

# Multi-Stage Algorithms for Solving a Generalized Capacitated P-median Location Problem

Mohammed EL AMRANI<sup>1</sup>, Youssef BENADADA<sup>2</sup>

Smart Systems Laboratory, Rabat IT Center  
ENSIAS, Mohammed V University in Rabat  
Rabat, MOROCCO

**Abstract**—The capacitated p-median location problem is one of the famous problems widely discussed in the literature, but its generalization to a multi-capacity case has not. This generalization, called multi-capacitated location problem, is characterized by allowing facilities to use one of several capacity levels. For this purpose, a predefined list of capacity levels supported by all potential facilities is established. In this paper, we will detail the mathematical formulation and propose a new solving method. We try to construct, indeed, a multi-stage heuristic algorithm that will be called BDF (Biggest Demand First). This new method appears in two approaches: Integrated BDF (IBDF) and Hybridized BDF (HBDF) will be improved by using a local search optimization. A valid lower bound to the optimal solution value is obtained by solving a lagrangian relaxation dual of the exact formulation. Computational results are presented at the end using new instances with higher ratio between the number of customers, facilities and capacity levels or adapted from those of p-median drawn from the literature. The obtained results show that the IBDF is much faster with medium quality solution while HBDF is slower but provides very good solutions close to the optimality.

**Keywords**—Location; p-median; multi-capacity; heuristic; LNS; lagrangian relaxation; lower bound

## I. INTRODUCTION

The location of facilities is a major problem for strategic or tactical decisions. It is much encountered in the industry as well as in the real life. Many interesting applications fields were its direct result, such as network design, telecommunications and customer distribution services. The objective is to propose an optimal assignment of customers to potential facilities subject to a number of constraints such as capacity and budget.

The CPMP (Capacitated P-Median location Problem) is a well-known variant that is characterized by the capacity constraints and the number p of medians predefined initially. It is hugely studied in the literature and constitutes several research studies in combinatorial optimization and operations research fields.

Let  $G(N, M, E)$  be a bipartite graph where N represents the set of customers, M is the set of potential sites to install the medians, and E is the set of edges that connect each vertex of N to a node of M. The p-median graph is in the form of a set of clusters, each one is composed of one facility (black triangle) connected to a set of customers (points) or only a closed facility (white triangle) (Fig. 1).

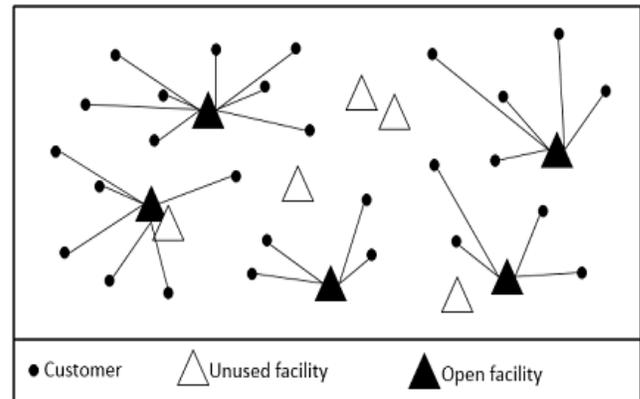


Fig. 1. Graphical representation of basic p-median.

In the industrial fields, service costs generally increase with the capacity used, and as this capacity could exceed in some cases the customers' demands, applying a basic CPMP can present a significant waste of capacity resources. In order to generalize the CPMP for more real problems and for more efficient resolution, we propose in this paper a new variant using different capacity levels. Thus, each facility is prepared to support several capacity levels and to use at the end one level at most. The total of the assigned customers' demands defines the adequate level of capacity, where each level has a corresponding cost. This new variant is called: Budget constraint Multi-Capacitated Location Problem (BMCLP).

The BMCLP's applications can appeared in many industrial areas, such as telecommunications, energy management, and several other fields. The goal of this problem is to minimize the overall cost of assigning customers to facilities. This cost is a multiplication of the unit cost measured by the distance and the demand of the served customer. The total opening costs of facilities is limited by a predefined budget.

The BMCLP is a new variant of location problems family, first time studied by the same authors in [10]. The CPMP problem is NP-complete according to the proof of [11], so its generalization BMCLP is too. Several other variants of CPMP are treated by [18], [3], [1] and [7]. Dynamic location problems are solved by [2], [8] and [6]. Network problems are appeared in [16], [9], [13] and [19]. For the resolution of the problem and its variants, several exact and approximate approaches are tested: the reference [3] applies a cutting plane algorithm based on the Fenchel cuts, references [18], [5] and [4] have chosen to

use the Branch & Price and branch & Bound methods based on Lagrangian relaxation, a resolution with column generation is applied by [4] and references [6], [15] and [12] used different approaches and techniques.

In this work, we will apply the Branch and Cut, a classical resolution method; it is an exact approach that consists of generating cuts at each node of the Branch and Bound tree. Then, we will build a heuristic, more adapted to this location problem variant called BDF (bigger demand first). To improve the solution quality, a local search LNS (Local Neighborhood Search) algorithm, used by [17], will complete the BDF approach.

The BDF, a method in the form of a multi-stage algorithm, is presented in two approaches. Firstly, by using it alone for the solution construction and it will be called IBDF for an Integrated BDF. Secondly by hybridizing it with the application of the branch and cut on a well-defined sub-problem, and it will be called HBDF for a Hybridized BDF.

In order to obtain a valid lower bound to the BMCLP, we use a heuristic procedure to solve the dual problem of our initial formulation. This heuristic procedure combines two different approaches A1 and A2, namely the relaxation of capacity constraints and the decomposition of the problem in two independent sub-problems. Procedure A1 is based on a lagrangian relaxation and sub-gradient optimization, while procedure A2 is based on an independent decomposition starting from the relaxed problem obtained in A1. Indeed, after the relaxation of the capacity constraints and putting them in the objective function, the problem becomes decomposable in two sub-problems, one with variables  $(x_{ij})$  and the second with  $(y_j^k)$ . These variables are defined in the next section below.

This paper is organized as follows. After the introduction, we discuss, in section two, the formulation of the new BMCLP. The third section is devoted to the solving methods, namely the new heuristic approaches, the Lagrange heuristic and the LNS algorithm. The computation of a valid lower bound is detailed in section four. Finally, computational results are presented in the penultimate section before the conclusion.

## II. FORMULATION

The BMCLP is a new variant of capacitated location problem that is characterized by capacity levels, each facility can be used at one level at most. The concept of capacity levels appears in the mathematical formulation with new variables and additional constraints such that each facility must respect the capacity of the level used.

Let  $G(N, M, E)$  be a bipartite graph where  $N$  represents the set of customers,  $M$  is the set of potential sites to install the medians, and  $E$  is the set of edges that connect each vertex of  $N$  to a node of  $M$ .

The BMCLP's graph is in the form of a number of clusters (Fig. 2); each one is composed of one median facility (colorful triangle) connected to a partition of customers set (points). It could also contain only a closed facility (white triangle). Colors represent capacity levels used  $(k_1, k_2, k_3)$  and uncolored triangle represents therefore unused facility.

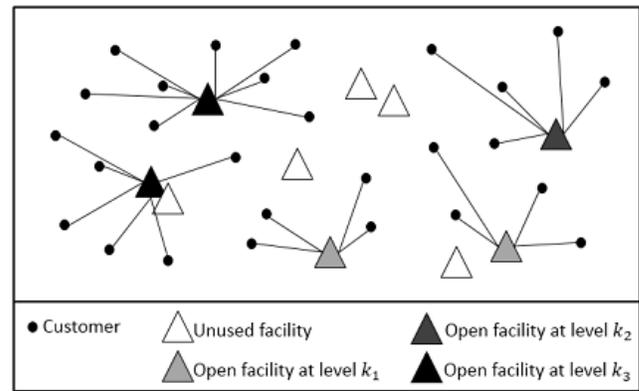


Fig. 2. Graphical representation of BMCLP.

The starting point for building the BMCLP mathematical formulation is the CPMP problem that is defined as follows:

$$\text{Min} \sum_{i \in N} \sum_{j \in M} d_i c_{ij} x_{ij} \quad (1)$$

$$\sum_{j \in M} x_{ij} = 1, \quad i \in N \quad (2)$$

$$\sum_{i \in N} d_i x_{ij} \leq u y_j, \quad j \in M \quad (3)$$

$$x_{ij} \leq y_j, \quad i \in N, j \in M \quad (4)$$

$$\sum_{j \in M} y_j = p \quad (5)$$

$$x_{ij} \in \{0,1\}, \quad i \in N, j \in M \quad (6)$$

$$y_j \in \{0,1\}, \quad j \in M \quad (7)$$

Where  $d_i$  is a demand of customer  $i$ ,  $c_{ij}$  represents the assignment cost of customer  $i$  to facility  $j$ ,  $u$  is the capacity of medians. We assume that all facilities have the same capacity  $u$ .  $p$  is the pre-known number of medians to use from the  $|M|$  available facilities.

$x_{ij}$  is a binary decision variable which equals to 1 if and only if the customer  $i$  is assigned to the facility  $j$ .

$y_j$  is also a binary decision variable that is 1 if and only if the facility  $j$  is open.

The objective function (1) contains only assignment costs, which can be in the form of transportation costs or response time. In this variant, the opening costs of facilities are determined by the number  $p$  in (5). Constraints (2) ensure that each customer is assigned to one and only one median. The constraints (3) require a capacity for each facility. The constraints (4) prohibit the assignment of a customer to a closed facility. The constraints (6) and (7) are the integrality constraints.

The constraints (4) are covered by capacity constraints (3) combined with the integrality ones.

Indeed, let's prove that:

$$\sum_{i \in N} d_i x_{ij} \leq u y_j \Rightarrow x_{ij} \leq y_j \quad \forall i \in N, j \in M$$

Let  $j \in M$  and  $y_j \in \{0,1\}$

If  $y_j = 1$  then

$$x_{ij} \leq y_j, \forall i \in N \quad \text{Because } x_{ij} \in \{0,1\}, \forall i \in N$$

Otherwise  $y_j = 0$  then

$$\sum_{i \in N} d_i x_{ij} \leq 0$$

As  $\forall i \in N, d_i x_{ij} \geq 0$  So  $\forall i \in N, d_i x_{ij} = 0$

Hence  $x_{ij} = 0$

And finally,  $x_{ij} \leq y_j$

**NB.** We keep the redundant constraints in the formulation because it increases the efficiency of the Cplex solver. Indeed, Cplex works with branch and cut methods after the relaxation of integrality constraints, these constraints (4) then constitute valid inequalities that decrease the search area, so the algorithm converges more quickly.

The BMCLP is a generalized p-median problem. This generalization concerns facilities that can be operated at several levels of capacity. To do this we must redefine the variable  $y$  and use other additional data. We will also have a change in the constraints, the constraint requiring a number  $p$  of facilities to open is replaced by a budget constraint limiting the opening costs of factories.

Let  $K$  denotes the set of levels,  $u^k$  the capacity of level  $k$ ,  $f^k$  the opening cost associated with each level  $k$ , and  $B$  the limit budget on the sum of facilities opening costs.

Let  $y_j^k$  be a binary decision variable that is 1 if and only if the facility  $j$  is open and used at the level  $k$ .

The mathematical formulation is as follows:

$$\text{Min} \sum_{i \in N} \sum_{j \in M} d_i c_{ij} x_{ij}$$

$$\sum_{j \in M} x_{ij} = 1, \quad i \in N$$

$$\sum_{i \in N} d_i x_{ij} \leq \sum_{k \in K} u^k y_j^k, \quad j \in M \quad (8)$$

$$x_{ij} \leq \sum_{k \in K} y_j^k, \quad i \in N, j \in M \quad (9)$$

$$\sum_{k \in K} y_j^k \leq 1, \quad j \in M \quad (10)$$

$$\sum_{j \in M} \sum_{k \in K} f^k y_j^k \leq B \quad (11)$$

$$x_{ij} \in \{0,1\}, \quad i \in N, j \in M$$

$$y_j^k \in \{0,1\}, \quad j \in M, k \in K \quad (12)$$

In this formulation, the constraints now take into account the multi-capacity concept. Constraints (9) represent valid inequalities that cut the feasible region. Constraints (10) force

the facility to be opened at one level at most. Constraint (11) is used to limit the opening budget of facilities.

The BMCLP problem allowed to modeling more real situations by opening facilities on several capacity levels. However, this generalization also increases the number of constraints and variables and made its resolution more difficult with a solver such as CPLEX, especially for the big size problems. It is for this reason that we seek in the next section a new heuristic approach more suitable to solve the problem. This new approach, called BDF, is in the form of multi-stage algorithm.

### III. SOLVING METHODS

In this section, we describe a heuristic procedure (BDF) for finding a good feasible solution to BMCLP that is based on the following two ideas.

- The customer with the highest demand has priority in an assignment to the same facility.

- The nearest facility is favored for the assignment of any customer.

The main idea of this method is to assign the customers, iteratively, to the nearest facility while satisfying the capacity constraint, the budget constraint is a priori ignored. When assigning customers, we give priority, as implies the name of the BDF, to customers with greater demands. However, the real factor considered is not the demand value alone but its multiplication at the distance to facility (factors appeared in objective function). As the assignment of customers to the nearest facility can create a cluster with violated capacity constraints, we keep only the first customers with the biggest demands. The customers not assigned to this facility will be re-assigned to the second nearest facility.

In order to be more accurate, we will create a new priority factor for customers against the same facility.

Let  $i$  be a customer,  $j_1$  its nearest first facility and  $j_2$  the second nearest one.

The dissimilarity factor called DF is defined as follows.

$$DF = d_i c_{ij_2} - d_i c_{ij_1}$$

This factor calculates the additional cost when a customer transfers his assignment from a facility to next one. The objective of this method is to minimize the DF factors so as to minimize the overall cost.

The BDF, a method in the form of a multi-stage algorithm, is presented in two approaches:

- By using it alone for the solution construction and it will be called IBDF, for an Integrated BDF.

- By hybridizing it with the application of the branch and cut on a well-defined sub-problem, and it will be called HBDF for a Hybridized BDF.

We use the notation IBDF0 and HBDF0 for IBDF and HBDF respectively without applying the LH and LNS algorithms.

Both approaches provide solutions that are generally not feasible; therefore, a Lagrange heuristic (LH) will be applied for the solution feasibility. The approaches will be improved later by using LNS, (Large neighborhood search) a local optimization method.

Given that the method consists of several sub-methods, we will start by developing them before establishing both algorithms.

#### A. Branch and Cut (B&C)

The Branch and cut is a combination of two algorithms into one, namely the Branch and Bound and cutting planes. We will not be interested in this algorithm because we use it implicitly through the Cplex solver. It will be still used for comparison with the solution obtained by our method for smaller instances.

#### B. Integrated Biggest Demand First (IBDF0)

The IBDF0 is an integrated method that can find for several instances very good solutions and sometimes optimal ones. However, this heuristic method remains unreliable and does not guarantee the feasibility. For this reason, we propose to use Lagrange heuristic to make the solution feasible.

##### IBDF0 Algorithm

- 1) For each customer,
  - Sorting facilities by distances to this customer in ascending order.
- 2)  $p \leftarrow 1$ ;  $UC \leftarrow N$  (UC for unassigned customers set)
- 3) Assigning each customer, from the UC set, to its  $p^{th}$  nearest facility;  $p \leftarrow p+1$
- 4) For each facility,
  - Sorting customers according to theirs DF in descending order.
  - Keeping the maximum customers without exceeding the higher capacity level. The remaining customers will be put in UC set.
  - Updating the capacity resources; the new capacity is the one available after satisfying the customers' demands.
- 5) If  $UC = \emptyset$  Then break
  - Else go to 3-

#### C. Hybridized Biggest Demand First (HBDF0)

The HBDF0 is the first iteration of the IBDF0 method hybridized with the B&C, its principle is to reduce the problem size. Indeed, it proposes assignments for certain customers considered to have important demands. After the application of this method, we set the corresponding variables to the assigned customers and start the execution of the B&C on the unassigned customers' sub-problem.

- 1) For each customer,
  - Sorting facilities by distances to this customer in ascending order.
- 2)  $UC \leftarrow N$  (UC for unassigned customers)
- 3) Assigning each customer to its nearest facility.
- 4) For each facility,

- Sorting customers according to theirs DF in descending order.
- Keeping the maximum customers without exceeding the higher capacity level. The remaining customers will be put in UC set.
- Updating the capacity resources; the new capacity is the one available after satisfying the customers' demands.

5) Updating the budget; the new budget is the one available after subtracting the corresponding costs at all levels used in all open facilities.

6) If  $UC = \emptyset$  then break

Else applying the formulation of the problem with the new updated data (UC, capacities, budget ...), using Branch and Cut.

#### D. Lagrange Heuristic (LH)

The resolution, with violated constraints, gives generally unfeasible solution. To have a feasible solution, we need to apply some existing heuristics. We decided to use the Lagrange heuristic, which consists of setting some variables of the problem, and re-solve the initial problem.

Let's set the variables  $y_j^k$  to their values obtained from the unfeasible solution and reconstruct the problem.

The new formulation after setting  $y_j^k$  is as follows:

$$\begin{aligned}
 & \text{Min} \sum_{i \in N} \sum_{j \in M} d_i c_{ij} x_{ij} \\
 & \sum_{j \in M} x_{ij} = 1, \quad i \in N \\
 & \sum_{i \in N} d_i x_{ij} \leq K_j, \quad j \in M \\
 & x_{ij} \in \{0,1\}, \quad i \in N, j \in M
 \end{aligned} \tag{13}$$

Where  $K_j = \sum_{k \in K} u^k y_j^k$  is a dependent constant on facility j.

The previous problem is linear and contains a reduced number of constraints and variables. It is in the form of knapsack problem with additional demand constraints. Therefore, the problem can be easily solved with Cplex. The solution of this problem is feasible but approximate.

#### E. Large Neighborhood Search (LNS)

To improve the solution obtained, we propose to use one of the local optimization methods. Among those that have demonstrated their effectiveness in combinatorial optimization and particularly in location and transportation problems, we mention LNS (Large Neighborhood Search). This method has the advantage of ensuring optimality for instances of small or medium size when the selected sample is also the problem studied.

LNS is a meta-heuristic used for local optimization. From a first solution, the search algorithm will try to improve it by successive samplings in its neighborhood. At each iteration, one or more clusters (facility and connected customers) are

deconstructed to obtain a sub-problem that is supposed to be easy to solve. Then we solve it using Branch and Cut algorithm to rebuild the solution. A stochastic element defines the sample used for each application of the method.

#### LNS Algorithm

Repeat until the stopping criterion.

1) Randomly selecting a part of the solution (i.e a set of clusters), this part will constitute a sub-problem easy to solve using Cplex.

2) Destroying the clusters of the sub-problem under consideration.

3) Executing Cplex and recovering the obtained partial solution.

4) Integrating the solution of the sub-problem found to the solution of the initial problem.

5) If during 10 iterations, no significant improvement is recorded, we declare the stopping criterion.

#### BDF Algorithms

After having defined all steps of the BDF method, we can then implement the following two algorithms:

##### IBDF:

1) *IBDF0*

2) *If the obtained solution is feasible*

- LSN
- Exit

3) *Else*

- LH
- LNS
- Exit

##### HBDF:

1) *HBDF0*

2) *If the obtained solution is feasible*

- LSN
- Exit

3) *Else*

- LH
- LNS
- Exit

#### IV. COMPUTATION OF THE LOWER BOUND

In this section, we will present a method based on lagrangian relaxation to determine a good lower bound. The calculation of this will allow us to evaluate the quality of the solution obtained. It can be noted that the formulation contains three different constraints' blocks; the first one with variables  $x_{ij}$  (constraints 2), the second with variables  $y_j^k$  (constraints 10 and 11) and the third which contains a combination of both (constraints 8 and 9). Eliminating the third block, the formulation becomes decomposable into two independent sub-

problems, one with variable  $x_{ij}$  and the second with variables  $y_j^k$ .

As we have mentioned above, constraints (9) are facultative and its violation has no impact on solution feasibility. In order to get the latter decomposition, we will forget constraints (9) and relax constraints (8).

The new relaxed problem is as follows:

$$\begin{aligned} \text{Min} \quad & \sum_{i \in N} \sum_{j \in M} d_i(c_{ij} - \lambda_j)x_{ij} + \sum_{j \in M} \sum_{k \in K} \lambda_j u^k y_j^k \quad (14) \\ & \sum_{j \in M} x_{ij} = 1, \quad i \in N \\ & \sum_{k \in K} y_j^k \leq 1, \quad j \in M \\ & \sum_{j \in M} \sum_{k \in K} f^k y_j^k \leq B \\ & x_{ij} \in \{0,1\}, \quad i \in N, j \in M \\ & y_j^k \in \{0,1\}, \quad j \in M, k \in K \end{aligned}$$

This problem is decomposable into two sub-problems:

Problem x:

$$\begin{aligned} \text{Min} \quad & \sum_{i \in N} \sum_{j \in M} d_i(c_{ij} - \lambda_j)x_{ij} \quad (15) \\ & \sum_{j \in M} x_{ij} = 1, \quad i \in N \\ & x_{ij} \in \{0,1\}, \quad i \in N, j \in M \end{aligned}$$

Problem y:

$$\begin{aligned} \text{Min} \quad & \sum_{j \in M} \sum_{k \in K} \lambda_j u^k y_j^k \quad (16) \\ & \sum_{k \in K} y_j^k \leq 1, \quad j \in M \\ & \sum_{j \in M} \sum_{k \in K} f^k y_j^k \leq B \\ & y_j^k \in \{0,1\}, \quad j \in M, k \in K \end{aligned}$$

These two problems are independent and can be solved in parallel using sub-gradient algorithm. However, we propose that at each iteration, we start with the problem x, we obtain the variables' values x which give information on the facilities that must be open then we add it as constraint in problem y before solving it. At each iteration of the sub-gradient algorithm, we can have a solution that present a lower bound for our problem; this bound will be improved from one iteration to another.

#### V. COMPUTATIONAL RESULTS

The BMCLP is a new problem that has not been found in the literature. Therefore, we cannot find instances for the test or for comparison. For this reason, we decide within this research to create instances using semi-random values based on justified choices and whose difficulty is measurable. We will also use p-median instances adapted to our problem to complete the calculation tests.

We turn both algorithms on an i7-2600 CPU @ 3.40 GHz machine with 8GB RAM. We use the programming language java version 7 and version 12.3 of Cplex.

The test set consists of five classes of instances representing five levels of difficulty (easy, medium, difficult, very difficult and complex). The difficulty of these instances is based on the size of the problem, which is generally measured by the number of customers, facilities, and capacity levels. However these two last numbers have a small impact on the problem's size. Each level of difficulty contains several test instances. Other difficulty factors are taken into account, namely the dispersion of customers against facilities and the available resources. Experience shows that the difficulty of the problem varies in proportion to the variance of customers' distances and their demands. At the same time, it varies inversely with the budget allocated for opening facilities and their capacity levels. Thus, by increasing the difficulty, while

keeping the feasibility, we multiply the number of iterations necessary to find the optimal solution.

These are the parameters used in the following result's table:

- LD: Level of Difficulty
- NC: Number of Customers
- NF: Number of Facilities
- NL: Number of Levels
- LB: Lower Bound
- Obj: Objective function value
- CPU: Execution time
- GAB: = Min(GAB1,GAB2)  
 $GAB1 = (BDF - B\&C) / BDF$   
 $GAB2 = (BDF - LB) / BDF$

The following table shows the different instances used and the execution results of the IBDH and HBDF methods as well as the lower bound (Table I):

TABLE I. COMPUTATIONAL RESULTS TABLE

| Insta<br>nce | LD             | NC   | NF   | NL | Branch & Cut |        | LB     | IBDF   |        |        | HBDF   |          |       |
|--------------|----------------|------|------|----|--------------|--------|--------|--------|--------|--------|--------|----------|-------|
|              |                |      |      |    | Obj          | CPU(s) |        | Obj    | CPU(s) | GAP    | Obj    | CPU(s)   | GAP   |
| <b>F1</b>    | Easy           | 10   | 3    | 2  | 184          | 0.02   | 183    | 184    | 0.001  | 0,00%  | 184    | 1.521    | 0.00% |
| <b>F2</b>    |                | 10   | 5    | 2  | 230          | 1.57   | 228    | 230    | 0.001  | 0,00%  | 230    | 2.932    | 0.00% |
| <b>F3</b>    |                | 20   | 5    | 3  | 430          | 3.88   | 426    | 430    | 0.006  | 0,00%  | 430    | 5.875    | 0.00% |
| <b>F4</b>    |                | 30   | 8    | 3  | 372          | 8.75   | 361    | 386    | 0.011  | 3,63%  | 372    | 9.104    | 0.00% |
| <b>M1</b>    | Medium         | 50   | 4    | 3  | 2120         | 159.45 | 2073   | 2423   | 1.013  | 12,51% | 2158   | 12.187   | 1.76% |
| <b>M2</b>    |                | 50   | 6    | 4  | 8429         | 132.31 | 8269   | 8697   | 5.012  | 3,08%  | 8501   | 16.345   | 0.85% |
| <b>M3</b>    |                | 70   | 6    | 4  | 5915         | 234.71 | 5758   | 7079   | 4.004  | 16,44% | 6002   | 29.297   | 1.45% |
| <b>D1</b>    | Difficult      | 100  | 10   | 5  | 12602        | 5991.1 | 12494  | 13923  | 11.105 | 9,49%  | 12802  | 136.548  | 1.56% |
| <b>D2</b>    |                | 100  | 15   | 5  | -            | -      | 14138  | 15572  | 12.435 | 9,21%  | 14435  | 24.364   | 2.06% |
| <b>D3</b>    |                | 200  | 15   | 8  | -            | -      | 835569 | 863935 | 13.432 | 3,28%  | 854112 | 51.784   | 2.17% |
| <b>V1</b>    | Very Difficult | 300  | 25   | 10 | -            | -      | 318418 | 345808 | 12.726 | 7,92%  | 319853 | 36.273   | 0.45% |
| <b>V2</b>    |                | 300  | 30   | 10 | -            | -      | 194222 | 227616 | 15.762 | 14,67% | 202615 | 77.238   | 4.14% |
| <b>V3</b>    |                | 402  | 30   | 12 | -            | -      | 369834 | 390885 | 21.253 | 5,39%  | 387483 | 83.684   | 4.55% |
| <b>V4</b>    |                | 402  | 40   | 12 | -            | -      | 405420 | 432915 | 23.932 | 6,35%  | 406778 | 91.105   | 0.33% |
| <b>C1</b>    | Complex        | 500  | 50   | 4  | -            | -      | 259809 | 299221 | 37.317 | 13,17% | 261644 | 117.634  | 0.70% |
| <b>C2</b>    |                | 1000 | 100  | 4  | -            | -      | 430119 | 471794 | 51.265 | 8,83%  | 451432 | 136.721  | 4.72% |
| <b>C3</b>    |                | 3038 | 600  | 10 | -            | -      | 94725  | 105740 | 56.216 | 10,42% | 99801  | 294.364  | 5.09% |
| <b>C4</b>    |                | 3038 | 700  | 10 | -            | -      | 85082  | 116380 | 54.823 | 26,89% | 89801  | 5011.784 | 5.25% |
| <b>C5</b>    |                | 3038 | 1000 | 10 | -            | -      | 297208 | 347564 | 47.823 | 14,49% | 298053 | 3456.273 | 0.28% |

From the numerical results, we note that the BDF method has yielded good results in most cases and across both approaches. However, we find that IBDF is much faster but less effective, while HBDF gives very good results. Although it is slower, it still works for a reasonable time. The local search LNS, improving the quality of the solution, allows in some cases and for small instances to reach the optimal solution. This is justified by the fact that the selected sub-problem to destroy coincides with global problem.

## VI. CONCLUSION

In this paper we introduced the budget constraint multi-capacity location problem. We proposed the BDF method in the integrated BDF and Hybridized BDF approaches, Lagrange heuristic to ensure the solution feasibility and for the improvement step, the local LNS search. To evaluate the solution quality, a valid lower bound to the optimal solution value is obtained by solving a lagrangian relaxation dual problem.

The solution achieved might not be an optimal BMCLP solution, however the branch and cut, used for small instances, allows to estimate its maximum distance from optimality. For large instances, a valid lower bound is calculated. Computational tests on problems adapted from those proposed in the literature and on new test problems with large dimensions show the effectiveness of the proposed multi-stage algorithm. As a perspective we propose to test other solving methods that ensure the optimality and work on much large scale problems.

## REFERENCES

- [1] Baldacci R., Hadjiconstantinou E., Vittorio M. & Mingozzi A., A new method for solving capacitated location problems based on a set partitioning approach, *Computers & operations research*, 29, 365-386 (2001)
- [2] Behmardi B. & Shiwoo L., Dynamic multi-commodity capacitated facility location problem in supply chain. 2008 Industrial engineering research conference, 1914-1919 (2008)
- [3] Boccia M., Sforza A., Sterle C. & Vasilyev I., A cut and branch approach for the capacitated p-median problem based on Fenchel cutting planes, *Journal of mathematical modelling and algorithms*, 7, 43-58 (2008)
- [4] Ceselli A., Two exact algorithms for capacitated p-median problem, *Springer-verlag berlin heidelberg*, 1, 319-340 (2003)
- [5] Ceselli A. & Giovanni R., A branch-and-price algorithm for the capacitated p-median problem, *Networks an international journal*, 45, 125-142 (2005)
- [6] Da Gama F. S. & Captivo M. E., A heuristic approach for the discrete dynamic location problem, *Location Science*, 6, 211-223 (1998)
- [7] Dantrakul S., Likasiri C. & Pongvuthithum R., Applied p-median and p-center algorithms for facility location problems, *Expert Systems with Applications*, 41, 3596-3604 (2014)
- [8] Dias J., Captivo M. E. & Climaco J., Capacitated dynamic location problems with opening, closure and reopening of facilities, *IMA journal of management mathematics*, 17,317-348 (2006)
- [9] Eberyab J., Krishnamoorthya M., Ernsta A. & Bolandb N, The capacitated multiple allocation hub location problem: Formulations and algorithms, *European journal of operational research*, 120, 614-631 (2000)
- [10] El Amrani M., Benadada Y. & Gendron B., Generalization of capacitated p-median location problem: modeling and resolution, *International IEEE Conference of logistics operations management*, 1-6 (2016)
- [11] Garey M. R. & Johnson D. S., *Computers and intractability, a guide to the theory of NP-completeness*, 338, The ACM digital library, Philadelphia (1979)
- [12] Klose A., A lagrangean relax-and-cut approach for the two-stage capacitated facility location problem, *European journal of operational research*, 126, 408-421 (2000)
- [13] Ling-Chieh K. & Watson-GANDY, C. D. T., An approximation algorithm for a competitive facility location problem with network effects, *European Journal of Operational Research*, 267, 176-186 (2018)
- [14] Lorena L. A. N. & Senne E. L. F., A column generation approach to capacitated p-median problems, 31, 863-876 (2003) <http://www.lac.inpe.br/lorena/instancias.html>
- [15] Mohammad S. K., Haldun S. & Cem I., A column generation approach for the location-routing problem with time windows, *Computers & Operations Research*, 90, 249-263 (2018)
- [16] Oded B., Jorg K. & Dmitry K., On covering location problems on networks with edge demand, *Computers & Operations Research*, 74, 214-227 (2016)
- [17] Ropke S., Pisinger D. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. (2004). Technical report, Department of Computer Science, University of Copenhagen, Denmark
- [18] Senne E. L. F. & Pereira M. A., A branch-and-price approach to p-median location problems, *Computers & operations research*, 32, 1655-1664 (2004)
- [19] Shu J., Ma Q. & Li S., Integrated location and two-echelon inventory network design under uncertainty, *Annals of operations research*, 181, 233-247 (2010)