Multi Phase Methodology for Solving the Multi Depot Vehicle Routing Problem with Limited Supply Capacity at the Depots

Javier de Prado, Sandro Moscatelli, Pedro Piñeyro[ID], Libertad Tansini[✉][ID], and Omar Viera

Department of Operations Research, Computer Science Institute (InCo), Faculty of Engineering, Universidad de la República, Montevideo, Uruguay
{moscatel,ppineyro,libertad,viera}@fing.edu.uy

Abstract. This paper focuses on a capacitated multi depot vehicle routing problem, where each depot has a finite supply capacity to meet the customers demand. To solve this problem we propose a multi phase methodology, that extends the "cluster first, route second" approach. It is based on iterative routings to find and reassign misplaced customers with respect to the depots and with the objective of improving the final routing. Several assignment and routing algorithms are considered to evaluate the proposed methodology under different settings. A mathematical model of the problem is given to perform a comparative study of the methodology against an exact solution method. The results obtained from the numerical experiments carried out allow us to conclude that the methodology can be successfully applied to the capacitated multi depot vehicle routing problem.

Keywords: Multi depot vehicle routing problem · Heuristics · Supply capacity · Clustering · Assignment

1 Introduction and Related Works

We address the problem of distribution of goods from several depots to a set of geographically dispersed customers with known coordinates and demand, assuming finite supply capacity at each depot and an unlimited fleet of homogeneous and capacitated vehicles. We refer to this problem as the Capacitated Multi-Depot Vehicle Routing Problem (CMDVRP). The objective is to determine a set of routes starting and ending at each depot, minimizing the total distance traveled and subject to the supply capacity of each depot and the capacities of the vehicles. The CMDVRP can be found in recent real life applications such as emergency facilities location-routing and city logistics problems [20,22]. The CMDVRP is an NP-hard problem since it can be considered an extension of the

Multi-Depot Vehicle Routing Problem (MDVRP), which in turn is an extension of the classical Vehicle Routing Problem (VRP) [8].

We present here a novel multi phase methodology to solve the CMDVRP inspired by the "cluster first, route second" approach. The initial phase consists of the assignment of costumers to depots and the final phase produces the routing of the VRPs related to all depots. Between these two phases, there is an intermediate phase for the reassignment of customers to depots with the aim to obtain a high quality solution in the cluster first part, improving in this way the final routing phase. The detection and reassignment of customers are based on a combination of misplaced-customer criterion and routing algorithm. A misplaced customer is reassigned to another depot, if this reassignment improves the cost of the general solution, which is the objective of the proposed methodology. The main idea behind the proposed methodology is that the complexity of the algorithms used in each phase can be chosen by the decision makers according to their needs and possibilities. The strength of the methodology is to provide good quality solutions in reasonable times, even in the case of using simple algorithms (easy to understand and code).

As far as we know, only few authors focus on the CMDVRP, and in particular, by means of the "cluster first, route second" approach. Giosa et al. [9] describe and compare several assignment algorithms for the clustering phase. Tansini et al. [17] compare the results obtained by a set of heuristic algorithms for the assignment of customers to depots with assignments obtained from solving the Transport Problem. Six heuristics for the clustering problem (assignment of customers to depots) are presented and analyzed in [10]. Also [18] consider this approach for the real-life problem of milk collection. Allahyari et al. [1] tackle the CMDVRP extension in which every customer is satisfied either by visiting the customer or by being located within an acceptable distance from at least one visited customer. Calvet et al. [4] consider the CMDVRP for the case of customers with stochastic demand and supply constraints on the depots due to the limited number of capacitated vehicles assigned to each of them. A collaborative routing problem with shared carriers and multiple depots (wholesalers) with limited storage is tackled in [21].

We note that many authors have considered the multi-depot vehicle routing problem with limited capacity on vehicles and/or route lengths, but not on the supply depots. For instance, Vidal et al. [19] propose a framework to solve the MDVRP, the Periodic VRP (PVRP), and the multi-depot periodic VRP with capacitated vehicles and constrained route duration. Contardo and Martinelli [6] suggest an exact algorithm for the MDVRP under capacity and route length constraints, exploiting the vehicle-flow and set-partitioning formulations. Recently, Pessoa et al. [15] propose a generic branch-cut-and-price solver for different vehicle routing variants and related problems.

The remainder of this paper is organized as follows. In Sect. 2 we introduce the mathematical formulation for the CMDVRP. The proposed methodology for solving the CMDVRP is further described in Sect. 3. In Sect. 4 we present the results of the comparison between the methodology against exact methods and

we also analyze the effectiveness of the exploration phase of the methodology. Finally, in Sect. 5, we provide the conclusions and some directions for future research.

2   The Capacitated Multi-Depot Vehicle Routing Problem (CMDVRP)

The CMDVRP can be formally described as follows, extending that presented in [14] for the MDVRP. Let $G = (V, E)$ be a directed graph, where V denotes the set of nodes $\{1, ..., n\}$ and $E \subseteq V \times V$ the set of arcs. Let D be the set of depot nodes $\{1, ..., m\}$, with $1 \leq m < n$, and U the set of customer nodes $\{(m + 1), ..., n\}$. For each node $i \in V$ there is a related quantity $q_i \geq 0$ that represents either the supply capacity for nodes $i \in D$ or the demand requirements in the case of nodes $i \in U$. For each arc $(i, j) \in E$ there is a routing cost $c_{i,j} \geq 0$. Let also consider the set of the possible routes $R = \{1, ..., (n - m)\}$. A route r can be defined as either the empty set or a finite sequence of at least three elements of V satisfying the following conditions: 1) in the extremes there is the same node i, with $i \in D$, 2) the internal nodes are customers nodes j with $j \in U$, and 3) for any pair of nodes $j, k \in U$, we have that $j \neq k$. We assume that for each route r there is a vehicle of capacity $p \geq 0$. Then, the objective is to determine the set of routes r in R in order to fulfill the demand of each customer without exceeding the vehicle and depot capacities, minimizing the total cost of routing. To formulate the CMDVRP as a Mixed Integer Linear Programming (MILP) we define the binary variables x_{ijkr} to be equal to 1 only if the arc (i, j) is in the route r of the depot k; 0 otherwise. Thus, the MILP proposed for the CMDVRP is as follows:

$$\min \sum_{i \in V} \sum_{j \in V} \sum_{k \in D} \sum_{r \in R} c_{ij} x_{ijkr} \tag{1}$$

subject to:

$$\sum_{j \in V} \sum_{k \in D} \sum_{r \in R} x_{ijkr} = 1, \qquad \forall i \in U \tag{2}$$

$$\sum_{j \in V \setminus \{i\}} x_{ijkr} = \sum_{j \in V \setminus \{i\}} x_{jikr}, \qquad \forall i \in V, \forall k \in D, \forall r \in R \tag{3}$$

$$\sum_{i \in U} \sum_{j \in V \setminus \{i\}} \sum_{k \in D} q_i x_{ijkr} \leq p, \qquad \forall r \in R \tag{4}$$

$$\sum_{i \in U} \sum_{j \in V \setminus \{i\}} \sum_{r \in R} q_i x_{ijkr} \leq q_k, \qquad \forall k \in D \tag{5}$$

$$\sum_{j \in U} x_{kjkr} \leq 1, \qquad \forall k \in D, \forall r \in R \tag{6}$$

$$\sum_{j \in U} x_{ijkr} = 0, \qquad \forall i, k \in D, i \neq k, \forall r \in R \tag{7}$$

$$y_i - y_j + (n - m) \sum_{k \in D} \sum_{r \in R} x_{ijkr} \leq n - m - 1, \qquad \forall i, j \in U, i \neq j \tag{8}$$

$$y_i \geq 0, \qquad \forall i \in U \tag{9}$$

$$x_{ijkr} \in \{0, 1\}, \qquad \forall i, j \in V, \forall k \in D, \forall r \in R \tag{10}$$

The objective function (1) is the minimization of the total cost of distance traveled. Constraints (2) state that each customer is included in a single route. Constraints (3) are for the route continuity. Constraints (4) and (5) represent the vehicle and depot capacity, respectively. Constraints (6) and (7) state that one route is assigned at most to a single depot. In (8) are the constraints of Miller-Tucker-Zemlin for subtours elimination [3]. Finally, constraints (9) and (10) are for the domain of values of the decision variables.

Although the main difference between CMDVRP and MDVRP are the constraints of (5), we note that, in general, a more restricted problem is more difficult to solve.

3   Multi-phase Methodology for the CMDVRP

It is worth to note that the assignment problem and the routing problem in the "cluster first, route second" approach are not independent from each other. A bad assignment solution will result in routes of higher total cost, even if an effective routing algorithm is used. Motivated by this, we consider an improvement to this approach, by means of a multi-phase methodology (MPM) for solving the CMDVRP. It begins from an initial assignment of costumers to depots and in the final phase produces the routing of the VRPs related to all depots. We introduce an intermediate phase in which misplaced costumers are detected and may be reassigned to another depot, if it improves the cost of the overall solution. Successive reassignment of misplaced costumers, based on the routing, will in most cases lead to an improvement of the solution. An outline of the proposed methodology for the CMDVRP is as follows:

1. **Assignment phase**: choose and apply an assignment algorithm of customers to depots taking into account demand and supply restrictions. The choice may depend on computational time and other restrictions.
2. **Exploration phase**: choose and apply a routing algorithm for all VRPs related to the depots. Again, the choice may depend on computational time and other restrictions. Then, repeat until no further improvement can be achieved:

(a) Detect misplaced customers based on the current assignment and eventually other restrictions.

(b) Reassign misplaced customers and run the selected routing algorithm. Accept the reassignment if it improves the cost of the overall solution.

3. **Final routing phase**: choose and apply the final routing algorithm for all VRPs related to the depots.

One of the advantages of the suggested MPM is that each phase offers the possibility of choosing different algorithms depending on the specific characteristics of the problem, the problem instances, hardware limitations, time restrictions, etc. They can be exchanged and combined in different manners. Thus, a specific selection of algorithms for each phase produces a particular MPM instantiation that can be considered a heuristic procedure to solve the CMDVRP. Next the MPM phases are explained in more detail and some algorithms that can be used in each one are mentioned.

3.1 Assignment Phase

Since each phase of the methodology offers a great variety of possibilities to instantiate and since there are several known methods that can be used to obtain an initial assignment of customers, in this work we narrow down the study to two assignment schemes.

We use the urgency assignment (Ur) which is a simple and fast assignment method that considers an urgency value μ_c for each customer c that determines the order in which customers are assigned to depots with limited supply capacity [9], as follows:

$$\mu_c = \Big[\sum_{d \in D} dist(c, d) \Big] - dist(c, d') \tag{11}$$

where $dist(c, d)$ is the distance of customer c to depot d, and $dist(c, d')$ is the distance to the closest depot d'. This measure accounts for the cost of assigning a customer to a depot other than its closest depot. Customers with more urgency (higher μ_c value) will be assigned first. Once a depot is complete it will no longer be considered for the further assignments and will hence not participate in the urgency calculations. Note that after each assignment the urgency of some customers must be recalculated.

Alternatively in this study the modified urgency assignment (MUr) is used as another assignment method and is defined as the combination of the urgency assignment [9] and the cluster assignment [10]. Customers are assigned to depots with the same criterion as in the urgency assignment until a fraction of them have been assigned (in this case 1/4) and then finalizes by assigning customers to the closest cluster made up of each depot and the already assigned customers, where it is feasible to assign the customer, i.e. will not exceed the total capacity of the depot.

There are other interesting assignment algorithms that could be explored such as the sweep approach [11] or using a grid or Voronoi diagrams [2]. We note

that some of them do not consider the capacity of the depots and may require an adaptation or post-processing in order to give an acceptable initial assignment.

3.2   Exploration Phase

The exploration phase is the keystone of the proposed methodology. It is characterized by: 1) the definition of misplaced customers; 2) the processing order of the misplaced customers; 3) the routing algorithm used iteratively; 4) the criterion that determines if each misplaced customer should be reassigned or not; and 5) the reassignment strategy. In the following sections we describe the definitions and algorithms to be used in our study for this phase.

Definition of Misplaced Customers: Here, the definition of misplaced customers, the processing order of the misplaced customers, and the criterion that determines if each misplaced customer should be reassigned or not, all of them depend on the routing algorithm that is used iteratively to obtain the results of the VRPs related to all depots.

It is possible to infer that different definitions of misplaced customers lead to different ways of exploring neighboring solutions. The first approach was to define misplaced customers as those whose two closest customers are assigned to another depot. In general, we can define a misplaced customer as that for which its n closest customers are assigned to other depots (possibly different), for certain positive integer $n > 0$. Thus, a more flexible definition considers a customer to be misplaced if considering its n closest customers, m of them are assigned to other depots, where $m \leq n$. Observe that this definition focuses on the cost of the solution, therefore the reassignment strategy considers the supply capacity of the depots.

Other approaches would be to consider constraints such as capacity and time windows in the definition of misplaced customers.

Processing Order of the Misplaced Customers: In this work, misplaced customers are processed in descending order of the following misplaced criterion:

$$\varphi_c = dist(c, d) - \left[\sum_{i=1}^{N} dist(c, c_i) \right] \tag{12}$$

where customer c has been assigned to the depot d and $c_1, ..., c_N$ are its N closest customers not assigned to d, but assigned all to the same depot. The value of φ_c can be positive or negative, where a high positive value of φ_c means that the customer c is very far from the assigned depot compared to the distance to its closest neighbors.

Routing Algorithm Used Iteratively: Three different algorithms for the routing of the customers assigned to each depot were tested in this paper for

the Exploration phase: Clarke and Wright algorithm [5], Sweep [11] and JSprit (available at https://jsprit.github.io/) which is a metaheuristic defined by the ruin-and-recreate principle [16]. It is a highly optimized method that consists of a large neighborhood search that combines elements of simulated annealing and threshold-accepting algorithms. Both Clarke and Wright and Sweep are classical and simple routing algorithms for the VRP, and also Clarke and Wright is a very popular constructive heuristic [12] and Sweep is the most elementary version of petal-type constructive heuristics [12]. It is worth noting that in each iteration, the routing algorithm only needs to bee applied for those depots with reassigned customers, since the others remain unchanged.

Reassignment Criterion: The reassignment criterion used in this paper is to reassign a customer if it produces a lower routing cost than the current assignment. It is important to note that once the reassignments are decided, it is only necessary to run the routing algorithm for the implicated depots. The routing for the rest of the depots remains unchanged.

Reassignment Strategy: Different approaches can be considered for the reassignment strategy. They should describe the conditions and the procedure to assign a misplaced customer to another depot, that will potentially improve the final routing. In general, the demand of customers and the capacities of depots should be considered. In this paper the reassignment strategy is determined by a two-stage procedure executed over an ordered list of misplaced customers. As part of the strategy, it has to be decided the number m of customers with the same target depot that may be considered to be reassigned simultaneously. This section explains the strategy suggested to reassign one misplaced customer ($m = 1$) at a time in the exploration phase.

In the first stage, the reassignment of a misplaced customer i to the depot d' of the closest customer i' is attempted, if d' has enough spare capacity to serve costumer i. If the reassignment produces a better overall routing result, it is accepted and the list of misplaced customers is recalculated. If there is no improvement, the next misplaced customer in order of misplaced criterion is considered to be reassigned. If depot d' does not have enough spare capacity to serve costumer i, then i is reassigned to the closest depot d'' (if it exists) that does have enough spare capacity to serve it. Again, if the reassignment produces a better overall routing result, the reassignment is accepted and the list of misplaced customers is recalculated.

The aim of the second stage is to try reassign the misplaced customers that remain in the list after the first stage. In this stage the same processing is done with the misplaced customers as in the previous stage except in the way of determining the alternative depot d'' and the reassignment moves. Let us assume that depot d' does not have enough spare capacity to serve the misplaced customer i under consideration, with d' as in the first stage. Then, a misplaced customer i'' assigned to d' is determined, that could potentially be reassigned to another depot d'', with d'' the depot of the closest customer to i'', allowing d' to serve the

customer i. If misplaced customer i'' exists, a double reassignment is attempted by means of assigning i'' to depot d'' and i to depot d'. If the double reassignment of customers produces a better overall routing result, the reassignment is accepted and the list of misplaced customers is recalculated.

The two stages described above, are repeated until there are no further misplaced customers (the list is empty) or no misplaced customer reassignment results in a cost improvement.

In the case of at least two misplaced customers ($m \geq 2$) being reassigned together, the procedure is similar but the destination depot has to have enough spare capacity to serve the set of misplaced customers under consideration.

3.3   Final Routing Phase

Several routing algorithms can be used to produce the final routing once the Exploration phase has finished. In this work the same three algorithms used in the Exploration phase were tested for the Final routing phase.

4   Evaluation of the Proposed Methodology

In this section we provide the results obtained from different numerical experiments of several MPM instantiations. Given the reasonable computational time observed for the MPM methodology, we performed all experiments with all combinations of assignment and routing algorithms for the phases of the methodology (assignment, exploration and final routing phases). The MPM instantiation with the best result obtained is shown in the tables (in the case of equal cost the one with the fastest time is chosen).

The mathematical model for the CMDVRP presented in Sect. 2 was coded in AMPL and solved with CPLEX 12.6.3.0 on a PC Intel Core i7, 16 CPUs, 64 GB of RAM (DDR4) and CentOS 7. The MPM instantiations were coded in Java and executed in a PC with Intel Xeon CPU E3-1220 V2, 4 GB of RAM and Windows 7.

4.1   Comparative Study with Exact Method

Solving the CMDVRP to optimality is extremely costly due to the computational complexity of the problem. Nevertheless, an important aspect of a comprehensive analysis for any proposed heuristic approach is to compare its results against exact methods both regarding objective values and running times.

In   https://www.fing.edu.uy/owncloud/index.php/s/XnvURwxKzQUaH1P it is available the benchmark set of instances used to compare different MPM instantiations against CPLEX. Table 1 presents the results obtained.

The first column of Table 1 provides the identification of the instances, with 20 nodes in total, 2 or 3 depots, and a sequential number. The capacity of each depot is in the range $[66, 125]$, and the vehicle capacity in the range $[50, 70]$. The sum of the customers demand is of 180 units for each instance. We note that

the distribution of the customers and depots is based on real map coordinates on certain islands of the Pacific ocean. Columns 2 to 4 report the name of the MPM instantiation, the costs and the total running times (in seconds) of all phases of the methodology for each one of the instances in the benchmark set. The name of each MPM instantiation is composed by four terms separated by a simple dash: the assignment algorithm (Ur or MUr), the criterion for misplaced customers and the routing algorithms used for the exploration and final phases, respectively. For example, a misplaced criterion 5c1n2m means that 5 closest customers are considered for determining if certain customer is misplaced, at least 1 of them is assigned to another depot, and 2 can be reassigned simultaneously. For all the experiments performed, we consider 1 to 5 closest customers for the misplaced criterion and between 1 or 2 customers to be reassigned simultaneously. The algorithms used in each phase and the misplaced criteria of the MPM instantiations listed in Table 1 were those for which we obtained the best results in the experiments. Columns 5 reports the cost obtained from CPLEX with a running time limited to 3600 s (no significant improvements were noticed with higher running times). Last column 6 in Table 1 provides the percentage gap between the cost of the MPM instantiation and CPLEX, calculated as $100 * (MPM_{Cost} - CPLEX_{Cost})/CPLEX_{Cost}$.

Table 1. Comparison of results for MPM instantiations against CPLEX.

Instance	MPM instantiation	Time	Cost	CPLEX cost	% Gap cost
20n2d01	MUr-5c1n2m-JSprit-JSprit	58.79	3064.9	3085.83	−0.68
20n2d02	Ur-5c1n2m-JSprit-JSprit	24.757	5726.34	5726.34	0.00
20n2d03	Ur-5c1n2m-JSprit-JSprit	22.591	224.18	229.92	−2.50
20n2d04	Ur-1c1n1m-Sweep-C&W	0.001	158.03	158.03	0.00
20n2d05	Ur-5c1n2m-JSprit-JSprit	45.336	354.39	361.26	−1.90
20n2d06	Ur-5c1n2m-Sweep-JSprit	0.266	5808.51	5808.51	0.00
20n2d07	MUr-5c1n2m-JSprit-JSprit	11.337	5873.72	5873.71	0.00
20n2d08	Ur-5c1n2m-JSprit-JSprit	37.826	5062.75	5062.75	0.00
20n2d09	Ur-5c1n2m-Sweep-JSprit	0.298	910.97	924.15	−1.43
20n2d10	Ur-1c1n1m-Sweep-C&W	0.001	292.26	292.26	0.00
20n3d01	Ur-1c1n1m-C&W-JSprit	0.354	2556.36	2726.02	−6.22
20n3d02	Ur-5c1n2m-Sweep-JSprit	0.248	159.84	159.84	0.00
20n3d03	Ur-1c1n1m-C&W-JSprit	0.329	123.98	123.98	0.00
20n3d04	Ur-1c1n1m-C&W-JSprit	0.354	4773.08	4973.73	−4.03
20n3d05	Ur-1c1n1m-C&W-JSprit	0.347	4576.78	4576.78	0.00
Average		13.522			−1.12

From the results in Table 1 we note that MPM outperforms CPLEX in 6 of the 15 instances, and achieves the same objective value in the remaining ones.

Thus, we can conclude that MPM is competitive with CPLEX because the gap error is always negative or zero and the running times of all the considered MPM instantiations are significantly lower than CPLEX (less than 60 s versus 3600 s). We also want to note that the most effective MPM instantiation considering both, costs and running times, is Ur-1c1n1m-C&W-JSprit, since it shows the two lowest percentage cost gaps and less than a half of a second of running time. This MPM instantation makes use of different algorithm approaches for the exploration and final routing phases. This seems to indicate that it would be enough to use a simple and fast algorithm for the exploration phase, and a good and eventually time consuming routing algorithm for the final phase.

4.2   Comparative Study with and Without Exploration Phase

A central part of the proposed methodology, is the intermediate exploration phase for the detection of misplaced customers and the reassignment of them to depots using a routing algorithm. In this section we analyze the impact of including the exploration phase in the methodology by means of a comparative study over ten large instances with different geographical characteristics, available also at the same web repository provided in Sect. 4.1. Some of them are based on instances of the TSPLIB (http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/vrp/), and others were randomly generated, trying to create clusters of customers with different densities.

Tables 2 and 3 report the results obtained for the MPM instantiations MUr-5c1n2m-Sweep-Sweep and Ur-2c1n1m-Sweep-JSprit without and with exploration phase, respectively. Due to the large size of the instances considered, we chose Sweep for the routing of the exploration phase, since it is a simple and fast routing algorithm, although not very efficient. For this reason, it is not the purpose of the experiments presented here to compare the quality of the solutions obtained of these MPM instantiations. The algorithms of the others phases and the misplaced criteria used for the MPM instantiations were those for which we obtained the best results in the experiments performed. Columns 1 to 6 provide the information about the instances: name, number of total nodes, number of depots, total depot capacity, vehicles capacity and total customer demand, respectively. Columns 7 to 10 show the costs and the total running times (in seconds) of all phases of the MPM methodology, without and with exploration phase, respectively. The last two columns report the percentage of gap for the costs and the time ratio (the ratio between the running times observed with and without exploration).

From Tables 2 and 3 we can appreciate that the exploration phase results in a performance improvement that may depend on the routing algorithms used for the exploration and final phases. In the case of the same algorithm (MUr-5c1n2m-Sweep-Sweep), the inclusion of the exploration phase results in a better final solution for all the instances, with an average improvement of 7.18%. Although the running times increased on average 17 times, they can still be considered very good taking into account the size of the instances. In the case of different routing algorithms (Ur-2c1n1m-Sweep-JSprit), we note from Table 3

that in most instances (7 of 10) the inclusion of the exploration phase results in a better final solution, with an improvement in costs from 0.18% to 6.83%. In addition, empirically it seems that the inclusion of the exploration phase does not cause a significant increase in the execution times. Indeed, in almost half of the instances there is a marked decrease in them. This may be due to the fact that the reassignment of customers to depots of the exploration phase simplifies the final routing, i.e., less effort is needed to obtain a good quality routing. Again, it can be seen that it is enough to use a simple and fast algorithm for the exploration phase, and a good and eventually time consuming routing algorithm for the final phase. However, in some cases using different routing algorithms for the two phases can result in a higher cost final solution, as it can be seen in Table 3.

Table 2. MUr-5c1n2m-Sweep-Sweep performance without and with exploration phase.

Inst.	Nodes	Dep.	D.Cap.	V.Cap.	Dem.	Without exp.		With exp.		% Gap	Time
						Cost	Time	Cost	Time	Cost	Ratio
L01	200	5	4250	80	3885	626409.84	0.319	591986.94	1.811	−5.50	5.68
L02	200	5	4250	80	3885	612327.72	0.043	564351.64	1.904	−7.84	44.28
L03	200	8	4400	80	3885	693744.67	0.078	690438.63	0.583	−0.48	7.47
L04	262	13	15920	500	12106	8438.75	0.060	7373.04	1.840	−12.63	30.67
L05	500	6	15000	300	12488	404082.94	0.327	364345.57	4.123	−9.83	12.61
L06	500	6	15000	300	11750	401862.56	0.440	368230.51	15.161	−8.37	34.46
L07	800	7	24500	300	22890	572609.36	0.984	546426.73	11.102	−4.57	11.28
L08	800	7	24500	300	24007	585369.28	1.978	542194.85	22.870	−7.38	11.56
L09	1050	50	50073	500	40801	482949.62	3.818	443298.2	16.027	−8.21	4.20
L10	1050	50	44450	500	40411	452780.95	1.934	420944.92	17.881	−7.03	9.25
Average										−7.18	17.15

Table 3. Ur-2c1n1m-Sweep-JSprit performance without and with exploration phase.

Inst.	Nodes	Dep.	D.Cap.	V.Cap.	Dem.	Without exp.		With exp.		% Gap	Time
						Cost	Time	Cost	Time	Cost	Ratio
L01	200	5	4250	80	3885	481912.19	10.202	487401.81	8.475	1.14	0.83
L02	200	5	4250	80	3885	468610.92	7.409	467768.58	6.902	−0.18	0.93
L03	200	8	4400	80	3885	578998.86	5.098	578998.86	4.814	0.00	0.94
L04	262	13	15920	500	12106	7022.09	23.496	6542.62	26.43	−6.83	1.12
L05	500	6	15000	300	12488	299702.93	60.413	298152.39	60.57	−0.52	1.00
L06	500	6	15000	300	11750	308062.74	56.938	312336.36	50.678	1.39	0.89
L07	800	7	24500	300	22890	466505.25	166.581	463011.41	157.925	−0.75	0.95
L08	800	7	24500	300	24007	469352.3	123.842	467293.69	125.404	−0.44	1.01
L09	1050	50	50073	500	40801	390585.03	19.974	388445.21	21.17	−0.55	1.06
L10	1050	50	44450	500	40411	414786.54	17.403	409329.19	18.793	−1.32	1.08
Average										−0.80	0.98

5   Conclusions

The Capacitated Multi-Depot Vehicle Routing Problem (CMDVRP) is an extension of the MDVRP that considers limited supply on the depots. As far as we know the CMDVRP problem has received much less attention in the literature than other MDVRP extensions. In order to solve this NP-hard problem, we introduce a Multi-Phase Methodology (MPM) that extends the well-known approach of "cluster first, route second". The most relevant feature of MPM is an intermediate exploration phase for detecting and reassigning misplaced customers based on VRP algorithms. As the VRP is a well-known and widely studied problem, the strength of the proposed MPM is to give a straightforward an efficient general framework for the direct use of VRP algorithms, in many cases publicly available and free, to solve the CMDVRP. Each MPM phase offers the possibility of choosing different algorithms depending on the specific characteristics of the problem, the problem instances, hardware limitations, time restrictions, etc. A specific selection of algorithms, for each phase, produces a particular MPM instantiation that yields a heuristic procedure to solve the CMDVRP.

From the results obtained of the numerical experiments carried out, we can conclude that the multi-phase methodology suggested can result in competitive heuristics compared to exact methods. In particular, it may be useful for users who often need to find solutions of quality in a reasonable computational time. We point out that it would be enough to use a simple and fast routing algorithm for the exploration phase, and a good and eventually time consuming routing algorithm for the final phase. We also note that in general the exploration phase produces better solutions without causing a significant increase in the execution times but, in many cases, there is a decrease in them. This may be due to the fact that the reassignment of customers to depots during the exploration phase makes that less effort is needed to obtain a good quality final routing.

The proposed multi-phase methodology enables and facilitates the use of different combinations of algorithms and the possibility to define the criterion for misplaced customers that may include geographical information, supply capacity constraints, time windows and others constraints. Therefore, it has a great potential to be adapted to specific MDVRP variants such as Periodic-VRP (PVRP), MDVRPTW or CMDVRPTW. The exploration phase of MPM allows the introduction of randomness for example in the order in which the misplaced customers are considered to be reassigned or in the reassignment strategy. We believe that the methodology could benefit from employing a randomized strategy in order to explore the solution space more extensively and eventually escape from local optimal solutions.

A possible and interesting direction for future research is to compare the proposed methodology against different solution procedures of the literature for related problems, such as MDVRP (the problem without capacity constraints on the depots). In order to make this comparison, we adapted the instances suggested by [7] and available at http://neumann.hec.ca/chairedistributique/data/mdvrp/. We consider those MDVRP instances of [7] without supply capacities on the depots nor time constraints on the routes duration, but do have

restrictions on the vehicle fleet size. Preliminary results obtained in tests comparing different instances of MPM and the multiphase SFLA-PLEONS algorithm of [13], which as far as we know is one of the faster and more accurate algorithms in the literature for MDVRP, shown that MPM methodology is competitive with fastest running times. The objective is to continue doing more tests varying the instantiations of MPM methodology and also look for other instances of MDVRP in the literature.

Finally, some of the results of the numerical experiment reported, deserve a further analysis. One of them is to analyze the causes of why the addition of the exploration phase does not increase the execution times of the overall methodology, as we empirically observed in the numerical experiments reported in Tables 2 and 3. Another issue is in which cases and why it is sufficient to use a simple and fast algorithm for the exploration phase, and a good and eventually time consuming routing algorithm for the final phase, to obtain good quality solutions.

References

1. Allahyari, S., Salari, M., Vigo, D.: A hybrid metaheuristic algorithm for the multi-depot covering tour vehicle routing problem. Eur. J. Oper. Res. **242**(3), 756–768 (2015)
2. Aurenhammer, F.: Voronoi diagrams - survey of a fundamental geometric data structure. ACM Comput. Surv. **23**(3), 345–405 (1991)
3. Bektas, T., Gouveia, L.: Requiem for the Miller-Tucker-Zemlin subtour elimination constraints? Eur. J. Oper. Res. **236**(3), 820–832 (2014)
4. Calvet, L., Ferrer, A., Gomes, M.I., Juan, A.A., Masip, D.: Combining statistical learning with metaheuristics for the multi-depot vehicle routing problem with market segmentation. Comput. Ind. Eng. **94**, 93–104 (2016)
5. Clarke, G., Wright, J.W.: Scheduling of vehicles from a central depot to a number of delivery points. Oper. Res. **12**(4), 568–581 (1964)
6. Contardo, C., Martinelli, R.: A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. Discret. Optim. **12**, 129–146 (2014)
7. Cordeau, J.F., Laporte, G., Mercier, A.: Improved tabu search algorithm for the handling of route duration constraints in vehicle routing problems with time windows. J. Oper. Res. Soc. **55**(5), 542–546 (2004)
8. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness, 1st edn. W.H. Freeman & Co., New York (1979)
9. Giosa, D., Tansini, L., Viera, O.: Assignment algorithms for the multi-depot vehicle routing problem. In: Proceedings of the 28th Conference of the Sociedad Argentina de Informática e Investigación Operativa (1999)
10. Giosa, D., Tansini, L., Viera, O.: New assignment algorithms for the multi-depot vehicle routing problem. J. Oper. Res. Soc. **53**(9), 977–984 (2002)
11. Laporte, G.: The vehicle routing problem: an overview of exact and approximate algorithms. Eur. J. Oper. Res. **59**(3), 345–358 (1992)
12. Laporte, G.: What you should know about the vehicle routing problem. Nav. Res. Logist. **54**(8), 811–819 (2007)

13. Luo, J., Chen, M.-R.: Improved shuffled frog leaping algorithm and its multi-phase model for multi-depot vehicle routing problem. Expert Syst. Appl. **41**(5), 2535–2545 (2014)
14. Montoya-Torres, J.R., Franco, J.L., Isaza, S.N., Jiménez, H.F., Herazo-Padilla, N.: A literature review on the vehicle routing problem with multiple depots. Comput. Ind. Eng. **79**, 115–129 (2015)
15. Pessoa, A., Sadykov, R., Uchoa, E., Vanderbeck, F.: A generic exact solver for vehicle routing and related problems. Math. Program. **183**, 483–523 (2020)
16. Schrimpf, G., Schneider, J., Stamm-Wilbrandt, H., Dueck, G.: Record breaking optimization results using the ruin and recreate principle. J. Comput. Phys. **159**(2), 139–171 (2000)
17. Tansini, L., Urquhart, M., Viera, O.: Comparing assignment algorithms for the multi-depot VRP. Technical report, INCO, FING, UDELAR (2001)
18. Urquhart, M., Viera, O.: A vehicle routing system supporting milk collections. Opsearch **39**, 46–54 (2002)
19. Vidal, T., Crainic, T.G., Gendreau, M., Lahrichi, N., Rei, W.: A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. Oper. Res. **60**(3), 611–624 (2012)
20. Zhang, B., Li, H., Li, S., Peng, J.: Sustainable multi-depot emergency facilities location-routing problem with uncertain information. Appl. Math. Comput. **333**, 506–520 (2018)
21. Zhang, W., Chen, Z., Zhang, S., Wang, W., Yang, S., Cai, Y.: Composite multi-objective optimization on a new collaborative vehicle routing problem with shared carriers and depots. J. Clean. Prod. **274**, 1–18 (2020)
22. Zhou, L., Baldacci, R., Vigo, D., Wang, X.: A multi-depot two-echelon vehicle routing problem with delivery options arising in the last mile distribution. Eur. J. Oper. Res. **265**(2), 765–778 (2018)