An Improved Heuristic Based on Clustering and Genetic Algorithm for Solving the Multi-Depot Vehicle Routing Problem

Fadoua Oudouar, Abdellah El Fallahi, El Miloud Zaoui

Abstract: This paper introduces the multi depot vehicle routing problem (MDVRP) with location depot, a hard combinatorial optimization problem arising in several applications. The MDVRP with location depot based on two well-known NP-hard problems: the facility location problem (FLP) and the multi depot vehicle routing problem (MDVRP) with location depot. In the first phase, the problem consists of selecting on which sites to install a warehouse and assigning one and only one warehouse to each customer. In the second phase, for each depot, we search to solve a vehicle routing problem (VRP), multiple vehicles homogenous leave from a single depot and return to the same one. Each route must respect the capacity of each vehicle. The goal of this problem is to minimize the total distances to the performed routes. The MDVRP with location depots is classified as an NP-hard problem. Hence, the use of exact optimization methods may be difficult to solve this problem. In this work, we propose a method to solve it to optimality. The proposed procedure initially uses K-means algorithm to optimize the location selection and customer assignment, then planning the routes from the selected warehouses to a set of customers using Clarke and Wright saving method. The routes from the resulting multi-depot vehicle-routing problem (MDVRP) are improved using a Genetic Algorithm (GA). The proposed approach is tested and compared on a set of twelve benchmark instances from the MDVRP literature. The computational experiments confirm that our heuristic is able to find best solutions.

Keywords: Location depots, routing, K-means method, Clarke and Wright algorithm, Genetic Algorithm.

I. INTRODUCTION

Vehicle Routing Problem is a generalisation of a large number of routing problems. The problem is to find the optimal number of routes to serve a set of customers while minimizing the routing costs. [1]. The objective is to find a set of routes for an identical vehicle to serve a set of customers while minimizing the routing costs. The MDVRP is an NP-hard type of VRP [2]–[4]. It is one of the most popular routing problems. This problem is used to determine the optimal set of routes for satisfying the delivery demands of customers assigned to the corresponding depot. Since, the MDVRP is NP-hard because it generalizes the VRP as further complexity is added through multiple depots. The basic goal is to grouping customers into clusters, where each one of the clusters is served by a set of vehicles from the nearest depot. To solve the MDVRP [5] search to solve the routing problem with two depots. [6] propose a branch-and-bound algorithm to the problem. Recently, many researchers have developed metaheuristics for the MDVRP, which included Genetic Algorithm (GA), Tabu Search (TS), among others. [7] proposed a genetic clustering. They developed a clustering method inter a genetic algorithm. [8] studied the MDVRP with time windows and heterogeneous vehicles. The problem objective aims to minimize the total cost, including travelling distance, times costs and vehicle utilization cost. In 2008 [9] developed an hybrid genetic algorithm (HGA), which combined three different heuristics: the Clarke and Wright (CW), nearest neighbor (NN), and iterated swap. In their approaches, theses authors are used the two first heuristics to generate an initial solution, then the solutions are improved using the last one. [10] presented the branch and price algorithm to solve that model. The principle of MDVRP with location depots is to alternate between a depot location problem and a routing problem [11]. A survey of these studies, based on either exact methods or heuristic, can be found in Montoya-Torres et al 2015 [12]. This type of problem combines two decisions simultaneously: planning vehicle routes and assigning routes to depots. The main contributions of this work are the following. We first model the MDVRP with homogeneous fleet, and we develop a hybrid solution algorithm. In The proposed algorithm, the clustering stage and the vehicle routing stage are alternated iteratively. We then apply our algorithm for instances from the literature, and compare its performance against published solutions [13].

The contribution of this paper is organized as follows. Section 1 defines the problem, and we describe the proposed linear programming model. In the section 2, we present the depot-location phase solved by clustering method. The solution of the vehicle routing problem for each depot is described in section 3. The numerical results and the computational experiments are presented in section 4. The last section, gives some concluding and remarks.

II. PROBLEM DESCRIPTION AND FORMULATION

In the MDVRPLD, a set of customers with known demands, a set of identical vehicles with limited capacity are given. Each customer is to be assigned to a depot, and visited by only one vehicle. The objective is to locating a set of facilities, affecting customer to the nearest depot and finding the optimal routes for each depot. This is a challenging problem because, it combines the facility location and the vehicle routing decisions. Figure 1 illustrates an example.
An improved heuristic based on clustering and genetic algorithm for solving the Multi-Depot Vehicle Routing Problem

for the problem. This example contains three depots and fifteen customers. The first route is executed (Depot 1 - Customer 1 - Customer 3 - Customer 5 - Depot 1), the second (Depot 2 - Customer 6 - Customer 4 - Customer 2 - Depot 2) [14].

![Depot and Customers](image)

Fig. 1: MDVRP with six customers and two depots

The MDVRP with location depot studied in this paper can be formally described as follows:
- Each route must begin and return at the same warehouse.
- Each customer is served by a single vehicle.
- The sum of the demands assigned to each route should not exceed the vehicle capacity.

Sets:
- \( E \) : Set of nodes \((i, j)\).
- \( V \) : Set of nodes \(V = V_c \cup V_d\).
- \( V_c \) : Set of customers to be visited \(V_c = \{1, \ldots, N\}\).
- \( V_d \) : Set of potential facilities \(V_d = \{N + 1, \ldots, N + M\}\).
- \( K \) : Number of identical vehicles.

Decision variables:
- \( X_{ijk} \in \{0,1\} \forall i, j, k \) (9)
- \( Y_i \in \{0,1\} \forall i \) (10)

The objective (1) minimizes the total travelling distance of all vehicles. Eqs. (2) and (3) represent the constraints that each client is visited by exactly one vehicle. The flow conservation constraint is present in Eq. (4). Vehicle capacity is present in Eq. (5). Eqs. (6) and (7) express that vehicle availability should be respected. In Eq. (8), show the subtour elimination. Finally, the binary variables are defined by Eqs. (9) and (10).

### III. MATERIALS AND METHODS

In this section, three phases to solve the problem. The first phase, based (involved) on clustering, the goal is allocate customers to depots. For this reason, we use k-means algorithm to clusters the customers. The proposed method was used for clustering the customers, determine the depots to be opened, and finally, allocate the clusters to open depots. The other phase, assigned clusters to vehicles using Clarke and Wright, and then sequence those ones on the same tour. The last phase, a genetic algorithm is applied to optimize the routes between the depots and the assigned clients, and to provide the best feasible solution. The steps of solution procedure are drawn in Fig. 2.

![Solution Procedure](image)

**A. K-means algorithm**

The K-Means method is an unsupervised learning algorithm that is employed to solve the clustering problems. It is defined by [16]. K-Means algorithm has a diverse and rich history (Jain, 2010). The principle of this method is to divide n individuals into k clusters. Where each individual or data points are affected to the nearest cluster [17].

In this work, we define k depots as centers, each center for a cluster. The next step is to allocate each customer to the nearest warehouse. The objective is to visiting all customers. Fig. 3 illustrates an example of distribution of customers and the location of the optimal depots.

Suppose a dataset (Customers) \( C = \{c_i\}_{i=1}^{N} \) of N customers in \( \mathbb{R}^n \) and a number k of depots. To find depot centers \( D_1, D_2, \ldots, D_K \) in \( \mathbb{R}^n \) such that the sum of distance between each customer \( c_i \) and its nearest depot center \( D_j \) is minimized using the following formulation:

\[
\min_{i=1}^{k} \sum_{j=1}^{k} \|c_i - D_j\|^2 
\]
The proposed algorithm is composed of the following steps:
1. Initialize a K-depot randomly or using the points of k-customers.
2. Assign each customer dataset \( c_i \) to the nearest depot \( D_j \):
   \[
   c_i \in D_j, \text{ if } ||c_i-D_j|| < ||c_i-D_l|| \\
   \text{for } i = 1, \ldots, N, j \neq l, \text{ and } j = 1, \ldots, k
   \]
3. Recalculate the new depot coordinate:
   \[
   D_l = \frac{1}{N_l} \sum_{i \in D_l} c_i
   \]
4. Iterate step two and three until the position of depots no changed.
5. Allocate the customers to the nearest depot.

This heuristic is influenced by the choice of initial K-point. In this work, a new version of the modified k-means algorithm is proposed. We apply an auxiliary phase to generate a start points (Initial depot coordinates).

The initial depot location are calculated as follow:
**Step 1:** Choose the first depot from the customers (same coordinates of a customer).
**Step 2:** From this coordinates, we calculate the distance between this depot and all customers. After, we choose the farther customer as a second depot.
**Step 3:** From these coordinates, we calculate the distance between these depots and all customers and we select the farther customer as the third depot.
**Step 4:** We repeat step 3 until we reach the number of depots.

After these steps, we will follow the second, the third and the fourth step of K Means algorithm.

**Fitness**

For our problem, the goal is to minimize the travelling distance of all routes among n depots. Let \( G_l \) be the total delivery distance required by a depot p and \( \min(G_l) \) represent the minimum delivery distance traversed by all vehicles:

\[
G_l = \sum_{p=1}^{n_p} d[k(n_p),k(0)] + \sum_{i=1}^{n_c} d[k(i-1),k(i)]
\]

- \( d(v,w) = \sqrt{(x_v-x_w)^2 + (y_v-y_w)^2} \) : Travel distance of a vehicle from customer v to w.
- \( k(i) \) : Position of the customer i.
- \( k(0) \) : Position of the depot.
- \( n_c \) : Number of customers in route r.
- \( n_p \) : Number of vehicles in depot p.

The fitness formulation is defined as:

**Fig. 3:** Visual illustration of the example solution of clustering

**B. Clarke and Wright (CW)**

The CW saving method is one of the most popular algorithms proposed for the different variant of the routing problem. This approach was developed in 1964 and is still feasible today [18], due to its simplicity and efficient calculation speed [19]. However, several implementation of CW to solve different type of routing problem [20], [21].

The Clarke and Wright algorithm is composed of the following steps [19]:
1. Calculate the distance matrix \( (d_{ij}) \), \((x_i, y_i)\) and \((x_j, y_j)\) are the geographical locations of customers i and j:
   \[
   d_{i,j} = \sqrt{(x_i-x_j)^2 + (y_i-y_j)^2}
   \]
2. The saving value between customers i and j is calculated:
   \[
   s_{ij} = d_{0i} + d_{0j} - d_{ij}
   \]
   Where \( d_{0i} \) is the traveling distance between the depot and the customer i.
3. List the saving \( s_{ij} \) in decreasing order.
4. Starting from the first in the saving list, Clarke and Wright includes link between i and j in a route if no constraints (capacity of vehicles, time windows, distance ….) will be violated. As we show in Fig.4.

**Fig. 4:** Presentation of the saving concept

**C. Genetic Algorithm**

John Holland introduces genetic Algorithm (GA) for the first time in the 1960s [7]. GA is a stochastic approach inspired by the class of evolutionary algorithms [22], they represent a type of optimization techniques. Due to its great flexibility, these algorithms are an excellent approach for solving complex problem, and generate high-quality solutions.

The principal of this method was divided on three levels: reproduction, natural selection and the diversity of individuals. In the literature, GA has received great attention to solve hard optimization problems, such as VRP, Capacitated VRP [23], MDVRP[9], MDVRP Time windows [24] and LRP (Location Routing problem)[25]. They are used to find the maximum or minimum of a function. In this paper, the objective is to apply GA for minimizing the traveling cost for a fleet of vehicles [26].
An improved heuristic based on clustering and genetic Algorithm for solving the Multi-Depot Vehicle Routing Problem

$$F = \sum_{p=1}^{dist_{total}} \min(C_t)$$

Selection

Selection step represents the process where the parents are selected for mating and produce new generation. For solving the MDVRP, we use the tournament method [27]. This selection strategy is first picked a set of k candidates from the population in a random manner, and then compares their fitness. The best individuals is the fittest one.

Crossover

In the crossover phase, the GA selects two parent P1 and P2 from tournament selection to generate from every process two offspring. In the MDVRP, the parent represent the routes. The goal of using crossover phase is to generate new routes from the routes that are found by using Clark and Wright algorithm. In the examples shown in Fig. 5, there are 9 customers designated.

![Parents and Children](image)

**Fig. 5:** Crossover of a set of customers in the same route

Mapping set

![Mapping Set](image)

Mutation

The inversion mutation is one of the popular method that used in finding an MDVRP solution using GA. A set of genes is selected from the parent and flips it to form an offspring. This mutation performs the solution in much higher rate. The inversion mutation works with one chromosome only, which is shown in Fig. 6.

![Mutation Example](image)

**Table- I:** Instance descriptions

<table>
<thead>
<tr>
<th>Instances</th>
<th>Cu</th>
<th>De</th>
<th>Ve</th>
<th>Ve/De</th>
<th>Cap/Ve</th>
</tr>
</thead>
<tbody>
<tr>
<td>P01</td>
<td>50</td>
<td>4</td>
<td>16</td>
<td>4</td>
<td>80</td>
</tr>
<tr>
<td>P02</td>
<td>50</td>
<td>4</td>
<td>16</td>
<td>4</td>
<td>160</td>
</tr>
<tr>
<td>P03</td>
<td>75</td>
<td>5</td>
<td>15</td>
<td>3</td>
<td>140</td>
</tr>
<tr>
<td>P04</td>
<td>100</td>
<td>2</td>
<td>16</td>
<td>8</td>
<td>100</td>
</tr>
<tr>
<td>P05</td>
<td>100</td>
<td>2</td>
<td>10</td>
<td>5</td>
<td>200</td>
</tr>
<tr>
<td>P06</td>
<td>100</td>
<td>3</td>
<td>18</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>P07</td>
<td>100</td>
<td>4</td>
<td>16</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>P08</td>
<td>249</td>
<td>2</td>
<td>28</td>
<td>14</td>
<td>500</td>
</tr>
<tr>
<td>P09</td>
<td>249</td>
<td>3</td>
<td>36</td>
<td>12</td>
<td>500</td>
</tr>
<tr>
<td>P10</td>
<td>249</td>
<td>4</td>
<td>32</td>
<td>8</td>
<td>500</td>
</tr>
<tr>
<td>P11</td>
<td>249</td>
<td>5</td>
<td>30</td>
<td>6</td>
<td>500</td>
</tr>
<tr>
<td>P12</td>
<td>80</td>
<td>2</td>
<td>10</td>
<td>5</td>
<td>60</td>
</tr>
<tr>
<td>PR1</td>
<td>48</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>200</td>
</tr>
</tbody>
</table>

- Cu: Set of Customers.
- De: Set of Depots
- Ve: Set of Vehicles
- Ve/De: Set of Vehicles/Depot
- Cap/Ve: Capacity/Vehicle

The details of each MDVRP instances are reported in Table I. The first column of the table as the name of instances, the second column indicates the number of customers in each instances, the third number of candidate depots. The set of the vehicle for each benchmark, the number of vehicles for each depot and the vehicle capacity are presented in columns 4–6. Computational results obtained by using this approach to a set of instances are detailed in Table II. In this table, the second columns contains the number of depots used in our solution. For clarity, columns 3-7 contain, respectively, the traveling cost obtained for each depot. The total cost of each instance is presented in column 8.

IV. RESULTS AND DISCUSSION

The proposed approach has implemented in the C programming language, and the computational results have been tested on PC I5, RAM 8Go. The performance of the proposed heuristics in solving our problem is tested by considering 12 standard MDVRP benchmark instances taken from the literature [28].

Retrieval Number: C5256098319/2019©BEIESP
DOI:10.35940/ijrte.C5256.098319

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication

6538
Table- II: Numerical results for the MDVRP instances

<table>
<thead>
<tr>
<th>Instance</th>
<th>De</th>
<th>Traveling cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>P01</td>
<td>4</td>
<td>131.93, 96.11</td>
</tr>
<tr>
<td>P02</td>
<td>4</td>
<td>101.85, 165.88</td>
</tr>
<tr>
<td>P03</td>
<td>5</td>
<td>118.07, 111.96</td>
</tr>
<tr>
<td>P04</td>
<td>2</td>
<td>497.69, 522.88</td>
</tr>
<tr>
<td>P05</td>
<td>2</td>
<td>375.06, 402.75</td>
</tr>
<tr>
<td>P06</td>
<td>3</td>
<td>260.09, 363.84</td>
</tr>
<tr>
<td>P07</td>
<td>4</td>
<td>171.58, 190.21</td>
</tr>
<tr>
<td>P08</td>
<td>2</td>
<td>2091.79, 2484.09</td>
</tr>
<tr>
<td>P09</td>
<td>3</td>
<td>1365.7, 1481.14</td>
</tr>
<tr>
<td>P10</td>
<td>4</td>
<td>721.01, 1037.11</td>
</tr>
<tr>
<td>P11</td>
<td>5</td>
<td>571.89, 563.23</td>
</tr>
<tr>
<td>P12</td>
<td>2</td>
<td>708.86, 735.93</td>
</tr>
<tr>
<td>PR1</td>
<td>4</td>
<td>137.14, 286.89</td>
</tr>
</tbody>
</table>

Table- III: Number of vehicles for each depot and instance

<table>
<thead>
<tr>
<th>Instance</th>
<th>Ve</th>
<th>Ve /De</th>
</tr>
</thead>
<tbody>
<tr>
<td>P01</td>
<td>16</td>
<td>3 2 4 4 - 13</td>
</tr>
<tr>
<td>P02</td>
<td>8</td>
<td>1 2 2 2 - 7</td>
</tr>
<tr>
<td>P03</td>
<td>15</td>
<td>3 2 2 3 3 13</td>
</tr>
<tr>
<td>P04</td>
<td>16</td>
<td>7 8 - - - 15</td>
</tr>
<tr>
<td>P05</td>
<td>10</td>
<td>4 5 - - - 9</td>
</tr>
<tr>
<td>P06</td>
<td>18</td>
<td>5 6 7 - - 18</td>
</tr>
<tr>
<td>P07</td>
<td>16</td>
<td>4 4 4 4 - 16</td>
</tr>
<tr>
<td>P08</td>
<td>28</td>
<td>12 14 - - - 26</td>
</tr>
<tr>
<td>P09</td>
<td>36</td>
<td>9 9 9 - - - 27</td>
</tr>
<tr>
<td>P10</td>
<td>32</td>
<td>5 8 7 7 - - 27</td>
</tr>
<tr>
<td>P11</td>
<td>30</td>
<td>5 5 6 7 28</td>
</tr>
<tr>
<td>P12</td>
<td>10</td>
<td>4 6 - - - 10</td>
</tr>
<tr>
<td>PR1</td>
<td>4</td>
<td>1 1 1 1 - 4</td>
</tr>
</tbody>
</table>

The results in table IV show that the proposed approach has obtained 5 new best solutions out of the 13 instances. This result indicates that our method can provide a competitive result in term of global average gap and number of best solutions. For example the instance P03 with five facilities, the total cost equal to 641, 19 and an average Gap of -0.03%. The same for the P01, P07, P10 and P11.

Fig 7: Comparison of distances between proposed mod-el and best solution

The presentation of results for each instance is shown in fig. 7, where the blue line represents the heuristic value, and the red line indicates the best-known solution of the literature. The average relative gap between the proposed approach and the best-known solution (BKS) are illustrated in this figure. The relative gap value is defined as 100 ×(solution obtained – known solution)/known solution. It was observed from the computational results, the effectiveness of this approach in term of solution quality.

V. CONCLUSION

In this study, we propose an effective three-phase heuristic to solve the multi depot vehicle routing problem with location depot, which aims to determine the location strategy and the routing plan of fleet homogenous vehicles. In the first phase, we apply the several clustering algorithm k-mean to determine the clusters. The second phase, we use the Clark and Wright method to determine the optimal number
An improved heuristic based on clustering and genetic Algorithm for solving the Multi-Depot Vehicle Routing Problem

of routes for each depot. In the last phase, the scheduled routes are optimized using GA. The objective of this problem is to minimize the total travelling cost.

We compared the proposed algorithm with the best solution for MDVRP on a set of benchmark instances from the literature. The obtained results provide the efficacy and effectiveness of the proposed approach. The results suggest that the proposed algorithm can be adapted to other Vehicle Routing Problems as the periodic vehicle routing problem (PVRP) and the location routing problem (LRP).

As a future research on the MDVRP, we are planning to use time windows to serve each customer and see how this constraint could affect the total cost.

REFERENCES