Regular Issue

A Greedy Randomized Adaptive Search With Probabilistic Learning for solving the Uncapacitated Plant Cycle Location Problem

Israel López-Plata[1]*, Christopher Expósito-Izquierdo[1], Eduardo Lalla-Ruiz[2], Belén Melián-Batista[1], J. Marcos Moreno-Vega[1]

[1] Universidad de La Laguna, La Laguna (Spain)
[2] University of Twente, Enschede (Netherlands)

Received 17 December 2020 | Accepted 18 December 2021 | Published 25 April 2022

Abstract

In this paper, we address the Uncapacitated Plant Cycle Location Problem. It is a location-routing problem aimed at determining a subset of locations to set up plants dedicated to serving customers. We propose a mathematical formulation to model the problem. The high computational burden required by the formulation when tackling large scenarios encourages us to develop a Greedy Randomized Adaptive Search Procedure with Probabilistic Learning Model. Its rationale is to divide the problem into two interconnected sub-problems. The computational results indicate the high performance of our proposal in terms of the quality of reported solutions and computational time. Specifically, we have overcome the best approach from the literature on a wide range of scenarios.

Keywords

Greedy Randomized Adaptive Search Procedure, Probabilistic Learning Model, Uncapacitated Plant Cycle Location Problem.

DOI: 10.9781/ijimai.2022.04.003

I. Introduction

Location-routing problems are a family of hard combinatorial optimization problems found in the field of distribution network design. The objective is to open a subset of depots in potential locations with the aim of fulfilling the demand of customers by means of a fleet of vehicles. Traditionally, these optimization problems have been tackled separately. However, the evolution of computers, the emergence of new optimization techniques, and the necessity of holistic solutions for new problem applications have aroused a renewed interest in their joint solution.

Nowadays, in city and last-mile logistics, freight transportation stakeholders and service providers have to regularly redesign and improve their logistics processes to satisfy customers requirements while reducing infrastructure and transportation costs. The design of transportation networks in the context of less-thana- truck deliveries leads to the definition of the Uncapacitated Plant Cycle Location Problem (UPCLP).

The UPCLP is an NP-Hard optimization problem whose main goal is to select a subset of locations from a bigger set of potential locations where establish plants to serve a determined set of customers. The number of plants is unknown in advance, but it is important to remark that due to the operations required to create a plant, both the set up and the assignment of a customer to it have a specific cost. Each plant has one vehicle to serve all its customers following a determined route that also have an associated cost. A solution for the UPCLP solves two different subproblems:

- Obtaining the set of locations where open the plants to serve all customers, minimizing the cost to open the plants and assigning every customer to a determined plant.
- Determining the routes followed by the vehicles to serve its assigned customers with the less possible cost.

In the strong sense, this problem combines two well-known optimization problems: the Uncapacitated Facility Location Problem [1] and the Multi-Depot Travelling Salesman Problem [2]. Section III explains the UPCLP in detail.

The applications for the UPCLP are those related to the location of plants where the service or freights distributed to customers are not affected by plant or vehicle capacity constraints. Related applications can be found in humanitarian logistics [3], telecommunications [4], [5], distribution system design [6], postal delivery [7], [8], among others.

The main goals of the present paper are described as follows:

- Proposing an optimization model for the UPCLP.
- Developing a metaheuristic approach based on the paradigm of the Greedy Randomized Adaptive Search Procedure (GRASP) that incorporates a probability distribution for selecting locations within the UPCLP. Its goal is to obtain faster solutions than the optimization model, and on the other hand, provide feasible solutions for larger scenarios that may appear in practical cases and where the optimization model is unable to provide a solution.

* Corresponding author.

E-mail address: ilopezpl@ull.edu.es

- Assessing the performance of the GRASP in comparison with the best approach so far and our optimization model on problem instances from the literature. Additionally, with the aim of evaluating our metaheuristic on structured scenarios, a set of instances adapted from TSPLIB [9] to simulate largescale scenarios are also tackled.

The remainder of this work is organized as follows. Section II reviews the literature related to the UPCLP. Section III explains the detail of the UPCLP. Afterwards, Section IV presents an optimization model of the UPCLP. Section V describes a GRASP with Probabilistic Learning Model to solve the UPCLP from an approximate standpoint. Section VI discusses the applicability of the optimization model and a metaheuristic approach to realistic scenarios and checks their performances in comparison with a previous approach. Finally, Section VII extracts the main conclusions from the work and suggests several lines for further research.

II. Literature Review

The interest in location-routing problems by the scientific community has increased over the last years. One of the seminal papers in the field can be attributed to Watson-Gandy and Dohrn in 1973, as indicated in [10]. The main reason for the joint treatment of decisions concerning the location of plants and vehicle routing is found in that solving them independently gives rise to only suboptimal results in most cases. The suboptimality of the results has been demonstrated in a multitude of works published in the literature. This is the case of the two-phase tabu search proposed in [11].

Evidence of the increasing interest is the fact that numerous surveys dedicated to the analysis and classification of works related to location-routing problems have been published so far. Some outstanding examples are the papers [12], [10], [13], [14], and [15]. New variants and applications of location-routing problems are also discussed in [16].

The interest in this kind of problem has arisen from several practical fields. This is the case of waste management. The work [17] aims to identify the best place to open treatment centers and how to route the hazardous waste derived from industrial activity to disposal centers in a region of Turkey. In this case, the optimization goal is to minimize the transportation risk and the total cost associated with the fixed annual cost of opening a treatment technology and a disposal facility. The authors propose a mixed-integer programming model that incorporates constraints to handle mass balance or the minimum amount of waste required for technology. Furthermore, [18] introduces an improved metaheuristic with a specially-designed directed local search procedure to solve a general two-echelon multi-objective location routing problem in waste collection. In particular, two-echelon transportations must first collect waste generated in demand nodes to collection centers. Then, the waste must be transported while considering flow constraints and capacity constraints. [19] uses a K-Means clustering combined with an Ant Colony Optimization (ACO) to find the shortest routes between 2 nodes in a network of IoT devices optimizing the Quality of Service of the network. This paper divides the whole network into clusters depending of the types of subnetworks, which optimizes the routes creation. Lastly, [20] focuses on the capacitated location routing problem, where each depot has a fixed opening cost and a known capacity to satisfy the customers' demand. Also, the vehicles have capacities and travelling costs. Thus, the optimization criterion is to minimize the total cost, composed of depot, travelling, and vehicle costs. The problem is solved by means of a a hybrid genetic algorithm that explores unfeasible solutions and presents a high competitive performance in comparison with other approaches found in the literature in terms of solution quality and time efficiency.

Other practical applications of location-routing problems are telecommunication network design, electric vehicle transportation, good distribution or airline topologies, among others. For example, [21] addresses the problem of designing synchronous digital hierarchy rings in the context of mobile communications access networks. The problem consists in finding the number and type of the base station controllers to locate at each potential site and, on the other hand, in defining synchronous digital hierarchy rings such that each base transceiver station is in exactly one ring. The authors propose a mixed integer programming model and a heuristic method to solve the problem in realworld instances. [22] seeks to find the number and location of electric vehicle battery swap stations with an optimal route plan based on stochastic customer demands. The problem is solved by means of a hybrid variable neighbourhood search algorithm that combines a binary particle swarm optimization. [23] presents a transportation location routing problem in which the goal is to satisfy the demand of clients from a set of plants with maximum capacity and through intermediate eligible points called city distribution centers, which are sites dedicated to receive products from the plants and deliver them to the clients. The objectives are minimizing the total operation cost of the system and maintain balance in the vehicle operator's workload. Lastly, [24] includes a set of hubs to improve the routes followed by planes on iranian airspace, using a Multi-objective Genetic Algorithm to set the best places to locate these hubs.

Due to their performance, metaheuristics have become attractive alternatives to address location routing problems. Representative examples of these techniques have been proposed so far. Some of them are variable neighborhood search [25], multiple ant colony optimization algorithm [26], Simulated Annealing [27], Particle Swarm Optimization [28], hybrid PSO with Path Relinking [29], Tabu Search [30], GRASP with Path Relinking [31], and clustering analysis [32].

In spite of the existence of a wide corpus of papers in the literature about location-routing problems, the works briefly described in the following are of special interest in this paper. [33] introduced the Uncapacitated Plant Cycle Location Problem and proposed a preliminary version of the technique presented in the paper at hand. [34] presents a strategic problem that can be seen as a generalization of the location-routing problem in which the Capacitated Facility Location Problem and the Multi-Depot Vehicle Routing Problem are combined. The mentioned problem considers costs derived from vehicle usage, vehicle and location capacities, and customer demands. Similar multiroute capacitated approaches have been recently considered in [35], [36], and [37]. Moreover, [38] presents a multiobjective application of location-routing problems to home-to-work bus service. On the other hand, [39] proposes a MIP model and a Branch-and-Cut algorithm to solve several two-level network design problems. Furthermore, [40] and [41] address the PCLP with maximum service capacity constraints associated with the plants to set up. In the first paper, the authors propose a Branch and Cut, even guarantees the optimality of the reported solutions, it requires extremely large computational times (more than 1 hour) in a multitude of cases. In the latter paper, a tabu search is proposed in which an initial solution is obtained from an optimization model. Lastly, [42] proposes a metaheuristic approach based upon the Honey Bees Mating Optimization algorithm for solving the UPCLP. The computational results indicate the algorithm provides highquality solutions in reasonable computational times.

III. Uncapacitated Plant Cycle Location Problem

The Uncapacitated Plant Cycle Location Problem (UPCLP) is a deterministic optimization problem that seeks to select a subset of locations to set up plants with the aim of serving customers geographically distributed on a two-dimensional scenario.

Input data of the UPCLP is a set of m potential discrete locations (*e.g.*, places with the required technical equipment, safety places, etc.), denoted as M, in which to place plants (*e.g.*, industrial infrastructures, hubs, health-care services, warehouses, cross-docking centers, etc.) to serve a well-known set of n customers, denoted as N. Each available location, $j \in M$, could have at most one plant. This way, the set of plants is denoted as $P \subseteq M$. The number of plants set up at the available locations is $k \leq m$, but unknown in advance. That is, $|P| = k$. In this regard, setting up a plant at location $j \in M$ incurs a fixed cost, denoted as $o_j \geq 0$, which indicates, according to the application field, the opening cost, time required to establish a medical camp, etc.

In the UPCLP, each plant can serve an unlimited number of customers whereas each customer must be served directly by exactly one of the plants (*i.e.*, single-echelon approach). However, assigning a customer $i \in N$ to a plant at location $j \in M$ gives rise to a fixed cost, denoted as $c_{ij} \geq 0$, which indicates the cost of providing service to the customer. Without loss of generality, it is assumed that a given customer can be assigned to a plant at any location. The plant where a customer $i \in N$ is assigned to is denoted as $\sigma(i) \in P$. The set of customers served from a given location $j \in M$ is denoted as N_j, where $N = \cup_{j \in M} N_j$ and $N_j \cap N_{j'} = \emptyset, \forall j, j' \in M$. It should be noted that $N_j = H$ whenever no plant is set up at location $j \in M$. Furthermore, the set of customers assigned to a given plant must be served following a delivery route. In this regard, the travel cost between two customers or locations, $i, j \in N \cup M$, is symmetric and denoted as $d_{ij} > 0$, where $d_{ij} = d_{ji}$. All the travel costs satisfy the triangle inequality [43].

The previous description of the UPCLP indicates the following decisions have to be made: (i) selecting a subset of locations in which to set up plants, (ii) determining which each plant serves a subset of customers, and (iii) building vehicle routes to serve the customers (*i.e.*, the sequence in which those customers associated with each plant are going to be served).

Fig. 1. illustrates an example of the UPCLP composed of $m = 5$ locations and $n = 25$ customers. In this case, $k = 2$ plants have been set up. One of the plants serves customers 1, 10, 11, 19, 7, 8, 18, 5, 17, 14, 15, 2, 13, and 6, whereas the other plant serves the remaining customers.

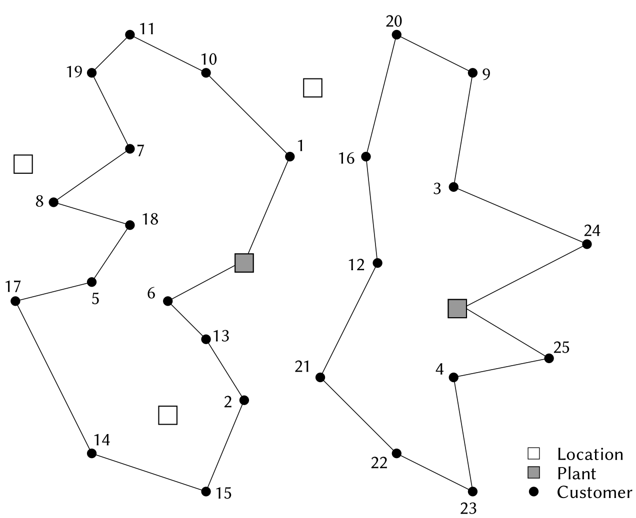


Fig. 1. Example of the Uncapacitated Plant Cycle Location Problem composed of $m = 5$ locations and $n = 25$ customers.

Finally, it worth mentioning that, whenever the UPCLP consists of only one location in which to place plants to serve the customers (*i.e.*, $m = k = 1$) and all the assignment costs are equal (*i.e.*, $c_{ij} = r$; $\forall i \in N$; $j \in M$, where r is a non-negative value), the UPCLP reduces to the Travelling Salesman Problem [44]. Consequently, in the strong sense, the UPCLP belongs to the NP-hard class of optimization problem.

IV. Optimization Model

In this section, we present a Mixed-Integer Linear Programming (MILP) model aimed at solving the Uncapacitated Plant Cycle Location Problem (UPCLP). With this goal in mind, in the following we firstly introduce the families of variables used in the model:

u_i	Integer variable associated with customer $i \in N$
x_{ij}	1 if the edge (i, j) is included in the solution, where $i, j \in M \cup N$. 0, otherwise
y_j	1 if a plant is set up at location $j \in M$. 0, otherwise
z_{ij}	1 if customer $i \in N$ is assigned to a plant set up at location $j \in M$. 0, otherwise

The objective function of the MILP is to minimize the costs derived from (i) setting up plants, (ii) assigning customers to the plants, and (iii) routing the customers, as shown in (1):

$$\min \sum_{j \in M} o_j \cdot y_j + \sum_{i \in N} \sum_{j \in M} c_{ij} \cdot z_{ij} + \sum_{i \in M \cup N} \sum_{j \in M \cup N} d_{ij} \cdot x_{ij} \quad (1)$$

Each customer is assigned to exactly one plant, as seen in (2):

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

Each plant can serve an unlimited number of customers, as shown in (3):

$$\sum_{i \in N} z_{ij} \leq n_{limit} \cdot y_j, \forall j \in M \quad (3)$$

where parameter n_{limit} has to be equal to or larger than n for modeling the uncapacitated version of this problem with regards to facilities. For switching to the capacitated version, then n_{limit} has to be less than n (leading to the PCLP). Finally, note that in our case, this constraint is redundant and can be omitted.

Degree constraints aimed at ensuring that each customer has previous (4) and next (5) nodes in its route:

$$\sum_{\substack{i' \in M \cup N \\ i' \neq i}} x_{ii'} = 1, \forall i \in N \quad (4)$$

$$\sum_{\substack{i' \in M \cup N \\ i' \neq i}} x_{i'i} = 1, \forall i \in N \quad (5)$$

Degree constraints aimed at ensuring that each location has previous (6) and next (7) nodes only if a plant has been set up:

$$\sum_{i \in N} x_{ij} = y_j, \forall j \in M \quad (6)$$

$$\sum_{i \in N} x_{ji} = y_j, \forall j \in M \quad (7)$$

Subtour elimination constraints in which all the customers can be served along the same route can be seen on equations (8) and (9):

$$u_i - u_{i'} + n \cdot x_{i'i} \leq n - 1, \forall i, i' \in N, i \neq i' \quad (8)$$

$$1 \leq u_i \leq n, \forall i \in N \quad (9)$$

The edge $(i; j)$ can be used if and only if customer $i \in N$ is assigned to a plant set up at location $j \in M$, as seen in (10):

$$x_{ij} \leq z_{ij}, \forall i \in N, \forall j \in M \quad (10)$$

Constraint (11) restricts if customers $i, i' \in N$ are assigned to plants set up at different locations, $j, j' \in M$, then they cannot be in the same route:

$$x_{i'i} + z_{ij} + z_{i'j'} \leq 2, \forall i, i' \in N, i \neq i', \forall j, j' \in M, j \neq j' \quad (11)$$

Finally, the domain of the decision variables is defined on equations (12), (13) and (14):

$$x_{ij} \in \{0,1\} \quad \forall i, j \in M \cup N \quad (12)$$

$$y_j \in \{0,1\} \quad \forall j \in M \quad (13)$$

$$z_{ij} \in \{0,1\} \quad \forall i \in N, j \in M \quad (14)$$

V. Greedy Randomized Adaptive Search Procedure With Probabilistic Learning Model

A Greedy Randomized Adaptive Search Procedure [45] with Probabilistic Learning Model (GRASP-PLM) is here presented to solve the Uncapacitated Plant Cycle Location Problem (UPCLP). In general terms, a GRASP is an iterative metaheuristic based upon two main components: a constructive phase aimed at building feasible solutions and an intensification phase dedicated to improving the quality of the found solutions. The high performance of the GRASP when tackling a wide range of heterogeneous combinatorial problems from the literature encourages us to consider it as a promising candidate to solve the UPCLP. The GRASP-PLM includes a joint probability distribution that allows selecting a subset of locations to set up plants.

The rationale behind our GRASP-PLM is to split the UPCLP into the following two interconnected sub-problems to be solved consecutively:

1. *High-level Problem* (*HP*). Determining the subset of locations in which to set up plants to serve the customers (*i.e.*, $P \subseteq M$).
2. *Low-level Problem* (*LP*). Given the plants, assigning the customers to the plants and determining the delivery routes to serve them.

The pseudocode of our GRASP-PLM is depicted in Algorithm 1. The first step is to obtain a solution of the HP, s_{HP} , by sampling the PLM (line 3). This process is described in Section A. Once the plants have been set up, the assignments and routes of the customers are determined by means of a constructive phase (line 4). This phase gives rise to a feasible solution of the UPCLP, denoted as s. A local optimum, s_{local}, is achieve from s. The PLM is updated in those cases in which the best solution found by the search is improved (lines 6-9). Lastly, the search is finished when a certain stop criterion is met (lines 2-10).

Algorithm 1. Pseudocode of the GRASP-PLM for the Uncapacitated Plant Cycle Location Problem

1: $s_{best} \rightarrow \emptyset$
2: **while** (stop criterion is not met) **do**
3: $s_{HP} \leftarrow$ Get high-level solution from the probabilistic learning model
4: $s \leftarrow$ Assign customers and determine routes associated with the plants in s_{HP}
5: $s_{local} \leftarrow$ Apply local search to s
6: **if** ($f(s_{local})$ $f(s_{best})$) **then**
7: $s_{best} \leftarrow s_{local}$
8: Update probabilistic learning model with s_{best}
9: **end if**
10: **end while**
11: Return s_{best}

A. High-Level Problem

The High-level Problem (HP) seeks to determine a subset of locations to set up plants aimed at serving the customers. The main decisions to make at this point are to (i) determine the number of those plants to set up and (ii) select a non-empty set of locations to set up plants. With these goals in mind, we propose to use a Probabilistic Learning Model (PLM). It is composed of the following vectors of probabilities:

1. v_1. It is a vector of m elements, where $v_1(i)$ is the probability of opening i plants. This probability is formally defined as follows:

$$v_1(i) = \frac{f_1(i)}{\sum_{j=1}^{n} f_1(j)}, \forall i = 1,2, \ldots, m \quad (15)$$

where $f_1(i)$ is the number of times i plants have been open in a previous high-quality solution found during the search. Initially, $f_1(i) = 1, \forall i = 1, 2, \ldots, m$.

2. v_2. It is a vector of m elements, where $v_2(i)$ is the probability of setting up a plant at location i. This probability is formally defined as follows:

$$v_2(i) = \frac{f_2(i)}{\sum_{j=1}^{n} f_2(j)}, \forall i = 1,2, \ldots, m \quad (16)$$

where $f_2(i)$ is the number of times a plant has been set up at location $i \in M$ in a previous high-quality solution found during the search. Initially, $f_2(i) = 1, \forall i = 1, 2, \ldots, m$.

A two-step process is carried out to sample solutions from the PLM. Firstly, a random probability is generated, denoted as $p_1 \in [0 \ldots 1]$. This probability allows to determine the number of plants to set up, $1 \leq k \leq m$, as follows:

$$k = \arg max_{i=1,2,\ldots,m} \left\{ \sum_{j=1}^{i} v_1(j) \mid \sum_{j=1}^{i} v_1(j) \leq p_1 \right\} \quad (17)$$

Finally, once the number of plants is known, a set P composed of k locations must be defined according to the probabilities in v_2. The pseudocode of this process is depicted in Algorithm 2. At each step, a probability $p_2 \in [0 \ldots 1]$ is generated. The location with the maximum cumulative probability no greater than p_2 is selected. The process finishes when k different locations have been selected. It should be noted that P constitutes a solution of the HP.

Algorithm 2. Pseudocode of selection of locations in which to set up plants

$P \leftarrow \emptyset$
while ($|P| < k$) **do**
$p_2 \leftarrow$ Generate random probability
$l \leftarrow \arg max_{i=1,2,\ldots,m}\{\sum_{j=1}^{i} v_2(j) \mid \sum_{j=1}^{i} v_2(j) \leq p_2\}$
$P \leftarrow P \cup (l)$
end while
Return P

As indicated in Algorithm 1, the PLM is updated every time a new best solution, s_{best}, is found during the search. This means that, if $|P| = k$ in s_{best}, the following operations are carried out:

$$f_1(k) = f_1(k) + 1 \quad (18)$$

$$f_2(l) = f_2(l) + 1, \forall l \in P \quad (19)$$

This way, the influence of selecting k plants and the relevant locations are increased for the following sampling process according to equations (15) and (16), respectively.

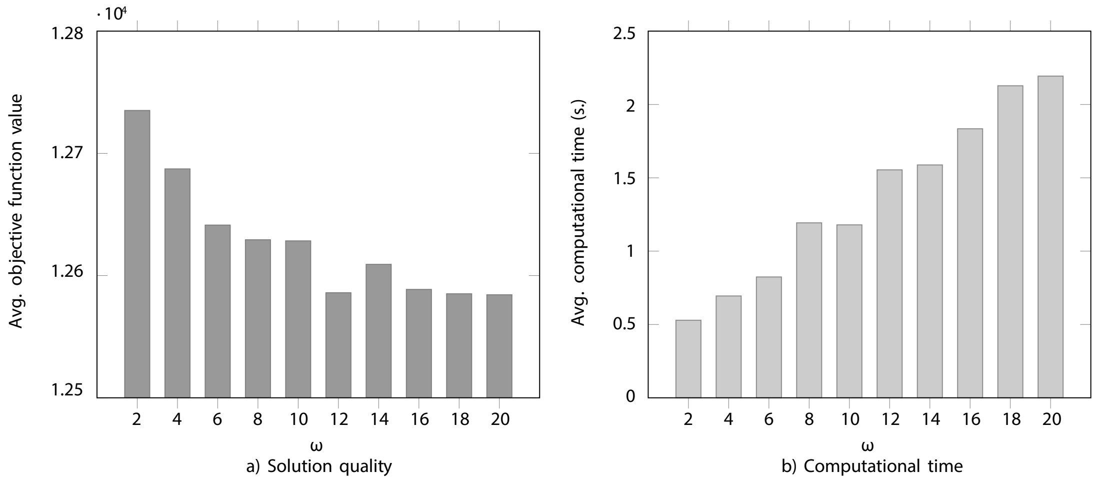


Fig. 2. Average objective function value and average computational time of the GRASP-PLM on a subset of problem instances.

B. Low-Level Problem

The solutions of the HP are not feasible solutions of the UPCLP. In order to overcome this fact, we must still determine the assignments of customers to plants and the routes to serve them. For this purpose, we use a GRASP.

The constructive phase of our GRASP builds a feasible solution of the UPCLP iteratively. At each step, one non-assigned customer is considered to be part of the solution. This customer can be assigned to one of the k plants defined by the high-level solution previously. Additionally, given a customer and a plant to serve it, this customer can be routed between each pair of consecutive nodes in the route of that plant.

A greedy function, $g : (a, b, c, p) \longrightarrow \mathbb{R}$, evaluates the impact on the objective function value of including the customer $b \in N$ between the consecutive customers or locations $a, c \in M \cup N$ in the route of the plant $p \in P$. That is:

$$g(a,b,c,p) = d_{ab} + d_{bc} - d_{ac} + c_{bp} \tag{20}$$

This impact on the objective function value originates in adding edge (a, b) and (b, c), assigning customer $b \in N$ to plant $p \in P$, and removing edge (a, c).

All the possible positions in which non-assigned customers can be placed in the routes of the selected plants are evaluated according to $g(\cdot,\cdot,\cdot,\cdot)$. These possibilities are ordered incrementally based on their impacts on the objective function value. The $\omega > 0$ best possibilities constitute a Restricted Candidate List (RCL). The value of parameter ω is set by the user. At each step, one candidate is extracted from the RCL according to the roulette wheel selection and the involved non-assigned customer is included in the solution under construction. The process finishes when all the customers have been assigned and routed. Consequently, a feasible solution of the UPCLP is, at this point, obtained.

The optimality of the solutions reported by the constructive phase is not guaranteed. The reason is found in using a greedy but myopic function to evaluate the impact of including non-assigned customers into the solutions under construction.

We propose an intensification phase based upon local search to explore the current region of the search space. With this goal in mind, we consider a single one-point movement to explore the neighbourhood of each solution obtained after applying the constructive phase. Given a feasible solution of the UPCLP, the one-point movement relocates a customer into a new position, in the same, different, or new route. We evaluate the impact on the objective function value of applying the one-point movement to relocate each customer into each possible target position. Particularly, removing a customer $i \in N$ from its current position in a route is computed as follows:

$$\mu(i) = d_{p(i)s(i)} - \left(d_{p(i)i} + d_{is(i)} - c_{i\sigma(i)} + o_{\sigma(i)} \cdot \phi(i,1)\right) \tag{21}$$

where $p(i)$ and $s(i)$ denote the previous and next nodes of customer i in its route, whereas $\phi(i, \alpha)$ is a binary variable that takes value 1 if and only if the route of i contains exactly α customers (*i.e.*, $|N_{\sigma(i)}| = \alpha$). It should be noted that, a plant can be removed from the solution when its route contains only one customer and this is relocated. Similarly, relocating a customer $i \in N$ before another node $j \in M \cup N$ is computed as follows:

$$\gamma(i,j) = \begin{cases} 2 \cdot d_{ij} + c_{ij} + o_j, & \text{if } j \in M \text{ and } \emptyset(j,0) = 1 \\ d_{p(j)i} + d_{ij} + c_{ij} - d_{p(j)j}, & \text{if } j \in M \text{ and } \emptyset(j,0) = 0 \\ d_{p(j)i} + d_{ij} + c_{i\sigma(j)} - d_{p(j)j}, & \text{otherwise} \end{cases} \tag{22}$$

The first case corresponds to those scenarios in which customer i is included in a new route starting from a plant set up at location $j \in M$. The remaining cases refer to those environments in which i is placed before a plant or another customer, respectively.

According to equations (21) and (22), the impact on the objective function value of relocating a customer $i \in N$ from its current position to the previous position of node $j \in M \cup N$ is computed as follows:

$$h(i,j) = \mu(i) + \gamma(i,j) \tag{23}$$

Finally, it is worth mentioning that the customers are randomly selected to be relocated. The travel costs between a customer $i \in N$ to relocate and the remaining nodes are sorted in increasing order so that we first evaluate relocating i before those nodes at minimum travel cost. In addition, at each step, the best improving neighbour solution is chosen.

VI. Computational Experiments

This section is dedicated to assessing the optimization model's performances introduced in Section IV and the Greedy Randomized Adaptive Search Procedure with Probabilistic Learning Model (GRASP-PLM) presented in Section V. In this regard, all the computational experiments presented hereunder have been conducted over the benchmark suite proposed in [42] and in-stances adapted from the TSPLIB [9]. All the problem instances are published to be freely used by the research community[1]. The mathematical model has been executed with CPLEX 12.3, set to all-default. Our proposed

[1] https://sites.google.com/site/gciports/plantcycle

optimization technique has been im-plemented in Java Standard Edition 7. In all cases, we have used a computer equipped with an Intel i7-3.50 GHz and 16 GB of RAM and performed 10 executions of each problem instance.

A. Parameter Setting

We have carried out a parameter setting before applying the GRASP-PLM. The parameters whose values must be determined are the size of the Restricted Candidate List (RCL), denoted as ω, and the number of iterations to perform.

Fig. 2. shows the average objective function value of 10 executions over a subset of problem instances with different sizes and the average computational time required by our GRASP-PLM when varying ω from 2 up to 20.

As can be checked, there is a strong tendency to improve the quality of the solutions reported by the GRASP-PLM when increasing the number of elements included in the RCL. However, increasing the value of ω gives rise to require larger computational times. The reason is that considering a large number of elements allows the search to have a relevant diversity, but it is harder to build the RCL at each step. This is because every solution must be evaluated before including it in the RCL, with its corresponding computational cost. In order to obtain a good balance between quality and computational time, in the remainder of this paper, we have executed our GRASP-PLM with $\omega = 12$.

With the aim to determine if there are significant differences between the groups of solutions obtained with different values of ω, the Friedman test [46] is applied to the average objective function value of these solutions. The significance level of this test is 0.05, which indicates that there are statistically significant differences among the solutions under analysis. Fig. 3. shows the interquartile range returned by the Friedman test for every possible values of ω. Depending on its interquartile range, every group of solutions is classified on different groups, identified by letters. Groups of solutions classified with the same letter do not have significant differences between them. As can be observed on this graphic, results with $\omega \in [12, 20]$ belong to group f, which implies that their solutions do not have significant differences. This consolidates the decision of using $\omega = 12$ in the subsequent computational experiments.

Moreover, when assessing the number of iterations, we have evaluated the average objective function value for the different groups of instances for 1000 iterations and $\omega = 12$. Fig. 4., Fig. 5., and Fig. 6. show the results when $n = 10, 25, 100$, respectively. As can be checked, when the iterations are increased, the average quality in terms of objective function value increases. However, it should be noted that the performance improvement is accompanied by a linear increase of the computational time. In the following experiments, we have selected 100 as a number of iterations to perform for each instance on the basis of maintaining a suitable and competitive performance in terms of computational time with the other approach reported in the literature [42].

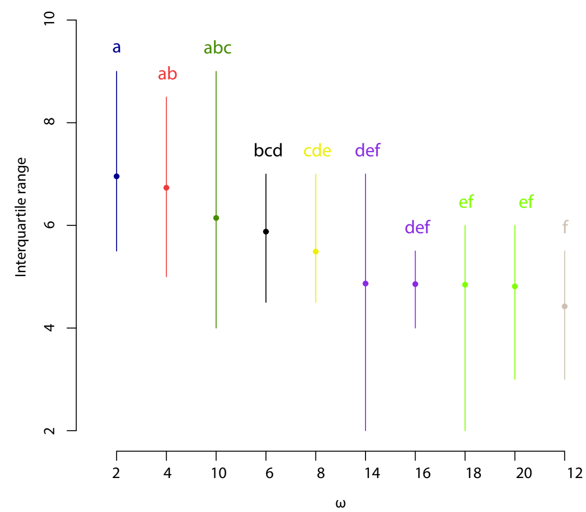


Fig. 3. Results of Friedman's test with different values of RCL size (i.e., ω).

B. Performance Evaluation

Tables I and II show the results obtained by the optimization model introduced in Section IV, the Honey Bees Mating Optimization algorithm (HBMO) proposed in [42], and our GRASP-PLM on a wide range of small-, medium-, and large-size instances proposed in [42]. In this case, column *Instances* reports the characteristics of the problem instances under analysis. For each problem instance, the number of customers, n, the number of locations, m, and the cost to set up a plant, o_p, are shown. In Table I, the computational results for the small- and medium-size problem instances are reported. These instances have a number of locations ranging from $m = 5$ up to $m = 25$, the number of customers ranges from $n = 10$ up to $n = 25$, whereas the cost to set up plants ranges from $o_p = 1$ up to $o_p = 1000$. On the other hand, Table II shows the results for the large-size problem instances. These instances have a number of locations ranging from $m = 50$ up to $m = 100$, the number of customers ranges from $n = 50$ up to $n = 100$, whereas the cost to set up plants ranges from $o_p = 1$ up to $o_p = 1000$. Each entry of the tables corresponds to a group of 5 problem instances. Hence, the average values are reported for each case.

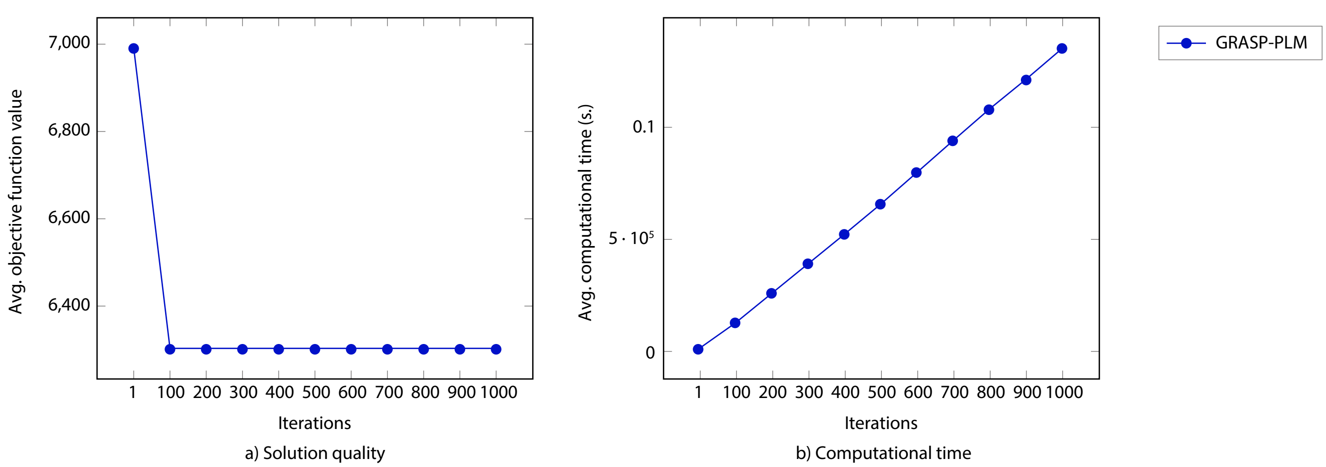
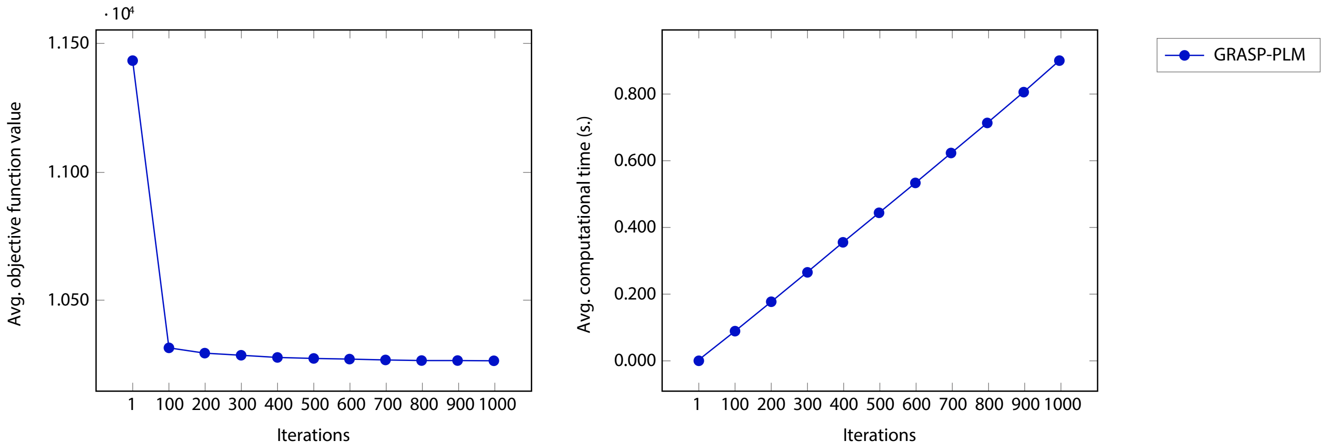


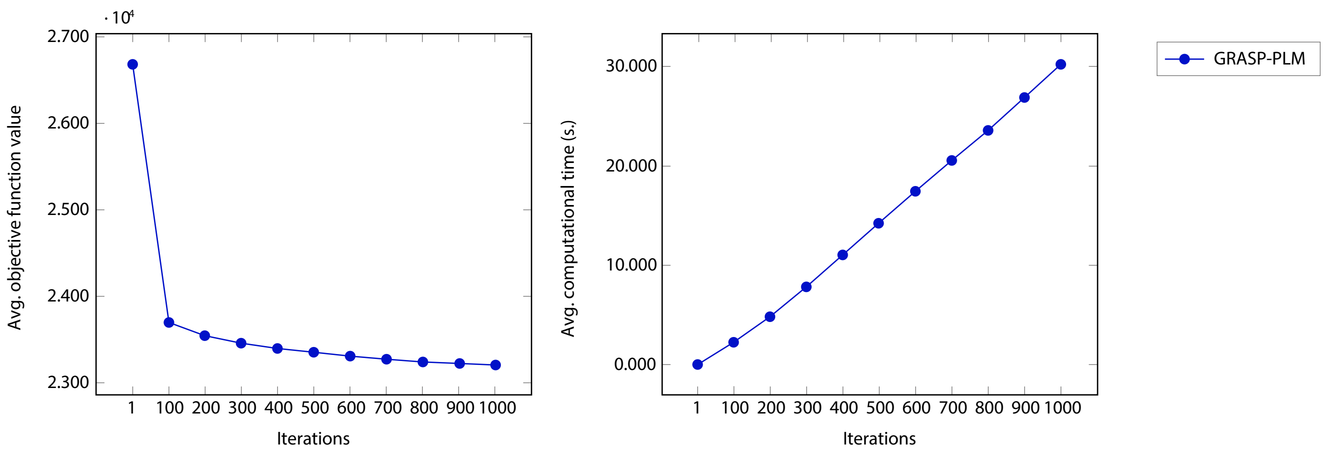
Fig. 4. Average objective function value and average computational time of the GRASP-PLM over the subset of problem instances with $n = 10$.



a) Solution quality

b) Computational time

Fig. 5. Average objective function value and average computational time of the GRASP-PLM over the subset of problem instances with $n = 25$.



a) Solution quality

b) Computational time

Fig. 6. Average objective function value and average computational time of the GRASP-PLM over the subset of problem instances with $n = 100$.

TABLE I. Computational Results for the Small- and Medium-Size Problem Instances

Instances			CPLEX		HBMO						GRASP-PLM					
					Deviation (%)			t (s.)			Deviation (%)			t (s.)		
n	m	O_p	Opt.	t (s.)	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.
10	5	1	6003.0	0.21	0.32	0.46	1.00	0.33	0.35	0.38	**0.00**	**0.04**	**0.04**	< 0.01	0.01	0.01
		250	6105.6	0.13	0.00	0.88	1.38	0.35	0.39	0.42	**0.00**	**0.00**	**0.00**	< 0.01	0.01	0.01
		500	6362.2	0.15	0.00	0.36	0.90	0.50	0.52	0.53	**0.00**	**0.00**	**0.00**	< 0.01	0.01	0.01
		964	6735.8	0.19	0.00	0.52	1.90	0.44	0.48	0.61	**0.00**	**0.00**	**0.00**	< 0.01	0.01	0.01
		1000	7956.6	0.20	0.35	0.35	0.35	0.52	0.56	0.60	**0.00**	**0.02**	**0.20**	< 0.01	0.01	0.01
10	10	1	4291.2	0.23	0.69	0.77	1.07	0.27	0.27	0.28	**0.00**	**0.00**	**0.00**	0.01	0.01	0.02
		13	5106.2	0.40	0.99	2.52	3.74	0.27	0.30	0.33	**0.00**	**0.00**	**0.00**	0.01	0.01	0.02
		250	5785.2	1.26	0.00	1.48	4.54	0.31	0.35	0.38	**0.00**	**0.00**	**0.00**	0.01	0.01	0.02
		500	6655.4	0.79	0.00	1.56	4.36	0.30	0.45	0.71	**0.00**	**0.00**	**0.00**	0.01	0.01	0.02
		1000	8042.2	0.74	0.01	1.34	3.02	0.34	0.44	0.59	**0.00**	**0.01**	**0.08**	0.01	0.01	0.02
25	10	1	8616.0	41.84	0.23	0.58	1.31	0.79	0.85	0.93	**0.00**	**0.00**	**0.03**	0.05	0.06	0.06
		250	10206.2	89.28	0.57	1.35	1.89	0.58	0.67	0.81	**0.00**	**0.04**	**0.16**	0.05	0.06	0.06
		500	12048.2	101.83	1.14	2.85	4.27	0.62	0.81	1.14	**0.00**	**0.39**	**0.91**	0.05	0.06	0.06
		508	11823.8	422.96	0.83	2.47	5.21	1.16	1.78	2.46	**0.00**	**0.11**	**0.33**	0.05	0.06	0.07
		1000	13524.8	114.33	1.65	4.83	7.68	1.21	1.82	2.88	**0.26**	**1.54**	**3.56**	0.05	0.05	0.06
25	25	1	6042.4	20.95	0.33	0.66	0.87	0.49	0.51	0.54	**0.00**	**0.00**	**0.00**	0.10	0.10	0.11
		70	7501.8	84.99	1.21	1.71	2.42	0.45	0.48	0.52	**0.00**	**0.00**	**0.00**	0.09	0.10	0.10
		250	9610.6	75.62	4.94	6.41	7.38	0.45	0.49	0.53	**0.00**	**0.14**	**0.41**	0.09	0.10	0.11
		500	9989.4	98.23	6.06	10.42	14.23	0.48	0.59	0.81	**0.02**	**0.34**	**1.02**	0.09	0.10	0.10
		1000	13096.2	1970.58	2.34	8.73	17.94	0.72	1.38	2.44	**1.71**	**3.47**	**5.41**	0.09	0.09	0.10

TABLE II. Computational Results for the Large-Size Problem Instances

Instances			HBMO						GRASP-PLM								
			Objetive function value			t (s.)			Objetive function value			Deviation (%)			t (s.)		
n	*m*	O_p	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.
50	50	1	9091.40	9149.40	9208.60	0.94	1.00	1.07	**9038.20**	**9046.60**	**9055.80**	**-0.59**	**-1.12**	**-1.66**	0.58	0.60	0.62
		250	14852.20	15071.80	15333.00	0.89	0.94	0.99	**13415.60**	**13498.14**	**13607.60**	**-9.70**	**-10.48**	**-11.30**	0.56	0.58	0.59
		454	19277.60	20717.64	22103.20	0.85	0.94	1.11	**16770.40**	**16991.82**	**17205.20**	**-12.96**	**-17.95**	**-22.10**	0.53	0.55	0.57
		500	19510.60	20606.24	21977.20	0.92	1.06	1.24	**16653.60**	**16962.30**	**17270.40**	**-14.56**	**-17.65**	**-21.33**	0.53	0.55	0.58
		1000	20792.20	23732.60	26991.40	1.10	2.36	4.68	**20391.60**	**21086.86**	**21981.00**	**-4.20**	**-11.98**	**-19.96**	0.52	0.55	0.57
100	50	1	16384.60	16457.72	16541.00	2.17	2.27	2.38	**15908.80**	**15935.32**	**15968.00**	**-2.87**	**-3.15**	**-3.45**	2.49	2.52	2.55
		250	24173.00	24686.96	25271.20	2.25	2.44	2.75	**21545.00**	**21782.50**	**21950.60**	**-10.62**	**-11.56**	**-12.97**	2.18	2.26	2.32
		500	31267.40	32430.72	33239.60	2.25	2.63	3.44	**26571.80**	**27178.06**	**27610.40**	**-15.03**	**-16.21**	**-16.95**	2.12	2.18	2.24
		964	35864.00	37897.32	40118.40	3.23	6.23	11.24	**31073.60**	**32194.08**	**33181.20**	**-13.30**	**-15.03**	**-17.21**	2.04	2.07	2.13
		1000	36165.20	38626.20	41159.60	3.41	5.42	8.86	**31509.40**	**32783.08**	**33630.60**	**-12.65**	**-14.94**	**-18.08**	2.04	2.10	2.15
100	100	1	13024.00	13124.72	13226.40	2.58	2.68	2.76	**12783.20**	**12804.74**	**12835.40**	**-1.84**	**-2.43**	**-2.95**	4.02	4.07	4.11
		13	13023.20	13097.48	13161.80	2.61	2.68	2.76	**12856.20**	**12884.52**	**12930.20**	**-1.28**	**-1.62**	**-1.75**	4.04	4.09	4.13
		250	24269.00	24911.24	25434.60	2.47	2.58	2.70	**21164.80**	**21417.90**	**21599.40**	**-12.79**	**-14.03**	**-15.08**	3.62	3.74	3.84
		500	31830.00	33783.60	36248.20	2.44	2.64	2.87	**26217.40**	**26825.74**	**27344.60**	**-17.59**	**-20.57**	**-24.55**	3.50	3.63	3.74
		1000	37435.60	40444.72	44159.60	3.16	4.46	6.19	**32248.20**	**33400.64**	**34472.20**	**-13.63**	**-17.17**	**-21.51**	3.41	3.54	3.69

TABLE III. Computational Results for Problem Instances Adapted From the TSPLIB [9]

Instances				CPLEX			GRASP-PLM								
							Objetive function value			Deviation (%)			t (s.)		
name	*n*	*m*	O_p	Obj.	Gap (%)	t (s.)	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.
burma14	11	3	1	57.0	0.00	0.13	57.00	57.00	57.00	0.00	0.00	0.00	0.05	0.08	0.31
			250	323.0	0.00	0.17	323.00	323.00	323.00	0.00	0.00	0.00	0.05	0.08	0.23
			500	569.0	0.00	0.11	569.00	569.00	569.00	0.00	0.00	0.00	0.04	0.05	0.06
			1000	1070.0	0.00	0.07	1070.00	1070.00	1070.00	0.00	0.00	0.00	0.05	0.06	0.07
ulises22	1	5	1	209.0	0.00	9.10	215.00	215.00	215.00	2.87	2.87	2.87	0.14	0.15	0.15
			250	520.0	0.00	13.48	520.00	520.00	520.00	0.00	0.00	0.00	0.12	0.13	0.14
			500	786.0	0.00	3.92	786.00	786.10	787.00	0.00	0.01	0.13	0.12	0.13	0.13
			1000	1345.0	0.00	5.05	1345.00	1345.00	1345.00	0.00	0.00	0.00	0.12	0.13	0.14
dantzig42	32	10	1	2519.0	26.03	3600.00	2243.00	2243.00	2243.00	-10.96	-10.96	-10.96	0.67	0.70	0.85
			250	3331.0	10.60	3600.00	3195.00	3284.20	3357.00	-4.08	-1.40	0.78	0.60	0.64	0.66
			500	3574.0	0.00	1809.00	3600.00	3633.50	3723.00	0.73	1.66	4.17	0.60	0.63	0.67
			1000	4536.0	0.00	1294.07	4536.00	4551.30	4571.00	0.00	0.34	0.77	0.56	0.61	0.85
hk48	36	12	1	33158.0	20.60	3600.00	33833.00	33956.00	34053.00	2.04	2.41	2.70	0.92	1.01	1.15
			250	41256.0	30.85	3600.00	36405.00	36441.50	36482.00	-11.76	-11.67	-11.57	0.91	0.94	0.96
			500	42324.0	31.64	3600.00	35588.00	35596.40	35609.00	-15.92	-15.90	-15.87	0.86	0.90	0.94
			1000	41308.0	27.41	3600.00	37827.00	38205.10	38378.00	-8.43	-7.51	-7.09	0.90	0.95	1.01
lin105	79	26	1	—	—	—	59713.00	60758.70	61868.00	—	—	—	7.72	7.96	8.26
			250	—	—	—	67029.00	67176.80	67332.00	—	—	—	7.96	8.52	10.00
			500	—	—	—	65138.00	66055.60	67283.00	—	—	—	7.64	8.02	8.53
			1000	—	—	—	67328.00	68483.50	69016.00	—	—	—	7.63	7.93	8.51
pr152	114	38	1	—	—	—	442586.00	459030.10	471043.00	—	—	—	23.77	24.46	25.95
			250	—	—	—	432127.00	447690.30	463756.00	—	—	—	23.99	25.05	25.91
			500	—	—	—	477308.00	496260.80	505313.00	—	—	—	25.15	25.39	25.72
			1000	—	—	—	461980.00	474970.20	490027.00	—	—	—	24.17	24.61	24.98

Column *CPLEX* in Table I reports the objective function value (*Opt.*) and computational time (*t* (*s*.)), measured in seconds, required by CPLEX when solving the optimization model. Columns *HBMO* and *GRASP-PLM* show the results obtained by the approximate techniques. In each case, the deviation (*Deviation* (%)) in terms of objective function value in comparison with the solutions obtained by CPLEX and the computational times (*t* (*s*.)) are shown. The minimum (*Min*.), average (*Avg*.), and maximum (*Max*.) deviations are reported in both cases. Similarly, the minimum (*Min*.), average (*Avg*.), and maximum (*Max*.) computational times used by the techniques are shown.

As can be checked in Table I, GRASP-PLM outperforms HBMO in terms of quality of the solutions and computational time. It should be noted that the deviations corresponding to the worst solutions reported by GRASP-PLM are still better than the average deviations reported by HBMO. Concerning the computational time, GRASP-PLM exhibits a competitive performance in comparison with HBMO and CPLEX. In this regard, GRASP-PLM maintains a stable temporal performance, requiring at most about 0.11 seconds on average. This computational advantage added to the relevant robustness shown by GRASP-PLM in terms of average deviations and difference between the best and the worst average deviations, makes our algorithm a competitive and suitable approach when tackling scenarios of this size.

Table II shows the computational results for the large-size problem instances. In this case, due to the fact that CPLEX is not able to provide a feasible solution, only the results obtained by the approximate approaches are reported. Namely, columns *HBMO* and *GRASP-PLM*. The objective function value (*Objective function value*) and computational time (*t* (*s*.)) are provided for each one. These columns include the minimum (*Min*.), average (*Avg*.), and maximum (*Max*.) computed values based upon the 10 executions. Moreover, we also provide the deviation (*Deviation* (%)) in terms of objective function value calculated in comparison with those objective function values provided by HBMO.

The computational results reported in Table II indicate that GRASP-PLM clearly improves HBMO on the basis of the quality of the solutions found. It should be noted that GRASP-PLM presents an average improvement of about 20% for a group of problem instances. Even though the computational times required by both methods are quite similar, it should be highlighted that GRASP-PLM presents a stable performance in terms of the difference between minimum and maximum computational times. On the other hand, as can be checked, HBMO reports the worst performance in this aspect. Hence, at the light of these results, it concludes that GRASP-PLM is also a competitive approach for large-size scenarios.

With the goal of checking if there are significant differences between the results obtained by the GRASP-PLM in comparison with those returned by the HBMO, the *paired-sample Wilcoxon test* is applied [47]. This test is performed with the results from the experiments summarized in Tables I and II, with a significance level of 0.05. This test concludes that there are significant differences between the solutions reported by both algorithms.

The performance of the GRASP-PLM has also been checked on representative problem instances adapted from other problems. In this case, the instances included in the TSPLIB [9] for the well-known Travelling Salesman Problem [44]. Concretely, a subset of points is randomly selected to be potential locations in which to set up plants, whereas the remaining points are customers.

The computational results for the instances adapted from TSPLIB are reported in Table III. As can be checked, as long as the size of the instances increases the performance of CPLEX in terms of quality of the solutions is compromised. Specifically, CPLEX is not able to provide a feasible solution for the largest instances (lin105 and pr152) within a time limit of 3600 seconds. Nevertheless, GRASP-PLM provides a feasible solution in all the cases. For the problem instances where both, CPLEX and GRASP-PLM, provide feasible solutions, we report the deviation of GRASP-PLM with respect to the best solution provided by CPLEX. In this regard, although GRASP-PLM is not able to provide the best solution in some cases, its temporal performance greatly outperforms CPLEX and maintains a similar performance regardless the variation of the value of o_p.

VII. Conclusions and Further Research

In this paper, a mathematical model and a metaheuristic approach based on Greedy Randomized Adaptive Search Procedure with a Probabilistic Learning Model (GRASP-PLM) for solving the Uncapacitated Plant Cycle Location Problem (UPCLP) have been studied. In order to evaluate their performances, extensive computational experiments over a wide range of problem instances from a benchmark suite proposed in the related literature is performed. Thus, these problem instances have been solved using both, the mathematical model implemented in a general purpose solver (i.e., CPLEX) and our GRASP-PLM. Moreover, we also report a comparison for these problem instances with an approximate approach published in the literature based on the Honey Bees Mating Optimization algorithm (HBMO) [42], which returns good quality solutions in short computational times when solving the UPCLP. Finally, a new set of instances from the well-known TSPLIB has been adapted in order to evaluate the performance of our approaches in different structured instances.

The computational results show that our GRASP-PLM exhibits a competitive performance in terms of computational times for the small- and medium-size problem instances in comparison with the computational times required by CPLEX and HBMO. Specifically, GRASP-PLM outperforms HBMO on the basis of the average objective function value. This improvement becomes even more substantial when tackling large-size problem instances, where CPLEX is not even able to provide a feasible solution. Unlike CPLEX, GRASP-PLM provides high-quality solutions through short computational times. It also reports a stable and slight increase in computational time when the problem size increases.

Considering the computational results provided in this work, we can claim that GRASP-PLM is an advisable algorithm for tackling the UPCLP in practical environments, being especially suitable in large-scale scenarios. It provides high-quality solutions by means of short computational times, in the range of a few seconds. Another outstanding characteristic exhibited by GRASP-PLM for this problem is that the variance of time and quality is quite stable in terms of the difference between the minimum and maximum values.

For future work, we intend to extend this model and algorithm to emergency scenarios. In these scenarios, we usually have several types of locations and vehicles and we have to schedule them for providing care to the victims as soon as possible. Lastly, the UPCLP can be studied assigning priorities to the customers, which can be applied also on emergency scenarios.

Acknowledgements

This work has been partially funded by the Spanish Ministry of Economy and Competitiveness with FEDER funds (project code PID2019-104410RB-I00/AEI/10.13039/501100011033).

References

[1] G. Cornuéjols, G. Nemhauser, and L. Wolsey, "The uncapacitated facility location problem," tech. rep., Cornell University Operations Research and Industrial Engineering, 1983.

[2] W. Ho, P. Ji, and P. K. Dey, "A multi-depot travelling salesman problem and its iterative and integrated approaches," *International Journal of Operational Research*, vol. 1, no. 4, pp. 382–395, 2006.

[3] S. V. Ukkusuri and W. F. Yushimito, "Location routing approach for the humanitarian prepositioning problem," *Transportation research record*, vol. 2089, no. 1, pp. 18–25, 2008.

[4] S. Schwarze, E. Lalla-Ruiz, and S. Voß, "Modeling the capacitated p-cable trench problem with facility costs," *Central European journal of operations research*, pp. 1–23, 2020.

[5] E. Lalla-Ruiz and S. Voß, "A popmusic approach for the multi-depot cumulative capacitated vehicle routing problem," *Optimization Letters*, vol. 14, no. 3, pp. 671–691, 2020.

[6] R. T. Berger, C. R. Coullard, and M. S. Daskin, "Location-routing problems with distance constraints," *Transportation Science*, vol. 41, no. 1, pp. 29–43, 2007.

[7] S. Çetiner, C. Sepil, and H. Süral, "Hubbing and routing in postal delivery systems," *Annals of Operations research*, vol. 181, no. 1, pp. 109–124, 2010.

[8] A. Bruns, A. Klose, and P. Stähly, "Restructuring of swiss parcel delivery services," *OR-Spektrum*, vol. 22, no. 2, pp. 285–302, 2000.

[9] G. Reinelt, "Tsplib–a traveling salesman problem library," *ORSA journal on computing*, vol. 3, no. 4, pp. 376–384, 1991.

[10] H. Min, V. Jayaraman, and R. Srivastava, "Combined location-routing problems: A synthesis and future research directions," *European Journal of Operational Research*, vol. 108, no. 1, pp. 1 – 15, 1998.

[11] D. Tuzun and L. Burke, "A two-phase tabu search approach to the location routing problem," *European Journal of Operational Research*, vol. 116, no. 1, pp. 87 – 99, 1999.

[12] A. Balakrishnan, J. Ward, and R. Wong, "Integrated facility location and vehicle routing models: Recent work and future prospects," *American Journal of Mathematical and Management Sciences*, vol. 7, no. 1-2, pp. 35–61, 1987.

[13] G. Nagy and S. Salhi, "Location-routing: Issues, models and methods," *European Journal of Operational Research*, vol. 177, no. 2, pp. 649 – 672, 2007.

[14] C. Prodhon and C. Prins, "A survey of recent research on location-routing problems," *European Journal of Operational Research*, vol. 238, no. 1, pp. 1 – 17, 2014.

[15] M. Schneider and M. Drexl, "A survey of the standard location-routing problem," *Annals of Operations Research*, vol. 259, no. 1-2, pp. 389–414, 2017.

[16] M. Drexl and M. Schneider, "A survey of variants and extensions of the location-routing problem," *European Journal of Operational Research*, vol. 241, no. 2, pp. 283 – 308, 2015.

[17] S. Alumur and B. Y. Kara, "A new model for the hazardous waste location-routing problem," *Computers & Operations Research*, vol. 34, no. 5, pp. 1406–1423, 2007.

[18] X. Yu, Y. Zhou, and X.-F. Liu, "The two-echelon multi-objective location routing problem inspired by realistic waste collection applications: The composable model and a metaheuristic algorithm," *Applied Soft Computing*, p. 106477, 2020.

[19] S. Kumar, V. Kumar-Solanki, S. K. Choudhary, A. Selamat, and R. González-Crespo, "Comparative study on ant colony optimization (aco) and k-means clustering approaches for jobs scheduling and energy optimization model in internet of things (iot)," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 1, pp. 107–116, 2020.

[20] M. Rabbani, R. Heidari, H. Farrokhi-Asl, and N. Rahimi, "Using metaheuristic algorithms to solve a multi-objective industrial hazardous waste location-routing problem considering incompatible waste types," *Journal of Cleaner Production*, vol. 170, pp. 227–241, 2018.

[21] A. Billionnet, S. Elloumi, and L. G. Djerbi, "Designing radio-mobile access networks based on synchronous digital hierarchy rings," *Computers & operations research*, vol. 32, no. 2, pp. 379–394, 2005.

[22] S. Zhang, M. Chen, and W. Zhang, "A novel location-routing problem in electric vehicle transportation with stochastic demands," *Journal of Cleaner Production*, vol. 221, pp. 567–581, 2019.

[23] I. A. Martínez-Salazar, J. Molina, F. Ángel-Bello, T. Gómez, and R. Caballero, "Solving a bi-objective transportation location routing problem by metaheuristic algorithms," *European Journal of Operational Research*, vol. 234, no. 1, pp. 25–36, 2014.

[24] M. Borhani, K. Akbari, A. Matkan, and M. Tanasan, "A multicriteria optimization for flight route networks in large-scale airlines using intelligent spatial information," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 1, pp. 123 – 131, 2019.

[25] B. Jarboui, H. Derbel, S. Hanafi, and N. Mladenović, "Variable neighborhood search for location routing," *Computers & Operations Research*, vol. 40, no. 1, pp. 47–57, 2013.

[26] C.-J. Ting and C.-H. Chen, "A multiple ant colony optimization algorithm for the capacitated location routing problem," *International Journal of Production Economics*, vol. 141, no. 1, pp. 34–44, 2013.

[27] T.-H. Wu, C. Low, and J.-W. Bai, "Heuristic solutions to multi-depot location-routing problems," *Computers & Operations Research*, vol. 29, no. 10, pp. 1393–1415, 2002.

[28] P. Yang and B. Li-Jun, "An integrated optimization problem in logistics and the pso solution," in *2006 International Conference on Service Systems and Service Management*, vol. 2, pp. 965–970, IEEE, 2006.

[29] Y. Marinakis and M. Marinaki, "A particle swarm optimization algorithm with path relinking for the location routing problem," *Journal of Mathematical Modelling and Algorithms*, vol. 7, no. 1, pp. 59–78, 2008.

[30] R. Caballero, M. González, F. M. Guerrero, J. Molina, and Paralera, "Solving a multiobjective location routing problem with a metaheuristic based on tabu search. application to a real case in andalusia," *European Journal of Operational Research*, vol. 177, no. 3, pp. 1751–1763, 2007.

[31] C. Prins, C. Prodhon, and R. W. Calvo, "Solving the capacitated location-routing problem by a grasp complemented by a learning process and a path relinking," *4OR*, vol. 4, no. 3, pp. 221–238, 2006.

[32] S. Barreto, C. Ferreira, J. Paixao, and B. S. Santos, "Using clustering analysis in a capacitated location-routing problem," *European Journal of Operational Research*, vol. 179, no. 3, pp. 968–977, 2007.

[33] I. López-Plata, C. Expósito-Izquierdo, E. Lalla-Ruiz,B. Melián-Batista, and J. M. Moreno-Vega, "A greedy randomized adaptive search procedure for solving the uncapacitated plant cycle problem," in *International Conference on Computer Aided Systems Theory*, pp. 263–270, Springer, 2015.

[34] J. Escobar, R. Linfati, M. Baldoquin, and P. Toth, "A granular variable tabu neighborhood search for the capacitated location-routing problem," *Transportation Research Part B: Methodological*, vol. 67, no. 0, pp. 344 – 356, 2014.

[35] R. Baldacci, A. Mingozzi, and R. Calvo, "An exact method for the capacitated location-routing problem.," *Operations Research*, vol. 59, no. 5, pp. 1284–1296, 2011.

[36] J. Belenguer, E. Benavent, C. Prins, C. Prodhon, and R. Calvo, "A branch-and-cut method for the capacitated location-routing problem," *Computers & Operations Research*, vol. 38, no. 6, pp. 931 – 941, 2011.

[37] C. Contardo, J.F. Cordeau, and B. Gendron, "An exact algorithm based on cut-and-column generation for the capacitated location-routing problem," *INFORMS Journal on Computing*, vol. 26, pp. 88–102, 2014.

[38] A. Perugia, L. Moccia, J. Cordeau, and G. Laporte, "Designing a home-to-work bus service in a metropolitan area," *Transportation Research Part B: Methodological*, vol. 45, no. 10, pp. 1710 – 1726, 2011.

[39] I. Rodríguez-Martín, J.-J. Salazar-González, and H. Yaman, "A branch-and-cut algorithm for two-level survivable network design problems," *Computers & Operations Research*, vol. 67, pp. 102–112, 2016.

[40] M. Labbé, I. Rodríguez-Martín, and J. Salazar-Rodríguez, "A branch-and-cut algorithm for the plant-cycle location problem," *Journal of the Operational Research Society*, vol. 55, no. 5, pp. 513 – 520, 2004.

[41] M. Albareda-Sambola, J. Díaz, and E. Fernández, "A compact model and tight bounds for a combined location-routing problem," *Computers & Operations Research*, vol. 32, no. 3, pp. 407 – 428, 2005.

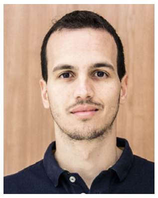
[42] B. Melián-Batista, J. Moreno-Vega, N. Vaswani, and R. Yumar, "A nature inspired approach for the uncapacitated plant cycle location problem," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2008)* (N. Krasnogor, M. B. Melián-Batista, J. A. Moreno-Pérez, J. M. Moreno- Vega, and D. A. Pelta, eds.), vol. 236 of *Studies in Computational Intelligence*, pp. 49–60, Springer Berlin Heidelberg, 2009.

[43] C. Fleming, S. Griffis, and J. Bell, "The effects of triangle inequality on the vehicle routing problem," *European Journal of Operational Research*, vol. 224, no. 1, pp. 1 – 7, 2013.

[44] G. Gutin and A. Punnen, *The Traveling Salesman Problem and Its Variations*. Springer, 1 edition ed., 2002.

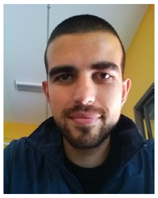
[45] T. Feo and M. Resende, "Greedy randomized adaptive search procedures," *Journal of Global Optimization*, vol. 6, no. 2, pp. 109–133, 1995.

[46] W. W. Daniel, *Applied nonparametric statistics*. Boston: PWS-Kent Publishing Company, 1990.
[47] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.



Israel López-Plata

He received the Computer Science Engineering degree in 2009 and the Ph.D. in Computer Science in 2020, both from the University of La Laguna, Spain. He has 12 year experience in software development related with logistics and Artificial Intelligence. Currently he is working as professor on Department of Computer and Systems Engineering, University of La Laguna. His research is focused on optimization applied to the resolution of logistic problems, in addition to other Artificial Intelligence techniques.



Christopher Expósito-Izquierdo

Christopher Expósito-Izquierdo is engineering in Computer Science and has obtained his Ph.D. on the optimization of the transshipment operations in maritime container terminals at the University of La Laguna. He is nowadays lecturer at the Department of Computer and Systems Engineering, University of La Laguna, in which his activity is mainly dedicated to the analysis and optimisation of logistic operations and transportation, especially focused on maritime container terminals.

Eduardo Lalla-Ruiz

Eduardo Lalla-Ruiz is an assistant professor at the Department of High Tech and Entrepreneurship in the Industrial Engineering and Business Information Systems (IEBIS) section at the University of Twente (The Netherlands). He holds a bachelor degree in industrial engineering, an advanced engineering degree in industrial automation, and a master of advanced studies in computer sci-ence and artificial intelligence from the University of La Laguna (Spain). At that same institution, Eduardo completed his Ph.D. and was awarded an extraordinary doctoral distinction. After finishing his Ph.D., Eduardo joined the Institute of Information Systems (University of Hamburg, Germany) as a researcher and lecturer. There, Eduardo became a fellow of the prestigious Alexander von Humboldt Research Foundation. Eduardo has published several research papers in high-quality journals, collaborated in international research projects, served as editor, guest editor, and referee in relevant journals, and participated in the organization of sessions and international conferences. Recently, he was awarded as runner up for the EURO excellence in practice award 2021 as well as the meritorious service award in Transportation Science (INFORMS). His research interests lie in the fields of operations research, logistics, artificial intelligence, and mathematical programming.

Belén Melián-Batista

She received the M.Sc. degree in mathematics and the Ph.D. in computer science from the University of La Laguna, Canary Islands, Spain, in 1999 and 2003, respectively. She is a Full Professor in Computer Science and Artificial Intelligence at the Department of Computer and Systems Engineering, University of La Laguna, Spain. She has published more than 30 scientific papers in leading scientific journals, and has been Advisor of 4 Ph.D. dissertations. She has been Principal Researcher in a variety of national research projects. Her research focuses on the design and implementation of metaheuristic techniques for solving combinatorial optimization problems in different application areas. Specifically, her current main interest is the application of these techniques to solve logistic and transportation problems.

J. Marcos Moreno-Vega

He received the M.Sc. degree in Mathematics in 1990 and the Ph.D. degree in Computer Science in 1997, both from the University of La Laguna, Canary Islands, Spain. Currently, he is a Full Professor at the Department of Computer and Systems Engineering, University of La Laguna, in the area of Computer Science and Artificial Intelligence. His research interests include soft computing, metaheuristics and data mining applied to logistic problems.