm Based Approach for Vehicle Routing Problem with Simultaneous Pick-up and Delivery

Kiran M\textsuperscript{1}, Cijo Mathew\textsuperscript{2}, Jacob Kuriakose\textsuperscript{3}

\textsuperscript{1}P. G. Scholar, \textsuperscript{2,3}Asst. Professor, M. A. College of Engineering, Kothamangalam, Ernakulam

Abstract—A key element of many distribution systems is the routing and scheduling of vehicles through a set of nodes requiring service. The Vehicle Routing Problem (VRP) can be defined as a problem of finding the optimal routes of delivery or collection from one or several depots to a number of cities or customers, while satisfying some constraints. In many practical situations, however, the vehicle is often required to simultaneously drop off and pick up goods at the same stop. Vehicle routing problem with simultaneous pick-up and delivery (VRPSPD) is a variation of VRP in which customers require both pick-up and delivery simultaneously. The importance of reverse logistics increased the importance of VRPSPD. The present study focuses on VRPSPD. Savings Algorithm is a heuristic algorithm developed by Clarke and Wright to solve VRP. A new algorithm is developed by modifying savings algorithm to solve the problem. The developed algorithm is tested using problem instances from VRP library.

Keywords—Vehicle routing problem, Reverse logistics, Vehicle routing problem with simultaneous pick – up and delivery, Savings Algorithm.

I. INTRODUCTION

The Vehicle Routing Problem (VRP) is a well-known problem in operational research where customers of known demands are supplied by one or several depots. The Vehicle Routing Problem can be defined as a problem of finding the optimal routes of delivery or collection from one or several depots to a number of cities or customers, while satisfying some constraints. The VRP has drawn enormous interests from many researchers during the last decades because of its vital role in planning of distribution systems and logistics in many sectors such as garbage collection, mail delivery, snow ploughing and task sequencing. The VRP is divided into many types. The important problems are VRP with Time Windows, VRP with Pick-Up and Delivery and Capacitated VRP. Huge research efforts have been devoted to studying the VRP since 1959 where Dantzig and Ramser have described the problem as a generalised problem of Travelling Salesman Problem (TPS) [9].

Thousands of papers have been written on several VRP variants such as Vehicle Routing Problem with Time Windows (VRPTW) [15], Vehicle Routing Problem with Pick-Up and Delivery (VRPPD) [10] and Capacitated Vehicle Routing Problem (CVRP) [12].

A growing environmental awareness has lead to increasing efforts to protect the environment In order to achieve a considerable waste reduction, used economic goods at the end of their lifecycle can be partly or fully recycled or disassembled, remanufactured, and partly or fully re-used. In addition, packaging and loading devices can also be either recycled or re-used. All of these activities result in a flow of materials from the end user back ‘up the supply chain’ or known as reverse logistics [1]. One of the most critical issues that affect the performance of the reverse logistics is the routing of vehicles. However, efficient and effective vehicle routing requires high computational efforts. Increasing importance of reverse logistics activities and the computational complexity of VRPSPD motivates this study.

The VRPSPD is a variation of the classic VRP, in which clients have both delivery and pickup demands and each customer, should be serviced by a single vehicle. Pickup and delivery should be performed simultaneously such that each customer is visited only once by a vehicle.

- Each customer should be serviced by a single vehicle.
- If a customer has both pickup and delivery demands, the delivery task is serviced before the pickup task.
- All vehicles have the same capacity
- At any point of a route, the loaded amount, that is, the sum of the up-to-that-point pickup and future-after-that-point delivery amounts cannot exceed the vehicle capacity.
- All vehicles start from and end at the depot.

The VRPSPD was first introduced by Min (1989), who studied the book delivery and pick-up activity between a central library and 22 local libraries with a given number of capacitated vehicles.
The VRP is known to be NP-hard (Golden et al., 1981) [3]. When all the pick-up demands of customers are set to 0, VRPSPD becomes VRP. The VRPSPD is NP-hard because its special case is NP-hard. Thus, we propose a heuristic algorithm that is capable of generating high quality solutions within relatively short computational times.

The purpose of this study is to introduce an alternative approach to solve VRPSPD. More specifically, this study proposes a modified savings algorithm (mSA) based approach, for solving VRPSPD.

The remainder of the paper is organized as follows; in Section 2, mathematical formulation and details of VRPSPD are presented. In Section 3, the detail of the proposed approach is introduced. The performance of the proposed approach is evaluated with several test problems and the results are discussed in Section 5. Finally, Section 6 concludes the paper.

II. VEHICLE ROUTING PROBLEM WITH SIMULTANEOUS PICK–UP AND DELIVERY

In VRPSPD customers require simultaneous pick-up of goods from their location in addition to delivery of goods to their location as illustrated in Figure 1. In Figure 1, red dots represent customer nodes while blue rectangle represents the depot; red arrows represent deliveries; green arrows represent pick-ups in customer nodes; solid lines are the arcs between nodes.

A) Literature Survey

VRPSPD was introduced into the literature by Min (1989) [14]. His study considered distribution problem of a public library. Vehicle routes were determined by a solution approach that based on clustering customers according to their demands and vehicle capacities first, and then solving TSP for each cluster.

Dethloff (2001) [6] emphasized the importance of VRPSPD for reverse logistics activities and he investigated the relation between VRPSPD and other types of routing problems. He proposed a mathematical formulation for this problem and developed an insertion based heuristic, which uses several insertion criteria. Nagy and Salhi (2005) [16] also proposed insertion based heuristics in order to solve VRPSPD. The basic steps of these heuristics are constructing partial routes for a set of customers, and then inserting the remaining customers to the existing route.

B) Problem Formulation

The problem formulation presented in this section is based on model proposed by Jan Detholff (2001). The VRPSPD problem can be described as follows: Given a single depot distribution/collection system, there are a set of customers each of whom has both pickup demand and delivery demand simultaneously. Each customer should be served once by a single vehicle. A homogeneous fleet of vehicles with given capacity and maximum travel distance deliver the goods from the depot to customers as well as pick up the goods from customers and back to the depot. The objective is to minimize the total travel distance of the entire fleet.

Let first define some notations and decision variables for the problem.

1) Notations:

a) Sets

J: Set of all customer locations
J₀: Set of all nodes, i.e. customer locations and depot,
J₀ = J₀ ∪ {0}
V: Set of all vehicles

b) Parameters

C: Vehicle capacity
C₀: Distance between nodes, i ∈ J₀, j ∈ J₀, i ≠ j:
C₀ = M, i ∈ J, C₀₀ = 0
D_j: Delivery amount of customer j ∈ J
n: Number of nodes, i.e., n = |J₀|
P_j: Pick-up amount of customer j ∈ J
M: Large number,
c) **Decision Variables**

- \( l'_v \): Load of vehicle \( v \in V \) when leaving the depot
- \( l_j \): Load of vehicle after serving customer \( j \in J \)
- \( \pi_j \): Variable used to prohibit subtours;
- \( x_{ijv} \): Binary variable indicating whether vehicle \( v \in V \) travels directly from node \( i \in J_0 \) to node \( j \in J_0 \) (\( x_{ijv} = 1 \)) or not (\( x_{ijv} = 0 \))

2) **Mathematical model:**

The corresponding mathematical model of VRPSPD is given as follows:

\[
\text{Minimize } z = \sum_{i \in J_0} \sum_{j \in J_0} \sum_{v \in V} c_{ij} x_{ijv} \tag{1}
\]

subject to

\[
\sum_{i \in J_0} \sum_{v \in V} x_{ijv} = 1 \tag{2}
\]

\[
\sum_{i \in J_0} x_{isv} = \sum_{i \in J_0} x_{sjv} \tag{3}
\]

\[
l'_v = \sum_{i \in J_0} \sum_{v \in V} x_{ijv} = 1 \tag{4}
\]

\[
l_j = l'_j - D_j + P_j - M(1 - x_{ijv}) \tag{5}
\]

\[
l_j = l_i - D_j + P_j - M(1 - \sum_{v \in V} x_{ijv}) \tag{6}
\]

\[
l'_v \leq C \tag{7}
\]

\[
l_j \leq C \tag{8}
\]

\[
\pi_j \geq \pi_i + 1 - n(1 - \sum_{v \in V} x_{ijv}) \tag{9}
\]

\[
\pi_j \geq 0 \tag{10}
\]

\[
x_{ijv} \in \{0,1\} \tag{11}
\]

In the model above, the objective function (1) aims to minimize the total travel distance. Constraints (2) assure servicing each customer node exactly once. Constraints (3) assure that if a vehicle arrives at a customer, then the same vehicle must also leave it. Initial vehicle loads are determined by constraint (4), each vehicle's initial load is the accumulated demand of all customer nodes assigned to that vehicle. Constraint (5) balances the load of vehicles after vehicles visit the first customer node on their route. For other customer nodes the loads of vehicles are calculated as in (6) similarly through their routes. The constraints (7) and (8) ensure load amount of a truck stay under the capacity limits. Constraints (9) are sub-tour elimination constraints and (10) maintains non-negativity of \( \pi_j \) and \( x_{ijv} \) is a binary decision variable as in (11).

III. **PROPOSED SAVINGS BASED ALGORITHM**

VRPSPD belongs to the class of NP-hard problems, for that reason the exact solution methods become highly time-consuming as the problem instances increase in size.

![Procedure of proposed algorithm](image)

Clarke and Wright in 1964 proposed a savings algorithm for solving VRP with more than one vehicle. The basic savings concept expresses the cost savings obtained by joining two routes into one route as illustrated in figure 3 [4].
Initially in figure 3(a) customers i and j are visited on separate routes. An alternative to this is to visit the two customers on the same route, for example in the sequence i-j as illustrated in figure 3(b).

In the first step of the savings algorithm Euclidean distance is calculated using the data and then the savings for all pairs of customers are calculated. Then all pairs of customer points are sorted in descending order of the savings. Subsequently, from the top of the sorted list of point pairs one pair of points is considered at a time. When a pair of points i-j is considered, the two routes that visit i and j are combined. All customers are allocated to vehicles in this manner. By using neighbouring list, customers assigned to the same vehicle are sequenced and at the last step the total distance travelled is calculated by summing up the distance travelled by each vehicle.

IV. COMPUTATIONAL EXPERIMENTS

In order to evaluate the performance of the proposed approach, twenty four medium-sized test problems are studied. These test problems are obtained by adding pick-up demand data to a selected set of CVRP problems from Augerat et al. (1995).

The all problems instance have different customer locations, and have different pickup and delivery demands from each customer. The problem characteristics are summarized in Table 1.

For example, A-n32-k6 denotes that the problem instance belongs to ‘A’ series, number of nodes is 32 (n32) and maximum number of vehicle that can be used is 5 (k5).

In each instance, Euclidean distance between two customers was calculated by their coordinates in the given data and the objective is to minimize the total distance covered by the entire vehicle.

The developed algorithm was programmed in MATLAB R2012b and executed on Intel Core i3 processor @ 2.10 GHz computer system, with 3GB RAM.
V. RESULT AND DISCUSSION

The results of the computations and CPU times with the details of test problems are shown in Table 2.

Column 2 shows the test instance number; column 3 indicated the no. of vehicle; column 4 shows the total distance travelled by vehicle and column 5 represents the computational / CPU time taken for solving the test instance.

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Instance</th>
<th>No. of Vehicle</th>
<th>Total Distance</th>
<th>Computational Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A-n32-k5</td>
<td>4</td>
<td>1195.4</td>
<td>0.0784</td>
</tr>
<tr>
<td>2</td>
<td>A-n33-k5</td>
<td>5</td>
<td>1092.9</td>
<td>0.2548</td>
</tr>
<tr>
<td>3</td>
<td>A-n33-k6</td>
<td>5</td>
<td>1129.5</td>
<td>0.2738</td>
</tr>
<tr>
<td>4</td>
<td>A-n34-k5</td>
<td>5</td>
<td>1342.3</td>
<td>0.1933</td>
</tr>
<tr>
<td>5</td>
<td>A-n36-k5</td>
<td>4</td>
<td>1232.5</td>
<td>0.2685</td>
</tr>
<tr>
<td>6</td>
<td>A-n37-k5</td>
<td>5</td>
<td>1185.8</td>
<td>0.2647</td>
</tr>
<tr>
<td>7</td>
<td>A-n37-k6</td>
<td>5</td>
<td>1199.2</td>
<td>0.2676</td>
</tr>
<tr>
<td>8</td>
<td>A-n38-k5</td>
<td>5</td>
<td>1309</td>
<td>0.0932</td>
</tr>
<tr>
<td>9</td>
<td>A-n39-k5</td>
<td>5</td>
<td>1338.6</td>
<td>0.2732</td>
</tr>
<tr>
<td>10</td>
<td>A-n39-k6</td>
<td>5</td>
<td>1255</td>
<td>0.2635</td>
</tr>
<tr>
<td>11</td>
<td>A-n45-k6</td>
<td>6</td>
<td>1656.7</td>
<td>0.2824</td>
</tr>
<tr>
<td>12</td>
<td>A-n45-k7</td>
<td>6</td>
<td>1693.8</td>
<td>0.2887</td>
</tr>
<tr>
<td>13</td>
<td>B-n31-k5</td>
<td>4</td>
<td>880.64</td>
<td>0.2594</td>
</tr>
<tr>
<td>14</td>
<td>B-n34-k5</td>
<td>5</td>
<td>1298.4</td>
<td>0.2703</td>
</tr>
<tr>
<td>15</td>
<td>B-n35-k5</td>
<td>5</td>
<td>1395.2</td>
<td>0.2516</td>
</tr>
<tr>
<td>16</td>
<td>B-n38-k6</td>
<td>6</td>
<td>1258.9</td>
<td>0.2593</td>
</tr>
<tr>
<td>17</td>
<td>B-n39-k5</td>
<td>5</td>
<td>910.64</td>
<td>0.2609</td>
</tr>
<tr>
<td>18</td>
<td>B-n41-k6</td>
<td>6</td>
<td>1421.6</td>
<td>0.2666</td>
</tr>
<tr>
<td>19</td>
<td>B-n43-k6</td>
<td>6</td>
<td>1025.1</td>
<td>0.2968</td>
</tr>
<tr>
<td>20</td>
<td>B-n44-k7</td>
<td>6</td>
<td>1244.8</td>
<td>0.2819</td>
</tr>
<tr>
<td>21</td>
<td>P-n16-k8</td>
<td>7</td>
<td>631.073</td>
<td>0.2312</td>
</tr>
<tr>
<td>22</td>
<td>P-n19-k2</td>
<td>2</td>
<td>295.344</td>
<td>0.2323</td>
</tr>
<tr>
<td>23</td>
<td>P-n20-k2</td>
<td>2</td>
<td>297.271</td>
<td>0.2175</td>
</tr>
<tr>
<td>24</td>
<td>P-n23-k8</td>
<td>8</td>
<td>758.498</td>
<td>0.0583</td>
</tr>
</tbody>
</table>

The solution found by the proposed approach for the instance A – n36 – k5 is illustrated in figure 4. The total distance travelled by all vehicles is obtained as 1232.5. The routes for the solution are listed in the following:


There are 4 vehicle used in the problem instance. The route details given above shows the sequence in which the vehicle visits its customers without violating any constraints. Every vehicle starts from the depot and visits all customers allocated to it and returns back to the depot.

Computational experiments with benchmark problem instances show that the proposed approach gives results within shorter time frame whereas the other approaches in literature takes about 60 to 200 seconds to reach the result. Considering these results and CPU times, it can be stated that, savings based proposed approach is efficient and performs well.

VI. CONCLUSION

The VRP is a well-known combinatorial optimization problem. Customers require simultaneous pick-up of goods from their location in addition to delivery of goods to their location in some cases.
The VRP-SPD is an extension of the CVRP and considers simultaneous distribution and collection of goods to/from customers. Collaborative vehicle routing problem with simultaneous pickup and delivery in transportation could integrate the limited resources of multiple logistics companies to avoid unreasonable investment and waste and achieve the efficient operation of the logistics system. Although the saving heuristic of Clarke and Wright (1964) is widely used in VRP, no such heuristics exist for VRPSPD.

Twenty four VRPSPD test problems of Augerat et al. were tested to find the efficiency of the approach. According to the results of computational experiments, it can be concluded that the proposed savings based approach both performs well and is efficient.

Several extensions are possible for the VRPSPD. One worth mentioning here is the possibility of using heterogeneous or mixed fleet vehicles. Indeed, the flexibility afforded by the use of different vehicles may yield further reductions in total cost and vehicle utilization. Another extension would be to consider the time windows of the problem, where vehicles serve all the clients during a given time interval without violating capacity constraints. These problems with application of Metaheuristic techniques like Genetic Algorithm, Particle swarm optimization etc... can be studied in future work.

REFERENCES


