Available online at www.sciencedirect.com
ScienceDirect
Computers & Operations Research 34 (2007) 1800–1823
computers & operations research
www.elsevier.com/locate/cor

Efficient primal-dual heuristic for a dynamic location problem☆

Joana Dias[a,*], M. Eugénia Captivo[b], João Clímaco[a]

[a] Faculdade de Economia and INESC-Coimbra, Universidade de Coimbra, Av. Dias da Silva,165, 3004-512 Coimbra, Portugal
[b] Universidade de Lisboa, Faculdade de Ciências, Centro de Investigação Operacional, Campo Grande, Bloco C6, Piso 4, 1749-016 Lisboa, Portugal

Available online 23 September 2005

Abstract

In this paper the dynamic location problem with opening, closure and reopening of facilities is formulated and an efficient primal-dual heuristic that computes both upper and lower limits to its optimal solution is described. The problem here studied considers the possibility of reconfiguring any location more than once over the planning horizon. This problem is NP-hard (the simple plant location problem is a special case of the problem studied). A primal-dual heuristic based on the work of Erlenkotter [A dual-based procedure for uncapacitated facility location. Operations Research 1978;26:992–1009] and Van Roy and Erlenkotter [A dual-based procedure for dynamic facility location. Management Science 1982;28:1091–105] was developed and tested over a set of randomly generated test problems. The results obtained are quite good, both in terms of the quality of lower and upper bounds calculated as in terms of the computational time spent by the heuristic. A branch-and-bound procedure that enables to optimize the problem is also described and tested over the same set of randomly generated problems.

Keywords: Location; Heuristics; Branch and bound

1. Introduction

The simple plant location problem (SPLP) consists of choosing the locations where to install facilities, in such a way that all clients can be served by, at least, one operating facility, minimizing the total costs involved (both fixed costs of opening facilities and costs of assigning clients to open facilities). If there

☆ This research was partially supported by the research projects POCTI/ISFL-1/152, POCTI/MAT/139/2001, POSI/ISFL-13/308 and POSI/SRI/37346/2001.
* Corresponding author. Fax: +351 239 403 511.
E-mail address: joana@fe.uc.pt (J. Dias).

0305-0548/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2005.07.005

are no capacity constraints associated to the facilities, each client will be assigned to exactly one facility (the operating facility corresponding to the smallest assignment cost). The SPLP has been widely studied in the literature [1–8].

The dynamic location problem is a generalization of the SPLP that considers the location of facilities during a planning horizon. In each period of the planning horizon the assignment of each client to an operating facility has to be guaranteed.

According to Erlenkotter [9], two main characteristics force the consideration of a dynamic location problem: the assignment costs change significantly during the planning horizon; there must be significant costs for relocating facilities. If the first characteristic is absent, the problem can be formulated as a SPLP, if the second characteristic is absent a set of disconnected SPLP can be considered (one for each period of the planning horizon).

Wesolowsky [10] was one of the first authors who studied the dynamic location problem. In his paper, the author generalizes the Weber problem, considering the existence of several time periods. Wesolowsky and Truscott [11] describe a discrete location problem considering a fixed number of open facilities in each time period. They describe a resolution method based on dynamic programming. Fong and Srinivasan [12,13] study the problem of determining a schedule of capacity expansions of facilities over a planning horizon (determining the location, size and timing of construction of facilities). They present a heuristic procedure that tries to improve a feasible solution by exchanging capacities between pairs of regions. Van Roy and Erlenkotter [14] describe the dynamic simple plant location problem without capacity constraints, considering that a facility can be open in the beginning of a time period (remaining open until the end of the planning horizon). If there are open facilities at the beginning of the planning horizon, these existing facilities can be closed at the end of a time period, remaining closed until the end of the planning horizon. The authors describe a branch-and-bound algorithm that uses a dual ascent procedure and present computational results that show the efficiency of the method. Laporte and Dejax [15] study a dynamic location-routing problem and present two different solution approaches to tackle the problem. Jacobsen [16] describes several multiperiod capacitated location models and methods. Shulman [17] describes a dynamic location problem with capacity constraints. The author considers a limited number of possible facility maximum capacities, accepting that more than one facility can be located at the same site in different time periods (therefore increasing the existing capacity in one location). The author presents an algorithm based on the Lagrangean relaxation technique. Galvão and Santibãnez-Gonzalez [18] consider a generalization of the p-median problem to several time periods (clients should be assigned to a set of p_k facilities in period k, minimizing installation and transportation costs). They describe a Lagrangean heuristic and show some computational results. Melachrinoudis et al. [19] describe a multiobjective, capacitated dynamic model for the location of landfills. Hinojosa et al. [20] deal with the problem of multiproduct dynamic location and develop a heuristic procedure based on a Lagrangean relaxation. Antunes and Peeters [21] present a dynamic location problem that considers the location of new facilities, or the closure, reduction or expansion of existing facilities (this work was based on a real case) and study the advantages and limitations of simulated annealing to solve this problem.

In most problems described in the literature that consider the possibility of opening and closing of facilities, no location can have its configuration changed more than once during the planning horizon. This means that if a facility is open at the beginning of a time period it will remain open until the end of the planning horizon, and if a facility is closed at the end of a time period, it will remain closed. The paper by Wesolowsky and Truscott [11] is an exception to this rule, but in this article the fixed costs of opening a facility are the same whether a facility was already operating in a given location or not.

Furthermore, the authors do not consider either the existence of operating costs during the time periods the facilities are operating, or the existence of fixed closing costs when a facility is closed. Chardaire et al. [22] and Canel et al. [23] consider the possibility of a facility being open, closed and reopen more than once. Nevertheless, the authors do not differentiate between open and reopen fixed costs (which, in most cases, are clearly different), and present a non-linear objective function.

The dynamic location problem formulated in this paper considers the possibility of opening, closing and reopening a facility more than once during the planning horizon. The differentiation between the opening and the reopening of a facility is convenient because it allows the differentiation of the corresponding fixed costs. There are several situations where these costs are clearly different (for instance, if the facilities have already been acquired or, in case of locating obnoxious facilities, if studies of environmental impact have already been done). The model proposed also considers the existence of operating and closing costs (that most of the times cannot be ignored). It is also possible to consider the existence of open facilities at the beginning of the planning horizon.

The model described is a generalization of the SPLP, so it is straightforward to conclude that it is a NP-hard problem [4]. A primal-dual heuristic is developed that builds primal and dual admissible solutions, trying to force the complementary conditions to be satisfied. Whenever the heuristic is unable to find the optimal solution to the problem, it provides a primal admissible solution and a lower bound to the optimal solution value. Therefore, it is always possible to check the quality of the best primal solution calculated.

In the next section the problem is formulated. In Section 3, the dual problem is derived. In Section 4, the primal-dual heuristic is described. In Section 5 a branch-and-bound procedure based on the heuristic of Section 4 guaranteeing the calculation of the optimal solution is described. In Section 6, the results of the computational tests are presented. In Section 7, some conclusions are pointed out, and future work directions are outlined.

2. Problem formulation

Let us define the following notation:

J	$\{1, \ldots, n\}$ set of indexes corresponding to the clients' locations;
I	$\{1, \ldots, m\}$ set of indexes corresponding to facilities' possible locations;
T	number of time periods considered in the planning horizon;
c_{ij}^t	cost of assigning client j to facility i in period t;
FA_{it}^{ξ}	fixed cost of opening a facility i at the beginning of period t, and closing the facility at the end of period ξ (the facility will be in operation from the beginning of t to the end of ξ);
FR_{it}^{ξ}	fixed cost of reopening a facility i at the beginning of period t, and closing it at the end of period ξ (the facility will be in operation from the beginning of t to the end of ξ);

and let us define the variables:

$$a_{it}^{\xi} = \begin{cases} 1 & \text{if facility } i \text{ is opened at the beginning of period } t \text{ and stays open until} \\ & \text{the end of period } \xi, \\ 0 & \text{otherwise,} \end{cases}$$

$$r_{it}^{\xi} = \begin{cases} 1 & \text{if facility } i \text{ is reopen at the beginning of period } t \text{ and stays open until} \\ & \text{the end of period } \xi, \\ 0 & \text{otherwise,} \end{cases}$$

$$x_{ij}^{t} = \begin{cases} 1 & \text{if client } j \text{ is assigned to facility } i \text{ during period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Variables r_{it}^{ξ} are only defined for period t greater than one. The fixed costs incurred by opening or reopening a facility i from period t to ξ should consider the opening costs at period t, fixed operating costs from t to ξ and also closure costs at period ξ. The dynamic location problem that allows facilities to open, close and reopen more than once during the planning horizon will be formulated as DLPOCR:

DLPOCR:

$$\text{Min} \quad \sum_{t=1}^{T}\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}^{t} x_{ij}^{t} + \sum_{i=1}^{m}\sum_{t=1}^{T}\sum_{\xi=t}^{T} FA_{it}^{\xi} a_{it}^{\xi} + \sum_{i=1}^{m}\sum_{t=2}^{T}\sum_{\xi=t}^{T} FR_{it}^{\xi} r_{it}^{\xi} \tag{1}$$

s.t.:

$$\sum_{i=1}^{m} x_{ij}^{t} = 1 \quad \forall j \in J, \ \ t = 1, \ldots, T, \tag{2}$$

$$\sum_{\tau=1}^{t}\sum_{\xi=t}^{T} (a_{i\tau}^{\xi} + r_{i\tau}^{\xi}) - x_{ij}^{t} \geqslant 0 \quad \forall\, i \in I, \ \ j \in J, \ \ t = 1, \ldots, T, \tag{3}$$

$$\sum_{\tau=1}^{t-1}\sum_{\xi=\tau}^{t-1} a_{i\tau}^{\xi} - \sum_{\xi=t}^{T} r_{it}^{\xi} \geqslant 0 \quad \forall\, i \in I, \ \ t = 1, \ldots, T, \tag{4}$$

$$\sum_{t=1}^{T}\sum_{\xi=t}^{T} a_{it}^{\xi} \leqslant 1 \quad \forall\, i \in I, \tag{5}$$

$$\sum_{\tau=1}^{t}\sum_{\xi=t}^{T} (a_{i\tau}^{\xi} + r_{i\tau}^{\xi}) \leqslant 1 \quad \forall\, i \in I, \ \ t = 2, \ldots, T, \tag{6}$$

$$a_{it}^{\xi}, x_{ij}^{t} \in \{0, 1\} \quad \forall\, i \in I, \ \ j \in J, \ \ t = 1, \ldots, T, \ \ \xi \geqslant t,$$

$$r_{it}^{\xi} \in \{0, 1\} \quad \forall\, i \in I, \ \ t = 2, \ldots, T, \ \ \xi \geqslant t. \tag{7}$$

Constraints (2) guarantee that, in every time period, each client is fully assigned to exactly one facility; constraints (3) assure that, in every time period, a client can only be assigned to facilities that are operational in that time period; constraints (4) guarantee that a facility can only be reopened at the beginning of period t if it has already been opened earlier and is not in operation at the beginning of period t; constraints (5) impose that a facility can only be opened once during the planning horizon; constraints (6) assure that, in every time period, only one facility can be opened in each location. Constraints (5) and (6) need to be considered explicitly only when there are negative fixed costs. If all fixed costs are greater than zero, then (5) and (6) could be replaced by

$$\sum_{\tau=2}^{t}\sum_{\xi=\tau}^{T} r_{i\tau}^{\xi} \leqslant 1 \quad \forall\, i \in I, \quad t = 2, \ldots, T. \tag{8}$$

The formulation presented will also be valid if there are facilities operating before the beginning of the planning horizon. Consider the set $I_c \subset I$ such that for $i \in I_c$, i is opened before the beginning of the planning horizon. The facilities belonging to I_c can remain opened during the first time period or can be closed even before the beginning of time period 1. Consider:

$$a_{i1}^{\xi} = \begin{cases} 1 & \text{if facility } i \text{ is already open before the first time period and stays open until} \\ & \text{the end of period } \xi, \\ 0 & \text{otherwise,} \end{cases}$$

$$a_{it}^{\xi} = \begin{cases} 1 & \text{if facility } i \text{ is closed before the beginning of the first time period, is reopen for the} \\ & \text{first time at the beginning of period } t \text{ and stays open until the end of period } \xi, \\ 0 & \text{otherwise.} \end{cases}$$

Variables a_{i1}^{ξ} are defined for $i \in I_c$ and variables a_{it}^{ξ} are defined for $i \in I_c, t > 1$. The fixed costs associated with variables a_{i1}^{ξ}, $i \in I_c$, correspond to the operating costs during the periods the facility is in operation, plus the costs of closing the facility at the end of ξ. The fixed costs associated with variables a_{it}^{ξ}, $i \in I_c$ and $t > 1$, correspond to the costs incurred by closing the facility before the beginning of time period 1, plus the costs of reopening the facility at the beginning of time period t, plus operating costs during the periods the facility is opened, plus the costs of closing the facility. Reinterpreting variables a_{it}^{ξ}, $i \in I_c$, in this way makes it possible to use the formulated problem without having to increase the number of restrictions or variables.

3. Formulation of the dual problem

3.1. Dual formulation

Multiplying constraints (5) and (6) by -1 and associating dual variables v_j^t with constraints (2), w_{ij}^t with constraints (3), u_i^t with constraints (4), ρ_i with constraints (5) and π_i^t with constraints (6), the dual problem of DLPOCR can be formulated as D-DLPOCR:

D-DLPOCR:

$$\text{Max} \quad \sum_{t=1}^{T}\sum_{j=1}^{n} v_j^t - \sum_{i=1}^{m} \rho_i - \sum_{t=1}^{T}\sum_{i=1}^{m} \pi_i^t \tag{9}$$

s.t.:

$$v_j^t - w_{ij}^t \leqslant c_{ij}^t \quad \forall\, i \in I, \quad j \in J, \quad t = 1, \ldots, T, \tag{10}$$

$$\sum_{j=1}^{n}\sum_{\tau=t}^{\xi} w_{ij}^{\tau} + \sum_{\tau=\xi+1}^{T} u_i^{\tau} - \rho_i - \sum_{\tau=t}^{\xi} \pi_i^{\tau} \leqslant FA_{it}^{\xi} \quad \forall\, i \in I, \quad t=1,\ldots,T, \quad \xi = t,\ldots,T, \tag{11}$$

$$\sum_{j=1}^{n}\sum_{\tau=t}^{\xi} w_{ij}^{\tau} - u_i^{t} - \sum_{\tau=t}^{\xi} \pi_i^{\tau} \leqslant FR_{it}^{\xi} \quad \forall\, i \in I, \quad t=2,\ldots,T, \quad \xi = t,\ldots,T, \tag{12}$$

$$w_{ij}^{t}, u_i^{t}, \rho_i, \pi_i^{t} \geqslant 0 \quad \forall\, i \in, \quad j \in J, \quad t=1,\ldots,T.$$

An equivalent condensed formulation is obtained by considering $w_{ij}^{t} = \max\{0, v_j^{t} - c_{ij}^{t}\}$:
CD-DLPOCR:

$$\text{Max} \quad \sum_{t=1}^{T}\sum_{j=1}^{n} v_j^{t} - \sum_{i=1}^{m} \rho_i - \sum_{t=1}^{T}\sum_{i=1}^{m} \pi_i^{t}$$

s.t.:

$$\sum_{j=1}^{n}\sum_{\tau=t}^{\xi} \max\{0, v_j^{\tau} - c_{ij}^{\tau}\} \leqslant FA_{it}^{\xi} - \sum_{\tau=\xi+1}^{T} u_i^{\tau} + \rho_i + \sum_{\tau=t}^{\xi} \pi_i^{\tau}$$
$$\forall\, i \in I, \quad t=1,\ldots,T, \quad \xi=t,\ldots,T \tag{13}$$

$$\sum_{j=1}^{n}\sum_{\tau=t}^{\xi} \max\{0, v_j^{\tau} - c_{ij}^{\tau}\} \leqslant FR_{it}^{\xi} + u_i^{t} + \sum_{\tau=t}^{\xi} \pi_i^{\tau} \quad \forall\, i \in I, \quad t=1,\ldots,T, \quad \xi=t,\ldots,T \tag{14}$$

$$u_i^{t}, \rho_i, \pi_i^{t} \geqslant 0 \quad \forall\, i \in I, \quad j \in J, \quad t=1,\ldots,T.$$

3.2. Complementary conditions

Let us define:

$$SA_{it}^{\xi} = FA_{it}^{\xi} - \sum_{\tau=\xi+1}^{T} u_i^{\tau} + \rho_i + \sum_{\tau=t}^{\xi} \pi_i^{\tau} - \sum_{j=1}^{n}\sum_{\tau=t}^{\xi} \max\{0, v_j^{\tau} - c_{ij}^{\tau}\}$$
$$\forall\, i \in I, \quad t=1,\ldots,T, \quad \xi=t,\ldots,T, \tag{15}$$

$$SR_{it}^{\xi} = FR_{it}^{\xi} + u_i^{t} + \sum_{\tau=t}^{\xi} \pi_i^{\tau} - \sum_{j=1}^{n}\sum_{\tau=t}^{\xi} \max\{0, v_j^{\tau} - c_{ij}^{\tau}\}$$
$$\forall\, i \in I, \quad t=2,\ldots,T, \quad \xi=t,\ldots,T, \tag{16}$$

$$S_{it}^{\xi} = \min\{SA_{it}^{\xi}, SR_{it}^{\xi}\} \quad \forall\, i \in I, \quad t=1,\ldots,T, \quad \xi=t,\ldots,T. \tag{17}$$

The following complementary conditions hold in presence of optimal primal and dual solutions to the problems DLPOCR and its dual problem D-DLPOCR (when there is no duality gap)

$$\left(\sum_{\tau=1}^{t}\sum_{\xi=t}^{T}(a_{i\tau}^{\xi}+r_{i\tau}^{\xi})-x_{ij}^{t}\right)w_{ij}^{t}=0 \quad \forall\, i\in I,\;\; j\in J,\;\; t=1,\ldots,T, \tag{18}$$

$$\left(\sum_{\tau=1}^{t-1}\sum_{\xi=\tau}^{t-1}a_{i\tau}^{\xi}-\sum_{\xi=t}^{T}r_{it}^{\xi}\right)u_{i}^{t}=0 \quad \forall\, i\in I,\;\; t=1,\ldots,T, \tag{19}$$

$$\left(\sum_{t=1}^{T}\sum_{\xi=t}^{T}a_{it}^{\xi}-1\right)\rho_{i}=0 \quad \forall\, i\in I, \tag{20}$$

$$\left(\sum_{\tau=1}^{t}\sum_{\xi=\tau}^{T}(a_{i\tau}^{\xi}+r_{i\tau}^{\xi})-1\right)\pi_{i}^{t}=0 \quad \forall\, i\in I,\;\; t=2,\ldots,T, \tag{21}$$

$$SA_{it}^{\xi}a_{it}^{\xi}=0 \quad \forall\, i\in I,\;\; t=1,\ldots,T,\;\; \xi=t,\ldots,T, \tag{22}$$

$$SR_{it}^{\xi}r_{it}^{\xi}=0 \quad \forall\, i\in I,\;\; t=2,\ldots,T,\;\; \xi=t,\ldots,T. \tag{23}$$

4. Primal-dual heuristic

The primal-dual heuristic that has been developed builds admissible primal solutions based on admissible dual solutions to problem D-DLPOCR, trying to force conditions (18)–(23) to be satisfied. If the heuristic can find a pair of primal and dual solutions that satisfy all the complementary conditions, then it has discovered the optimal solution. When this does not happen, the best dual solution known will give a lower bound to the optimum value of the primal objective function. The heuristic functioning scheme is the following:

1. Initialization of dual variables.
2. Dual ascent procedure for dual variables v_j^t.
3. Primal procedure.
4. Dual adjustment procedure for dual variables ρ_i. If the dual solution is changed go to 2.
5. Repeat the dual-primal adjustment procedure for variables v_j^t until there is no improvement in the dual objective function value.
6. Dual adjustment procedure for dual variables ρ_i. If the dual solution is changed go to 2.
7. Dual ascent procedure for dual variables u_i^t. If the dual solution is changed go to 2.

8. Dual descent procedure for dual variables u_i^t. If the dual solution is changed go to 2.
9. Dual adjustment procedure for variables π_i^t. If the dual solution is changed go to 2.

The heuristic will stop when the optimal primal solution is found, or when there are no improvements in primal or dual objective function values.

The order in which the several procedures are executed can be changed, giving rise to a number of very similar heuristics. The sequence presented was the one that gave the best results with the preliminary test problems solved.

4.1. Initialization of dual variables

Dual variables are initialized as follows:

1.

$$v_j^t = \min_i \{c_{ij}^t\}, \quad \forall j \in J, \quad t = 1, \ldots, T; \quad \pi_i^t = 0, \quad \forall\, i \in I, \quad t = 1, \ldots, T;$$

2.

$$u_i^t = \begin{cases} \max\left\{0, -\min\limits_{\substack{t \\ \xi \geqslant t}} FR_{it}^{\xi}\right\}, & \text{if } \exists FR_{it}^{\xi} < 0, \\ 0, & \text{otherwise} \end{cases} \quad \forall\, i \in I, \quad t = 1, \ldots, T;$$

3.

$$\rho_i = \max\left\{0, -\min_{\substack{t \\ \xi \geqslant t}} \left(FA_{it}^{\xi} - \sum_{\tau=\xi+1}^{T} u_i^{\tau} \right) \right\}, \quad \forall\, i \in I, \quad t = 1, \ldots, T.$$

4.2. Dual ascent procedure for variables v_j^t

This procedure tries to increase all dual variables v_j^t, $j \in J^+$, $J^+ \subset J$. If this procedure is executed in step 2 of the heuristic, then J^+ is the whole set J. Whenever this procedure is executed within other procedure, the set J^+ will be defined before dual ascent procedure is called. This procedure is a straightforward adaptation of the one described in [14]. The only difference is in the updating step of slacks $SA_{i\tau}^{\xi}$ and $SR_{i\tau}^{\xi}$: each time the value of v_j^t is increased, slacks $SA_{i\tau}^{\xi}$ and $SR_{i\tau}^{\xi}$, $\tau \leqslant t \leqslant \xi$, have to be updated (its value will be decreased by the same amount the dual variable was increased, if v_j^t is greater than or equal to c_{ij}^t).

4.3. Primal procedure

The primal-dual heuristic intends to build primal admissible solutions to DLPOCR based on dual admissible solutions, trying to force conditions (18)–(23) to be satisfied. Consider the following set definitions:

$$
\begin{aligned}
&I^* = \{(i, \tau, \xi) : S_{i\tau}^{\xi} = 0\},\\
&I_t^* = \{i : (i, \tau, \xi) \in I^* \text{ and } \tau \leqslant t \leqslant \xi\},\\
&I_t^+ = \{i : \text{facility } i \text{ is opened during period } t\},\\
&I_A^+ = \{(i, \tau, \xi) : a_{i\tau}^{\xi} = 1\},\\
&I_R^+ = \{(i, \tau, \xi) : r_{i\tau}^{\xi} = 1\}.
\end{aligned}
$$

The set I^* corresponds to (i, τ, ξ), such that $SA_{i\tau}^{\xi}$ and/or $SR_{i\tau}^{\xi}$ are equal to zero. The set I_t^* corresponds to the facilities that can be opened during period t. A facility i belongs to set I_t^+ if it is going to be opened during period t. Sets I_t^* and I_t^+ are not necessarily equal, because the procedure will always try to open the minimum number of facilities, guaranteeing that all clients will be assigned to one operating facility in every time period. Sets I_A^+ and I_R^+ are built during the primal procedure and determine which facilities will be (re) opened, when and for how long.

The primal procedure begins by including in set I_t^+ all facilities belonging to I_t^* that are considered essential during period t.

Definition 1. A facility i is considered *essential* during period t if there is at least one client j that has to be assigned to facility i during period t. This happens if and only if $\exists j \in J : v_j^t \geqslant c_{ij}^t \wedge v_j^t < c_{i'j}^t, \forall\, i' \in I, i' \neq i$.

Non-essential facilities would only be operational during time period t if there are clients j that cannot be assigned to *essential* facilities. In this case, the procedure includes in set I_t^+ the facility i belonging to I_t^* that corresponds to the smallest cost c_{ij}^t.

Primal procedure

1. $I_A^+ = I_R^+ = \emptyset$. $I_t^+ = \emptyset, \forall t$. Build sets I^* and I_t^*. $Num = 0$.
2. For $t = 1, \ldots, T$, include in set I_t^+ all facilities i such that $\exists j : v_j^t \geqslant c_{ij}^t$ and $v_j^t < c_{i'j}^t,\ \forall\, i' \neq i$.
3. For each client j such that $v_j^t < c_{ij}^t, \forall\, i \in I_t^+$, include in set I_t^+ facility i such that $c_{ij}^t = min_{v_j^t \geqslant c_{i'j}^t} c_{i'J}^t$. $Num = Num + 1$. If $Num = 1$ then $I_A^+ = I_R^+ = \emptyset$. $I_t^* = I_t^+$ and $I_t^+ = \emptyset,\ \forall t$, go to 2. Else go to 4.
4. Build sets I_A^+ and I_R^+. Update I_t^+. For $t = 1, \ldots, T$, assign each client j to facility $i' \in I_t^+$ such that $c_{i'j}^t = min_{i \in I}\{c_{ij}^t\}$.
5. Test complementary conditions (19)–(21).

In the primal procedure, steps 4 and 5 require special attention. As a matter of fact, building sets I_A^+ and I_R^+ is much more complicated than building set I^+ as described in Dynaloc [14] taking into account the remarks of Saldanha da Gama and Captivo [24]. If, for a facility $i \in I_t^+$, $S_{i\tau}^{\xi} = 0$, $\tau \leqslant t \leqslant \xi$ for more than one pair (τ, ξ), the choice of which variable to include in set I_A^+ or I_R^+ is not trivial. To build these sets two procedures were developed. For each facility i, these procedures include in I_A^+ or I_R^+ variables that guarantee facility i will be opened at least during periods t such that $i \in I_t^+$, and that satisfy constraints (4)–(6). Defining

$$t_1 = \min\{\tau : i \in I_\tau^+\} \quad \text{and} \quad t_2 = \max\{\tau : i \in I_\tau^+\}$$

procedure 1 tries to build a solution from period t_1 forward, and procedure 2 tries to build a solution from period t_2 backwards. In step 4 of the heuristic, both procedures are executed.

Procedure 1:

```
begin = 1; time = t1; opening = true;
WHILE time ⩽ t2
    IF i ∈ I+_time THEN
        τ = begin; ξ = T; t = time; stop = false;
        WHILE τ ⩽ t and not stop
            WHILE ξ ⩾ t and not stop
                IF ∃(i, τ, ξ) ∈ I* THEN
                    IF opening THEN
                        (i, τ, ξ) → I+_1A
                        opening = false
                    ENDIF
                    ELSE (i, τ, ξ) → I+_1R ENDELSE
                    time = begin = ξ + 1
                    stop = true
                ENDIF
                ELSE ξ = ξ − 1 ENDELSE
            ENDWHILE
            τ = τ + 1; ξ = T
        ENDWHILE
        IF not stop THEN
            IF opening THEN
                (i, begin, t2) → I+_1A
                opening = false
            ENDIF
            ELSE (i, begin, t2) → I+_1R ENDELSE
            time = t2 + 1
        ENDIF
    ENDIF
    ELSE time = time + 1 ENDELSE
ENDWHILE
```

This step can be described formally as follows:

Step 4 of primal procedure:

1. $i = 1$;
2. If $\exists t : i \in I_t^+$, go to 3; else go to 8;
3. $t_1 = min\{\tau : i \in I_\tau^+\}$; $t_2 = max\{\tau : i \in I_\tau^+\}$;

4. $I_{1A}^{+} = I_{A}^{+}$; $I_{1R}^{+} = I_{R}^{+}$. Execute Procedure 1;
5. $I_{2A}^{+} = I_{A}^{+}$; $I_{2R}^{+} = I_{R}^{+}$. Execute Procedure 2;
6. $sum1 = \sum_{(i,\tau,\xi)\in I_{1A}^{+}} FA_{i\tau}^{\xi} + \sum_{(i,\tau,\xi)\in I_{1R}^{+}} FR_{i\tau}^{\xi}$; $sum2 = \sum_{(i,\tau,\xi)\in I_{2A}^{+}} FA_{i\tau}^{\xi} + \sum_{(i,\tau,\xi)\in I_{2R}^{+}} FR_{i\tau}^{\xi}$;
7. If $(sum1 < sum2)$ $I_{A}^{+} = I_{1A}^{+}$, $I_{R}^{+} = I_{1R}^{+}$; else $I_{A}^{+} = I_{2A}^{+}$, $I_{R}^{+} = I_{2R}^{+}$;
8. $i = i + 1$; if $i > m$ stop. Else go to 2.

Procedure 2:

```
end = T;  time = t2;
WHILE time ≥ t1
        IF i ∈ I+_time THEN
                τ = 1;  ξ = end;  t = time; stop = false;
                WHILE ξ ≥ t and not stop
                        WHILE τ ≤ t and not stop
                                IF ∃(i, τ, ξ) ∈ I* THEN
                                        IF τ ≤ t1 THEN (i, τ, ξ) → I+_2A ENDIF
                                        ELSE (i, τ, ξ) → I+_2R ENDELSE
                                        time = end = τ − 1
                                        stop = true
                                ENDIF
                                ELSE τ = τ + 1 ENDELSE
                        ENDWHILE
                        ξ = ξ − 1; τ = 1
                ENDWHILE
                IF not stop THEN
                        (i, t1, end) → I+_2A
                        time = t1 − 1
                ENDIF
        ENDIF
        ELSE time = time − 1 ENDELSE
ENDWHILE
```

After the execution of step 4, a primal admissible solution is found, but conditions (19)–(23) can be violated. As can be seen from procedures 1 and 2, for each facility i there will be at most one violation of conditions (22) or (23). Step 5 of the primal procedure tries to decrease the number of violations of conditions (19)–(21). These conditions imply that:

- If $\rho_i \neq 0$, then facility i has to be opened during one or more time periods.
- If $\pi_i^t \neq 0$, then facility i has to be operating during period t.
- If $u_i^t \neq 0$, then, facility i has to be reopened at the beginning of period t, if it had already been opened and closed before t.

Proposition 1. *If ρ_i, π_i^t and u_i^t are all greater than zero then, if the complementary conditions* (19) *are satisfied, conditions* (20) *and* (21) *will also be satisfied.*

Proposition 2. *If ρ_i and π_i^t are simultaneously greater than zero and conditions* (21) *are satisfied, then conditions* (20) *will also be satisfied.*

These propositions follow directly from the definition of the complementary conditions (19)–(21), and are used in step 5 of the primal procedure.

Step 5 of the Primal Procedure

```
i = 1
WHILE i ⩽ m
        ρverified = false;  πverified = false;  t = 1;
        WHILE t ⩽ T
                IF u_i^t ≠ 0 and ∃(i, τ, ξ) ∈ I_A^+, ξ < t THEN
                        IF ∄(i, t, ξ) ∈ I_R^+ THEN
                                tmax = min {τ : (i, τ, γ) ∈ I_R^+ and τ > t}
                                tmax = min {T,  tmax}
                                IF ∃(i, t, ξ) ∈ I*, ξ < tmax THEN
                                        (i, t, ξ) → I_R^+
                                        ρverified = true; πverified = true
                                ENDIF
                        ENDIF
                        ELSE ρverified = true; πverified = true ENDELSE
                ENDIF
                IF π_i^t = 0 and πverified = false THEN
                        IF i ∉ I_t^+ THEN
                                IF ∃(i, τ, ξ) ∈ I_A^+ THEN
                                        tmin = max {ξ : (i, τ, ξ) ∈ I_A^+ ∪ I_R^+, ξ < t}
                                        tmax = min {τ : (i, τ, ξ) ∈ I_A^+ ∪ I_R^+, τ > t}
                                ENDIF
                                ELSE tmin = 1; tmax = T ENDELSE
                                IF ∃(i, τ, ξ) ∈ I* : τ ⩾ tmin and ξ ⩽ tmax THEN
                                        (i, τ, ξ) → I_A^+ or I_R^+
                                        ρverified = false
                                ENDIF
                        ENDIF
                        ELSE ρverified = true ENDELSE
                ENDIF
        ENDWHILE
        IF ρverified = false and ρ_i ≠ 0 THEN
                Choose (i, τ, ξ) ∈ I*
                (i, τ, ξ) → I_A^+
        ENDIF
ENDWHILE
```

4.4. Dual adjustment procedure for variables ρ_i

If it is possible for a variable ρ_i to decrease its value, the dual objective function value will automatically increase. The value of variable ρ_i can be decreased if $SA_{i\tau}^{\xi} \neq 0, \forall 1 \leqslant \tau \leqslant \xi$.

Increasing the value of the dual variable ρ_i, increases all slacks $SA_{i\tau}^{\xi}$. The change in these slacks allows the increase of some v_j^t that were blocked. However, variables ρ_i have a coefficient of minus one in the dual objective function. Therefore, they should only be increased if the enhancement of variables v_j^t is compensatory. It should be noted that it is worth trying to increase ρ_i only if $SR_{i\tau}^{\xi} \neq 0$ and $SA_{i\tau}^{\xi} = 0$. Otherwise, a change in the slack $SA_{i\tau}^{\xi}$ would not be reflected in dual variables v_j^t.

Dual adjustment procedure for dual variables ρ_i

Consider $I_j^{t^*} = \{i : \exists(\tau, \xi) \text{ with } \tau \leqslant t \leqslant \xi | (i, \tau, \xi) \in I^* \text{ and } v_j^t \geqslant c_{ij}^t\}$.

1. $i \leftarrow 1$;
2. $\Delta\rho_i \leftarrow min_{\tau \leqslant \xi}\{SA_{i\tau}^{\xi}\}$. If $\Delta\rho_i = 0$ then go to 3. Else go to 7.
3. $\Delta\rho_i = max\{SR_{i\tau}^{\xi} : \exists(i, \tau, \xi) \in I_R^+ \text{ with } SA_{i\tau}^{\xi} = 0 \text{ and } SR_{i\tau}^{\xi} \neq 0\}$.
4. If $\Delta\rho_i \neq 0$ then $\rho_i \leftarrow \rho_i + \Delta\rho_i$; $SA_{i\tau}^{\xi} \leftarrow SA_{i\tau}^{\xi} + \Delta\rho_i, \forall\tau, \xi \geqslant \tau$. Else go to 8.
5. $J^+ = \{(j, t) : I_j^{t^*} = \{i\}, \ \forall t\}$. Execute the dual ascent procedure for dual variables $v_j^t \cdot J^+ = J$. Execute the dual ascent procedure for dual variables v_j^t.
6. $\Delta\rho_i = \min\limits_{\substack{\tau \\ \xi \geqslant \tau}} SA_{i\tau}^{\xi}$.
7. $\Delta\rho_i = min\{\Delta\rho_i, \rho_i\}$. If $\Delta\rho_i \neq 0$ then $SA_{i\tau}^{\xi} \leftarrow SA_{i\tau}^{\xi} - \Delta\rho_i, \forall\tau, \xi \geqslant \tau$; $\rho_i \leftarrow \rho_i - \Delta\rho_i$.
8. If $i = \#I$ then stop. Else $i \leftarrow i + 1$; go to 2.

Proposition 3. *The dual adjustment procedure for dual variables ρ_i cannot worsen* CD-DLPOCR *objective function value.*

Proof. Decreasing the value of dual variable ρ_i, keeping the dual solution admissible, constitutes an improvement in the dual objective function value (because variable ρ_i has a coefficient of minus one in the objective function).

Increasing the value of ρ_i, increases the value of some slacks. Consider that Δv_j^t is the change in the value of variable v_j^t after the execution of the dual ascent procedure in step 5. Let $\varDelta_1\rho_i$ be the increase calculated in step 4, $\varDelta_2\rho_i$ the decrease calculated in step 7 and $\Delta\rho_i = \varDelta_1\rho_i - \varDelta_2\rho_i$. Consider $SA_{i\tau}^{'\xi}$ as the slack resulting from the execution of the dual adjustment procedure. It should be noted that $\varDelta_1\rho_i \geqslant \varDelta_2\rho_i$. To check that this inequality always holds, consider slack $SA_{i\tau}^{'\xi}$ that was equal to zero and that was increased by $\varDelta_1\rho_i$. During the execution of the dual ascent procedure (step 5) this slack can only decrease, therefore, $\varDelta_2\rho_i \leqslant SA_{i\tau}^{'\xi} \leqslant \varDelta_1\rho_i$.

After the procedure execution, two situations can arise:

1. If there is some slack $SA_{i\tau}^{'\xi}$ that has returned to its original value, it means that the increase in variable ρ_i was totally used by the dual ascent procedure for the increase in dual variables v_j^t,

so $\sum_j \sum_t \Delta v_j^t \geqslant \Delta_1 \rho_i$: $SA'^{\xi}_{i\tau} = SA^{\xi}_{i\tau} + \Delta\rho_i - \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t$, therefore if $SA'^{\xi}_{i\tau} = SA^{\xi}_{i\tau}$ then $\Delta\rho_i - \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t = 0$.

2. If $SA'^{\xi}_{i\tau} = SA^{\xi}_{i\tau}$ for all slacks, then it means that they are all greater than zero, and the procedure will decrease the value of the dual variable ρ_i. Consider slack $SA'^{\xi}_{i\tau}$, with $v_j^t \geqslant c_{ij}^t$, that after step 6 was changed to $SA'^{\xi}_{i\tau} = SA^{\xi}_{i\tau} + \Delta_1\rho_i - \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t$. As $SA'^{\xi}_{i\tau} > SA^{\xi}_{i\tau}$, this means that: $\Delta_1\rho_i - \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t > 0 \Leftrightarrow \Delta_1\rho_i > \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t$. If$\rho_i$ kept its value, the objective function value would diminish. However, this variable will be decreased by $\Delta_2\rho_i$, calculated as in step 7. This means that at least one slack will be equal to zero, so the increase in variable ρ_i will be compensated by the increase in the dual variables v_j^t:

$$SA'^{\xi}_{i\tau} = SA^{\xi}_{i\tau} + \Delta_1\rho_i - \Delta_2\rho_i - \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t = 0.$$

As $SA^{\xi}_{i\tau} \geqslant 0$ then: $\Delta_1\rho_i - \Delta_2\rho_i - \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t \leqslant 0 \Leftrightarrow \Delta_1\rho_i - \Delta_2\rho_i \leqslant \sum_j \sum_{t=\tau}^{\xi} \Delta v_j^t$.

In the worst case, the decrease in step 7 is equal to the increase in step 4, and the dual objective function does not change.

4.5. *Dual-primal adjustment procedure for variables v_j^t*

The dual-primal adjustment procedure for variables v_j^t detects violations of the complementary conditions (18), and decreases the values of some variables v_j^t, allowing other variables v_j^t to increase. This procedure can reduce the number of violations of complementary conditions (18) and, at the same time, can improve the value of the dual objective function.

Consider the following sets:

$$\begin{aligned}
I_j^{t^*} &= \{i : \exists(\tau, \xi) \text{ with } \tau \leqslant t \leqslant \xi | (i, \tau, \xi) \in I^* \text{ and } v_j^t \geqslant c_{ij}^t\},\\
I_j^{t+} &= \{i : i \in I_t^+ \text{ and } v_j^t > c_{ij}^t\},\\
J_i^{t+} &= \{(j, \tau) : I_j^{\tau^*} = \{i\} \text{ and } (i, \gamma, \xi) \notin I^*, \gamma \leqslant \xi \leqslant \tau < t \text{ or } t < \tau \leqslant \gamma \leqslant \xi\},\\
c_j^{t-} &= \max_i \{c_{ij}^t : v_j^t > c_{ij}^t\}.
\end{aligned}$$

The set I_j^{t+} indicates, for each client j, all operating facilities during period t such that v_j^t is greater than the assignment cost c_{ij}^t. A violation of the complementary condition (18) is detected by the existence of, at least, one pair (j, t) such that the number of elements in I_j^{t+} is greater than one. Decreasing the value of a variable v_j^t such that the number of elements in I_j^{t+} is greater than one, means that at least slacks $S_{i\tau}^{\xi}$, $\tau \leqslant t \leqslant \xi$, will be increased for two distinct facilities. This may promote the increase in the dual objective function. The set J_i^{t+} represents all variables v_j^t whose value can be increased with the rise of slack $S_{i\tau}^{\xi}$, $\tau \leqslant t \leqslant \xi$. This procedure is based on the works of Erlenkotter [3] and Van Roy and Erlenkotter [14] taking into account the remarks of Saldanha da Gama and Captivo [24].

4.6. Dual ascent procedure for variables u_i^t

Increasing variables u_i^t, increases slacks SR_{it}^{ξ}, $\xi \geqslant t$, but at the same time diminishes slacks $SA_{i\tau}^{\xi}$, $\tau \leqslant \xi < t$. If the procedure is able to increase slacks $S_{i\tau}^{\xi}$ that are blocking variables v_j^t, decreasing $S_{i\tau}^{\xi}$ that are not blocking any variable v_j^t, then it will be possible to improve the dual objective function value.

If there is $SR_{it}^{\xi}=0$ and $SA_{it}^{\xi} \neq 0$, then the increase in u_i^t can be of help. This situation occurs, for instance, when $(i,t,\xi) \in I_A^+$ with $SA_{it}^{\xi} \neq 0$ and $SR_{it}^{\xi}=0$. In this case, $SR_{it}^{\xi}=0$ should not increase more than $SA_{it}^{\xi} - SR_{it}^{\xi}$, because any further increase will not change the value of S_{it}^{ξ}. On the other hand, variable u_i^t cannot grow more than the minimum value of $SA_{i\tau}^{\xi}$, $\forall \tau \leqslant \xi < t$, so that the dual solution remains admissible. Increasing variable u_i^t can diminish the number of violations of complementary conditions (22).

Consider variables u_i^t organized as a sequence of pairs (i,t).

Dual ascent procedure for dual variables u_i^t

1. Initialize $(i,t) \leftarrow (i,t)_1$; $q \leftarrow 1$.
2. $\xi \leftarrow t$; $\Delta u_i^t \leftarrow 0$; $\delta \leftarrow 0$.
3. If $SR_{it}^{\xi}=0$ and $SA_{it}^{\xi} \neq 0$, then $\Delta u_i^t \leftarrow max\{\Delta u_i^t, SA_{it}^{\xi}\}$ and $\delta \leftarrow 1$.
4. If $\xi = T$ go to 5, else $\xi \leftarrow \xi + 1$, go to 3.
5. If $\delta = 0$, go to 7. Else $\Delta u_i^t \leftarrow min\{\Delta u_i^t, min_{\tau \leqslant \gamma < t} SA_{i\tau}^{\gamma}\}$, $SR_{it}^{\xi} \leftarrow SR_{it}^{\xi} + \Delta u_i^t, \forall \xi \geqslant t$. $SA_{i\tau}^{\xi} \leftarrow SA_{i\tau}^{\xi} - \Delta u_i^t, \forall \tau \leqslant \xi < t$ and $u_i^t \leftarrow u_i^t + \Delta u_i^t$.
6. $J^+ = \{(j,t) : I_j^{t*} = \{i\}, \forall t\}$. Execute the dual ascent procedure for variables v_j^t. $J^+ = J$. Execute the dual ascent procedure for variables v_j^t.
7. If $q = \#I \times T$ then stop. Else $q \leftarrow q+1$; $(i,t) \leftarrow (i,t)_q$, go to 2.

4.7. Dual descent procedure for variables u_i^t

Decreasing u_i^t will decrease slacks SR_{it}^{ξ}, $\xi \geqslant t$, and increase slacks $SA_{i\tau}^{\xi}$, $\tau \leqslant \xi < t$. To guarantee the admissibility of the dual solution, variable u_i^t can only be decreased if $\mathrm{SR}_{it}^{\xi} > 0, \forall \xi \geqslant t$. If the procedure is able to increase slacks $S_{i\tau}^{\xi}$ that are blocking dual variables v_j^t and decrease slacks that does not influence v_j^t values, then it is possible to improve the dual objective function value.

Dual descent procedure for variables u_i^t

1. Initialize $(i,t) \leftarrow (i,t)_1$; $q \leftarrow 1$.
2. If $u_i^t = 0$ go to 6; Otherwise, $\Delta u_i^t \leftarrow 0$; $\delta \leftarrow 0$.
3. If $SR_{it}^{\xi} > 0, \forall \xi \geqslant t$, then $\Delta u_i^t \leftarrow min_{\xi \geqslant t}\{SR_{it}^{\xi}\}$ and $\delta \leftarrow 1$.
4. If $\delta = 0$ go to 6. Else $\Delta u_i^t \leftarrow min\{\Delta u_i^t, u_i^t\}$; $SR_{it}^{\xi} \leftarrow SR_{it}^{\xi} - \Delta u_i^t, \forall \xi \geqslant t$. $SA_{i\tau}^{\xi} \leftarrow SA_{i\tau}^{\xi} + \Delta u_i^t, \forall \tau \leqslant \xi < t$ and $u_i^t \leftarrow u_i^t - \Delta u_i^t$.
5. $J^+ = \{(j,t) : I_j^{t^*} = \{i\}, \forall t\}$. Execute the dual ascent procedure for variables v_j^t. $J^+ = J$. Execute the dual ascent procedure for variables v_j^t.
6. If $q = \#I \times T$ then stop. Else $q \leftarrow q+1$; $(i,t) \leftarrow (i,t)_q$, go to 2.

4.8. Dual adjustment procedure for variables π_i^t

Increasing the value of π_i^t will increase slacks $S_{i\tau}^{\xi}$, $\tau \leqslant t \leqslant \xi$. If there are slacks $S_{i\tau}^{\xi}$, $\tau \leqslant t \leqslant \xi$ that are blocking dual variables v_j^t, then it is possible to improve the value of the dual objective function. However, it is only worth to increase π_i^t if the change in dual variables v_j^t compensates the loss of π_i^t in the objective function value (the variable π_i^t has a coefficient of minus one). If the procedure is able to diminish the value of π_i^t, maintaining the dual solution feasibility, then there is an immediate improvement in the dual objective function value.

Consider variables π_i^t organized as a sequence of pairs (i, t), and M a large positive number.

Dual adjustment procedure for variables π_i^t

1. Initialize $(i, t) \leftarrow (i, t)_1$; $q \leftarrow 1$.
2. $\Delta\pi_i^t = min_{\tau \leqslant t \leqslant \xi} S_{i\tau}^{\xi}$. If $\Delta\pi_i^t \neq 0$, then go to 6. Else $\Delta\pi_i^t \leftarrow M$.
3. $S_{i\tau}^{\xi} \leftarrow S_{i\tau}^{\xi} + \Delta\pi_i^t, \forall \tau \leqslant t \leqslant \xi$; $\pi_i^t = \pi_i^t + \Delta\pi_i^t$.
4. $J^+ = \{(j, t) : I_j^{t^*} = \{i\}, \forall t\}$. Execute the dual ascent procedure for variables v_j^t. $J^+ = J$. Execute the dual ascent procedure for variables v_j^t.
5. $\Delta\pi_i^t = min_{\tau \leqslant t \leqslant \xi} S_{i\tau}^{\xi}$.
6. $\Delta\pi_i^t = min\{\Delta\pi_i^t, \pi_i^t\}$. If $\Delta\pi_i^t \neq 0$ then $S_{i\tau}^{\xi} \leftarrow S_{i\tau}^{\xi} - \Delta\pi_i^t, \forall\ \tau \leqslant t \leqslant \xi$ and $\pi_i^t = \pi_i^t - \Delta\pi_i^t$.
7. If $q = \#I \times T$ then stop. Else $q \leftarrow q + 1$; $(i, t) \leftarrow (i, t)_q$, go to 2.

5. Branch-and-bound procedure

A branch-and-bound procedure was developed that guarantees the calculation of the optimum solution to DLPOCR, whenever the primal-dual heuristic described cannot find it. In this procedure, variables are always fixed first to zero and then to one. The tree is searched using a depth search procedure. At every node of the tree, the primal-dual heuristic is executed. The variable to be fixed is chosen according to the complementary conditions that are being violated by the current pair of primal and dual solutions, and according to the following order:

1. If $a_{i\tau}^{\xi} = 1$ and $SA_{i\tau}^{\xi} \neq 0$, then choose variable $r_{i\tau}^{\xi}$.
2. If $r_{i\tau}^{\xi} = 1$ and $SR_{i\tau}^{\xi} \neq 0$, then choose variable $a_{i\tau}^{\xi}$.
3. If $u_i^t \neq 0$ and $r_{it}^{\xi} = 0, \forall \xi \geqslant t$, then choose variable r_{it}^{ξ} such that $SR_{i\tau}^{\xi} = 0$.
4. If $\pi_i^t \neq 0$ and $a_{i\tau}^{\xi} = r_{i\tau}^{\xi} = 0, \forall \tau \leqslant t \leqslant \xi$, then choose variable $a_{i\tau}^{\xi}$ or $r_{i\tau}^{\xi}$ such that $S_{i\tau}^{\xi} = 0$.
5. If $v_j^t > c_{ij}^t$, for more than one facility $i \in I^+$, then choose variable $a_{i\tau}^{\xi} \in I_A^+$ or $r_{i\tau}^{\xi} \in I_R^+, \tau \leqslant t \leqslant \xi$.

The order presented was the one that gave the best results in the preliminary test problems solved. Variables are fixed to one or to zero using procedures similar to the ones described in [3]. To fix one variable to zero, its fixed cost is changed to $+ \propto$. This will not put at risk the admissibility of the dual solution. To fix the variable to one, its fixed cost is changed to zero. This will impose some changes in the dual solution that, in general, becomes inadmissible. If variable $a_{i\tau}^{\xi}$ or $r_{i\tau}^{\xi}$ is fixed to one, then all dual variables v_j^t such that $v_j^t > c_{ij}^t$, $\tau \leqslant t \leqslant \xi$, will have to be changed to $v_j^t = c_{ij}^t$ (and consequently slacks will have to be increased). This causes deterioration in the value of the dual objective function value, that is compensated by summing up to this value the fixed cost of the variable that was fixed to one.

Using a heuristic, to calculate a primal solution at every node of the branch-and-bound tree, has a major disadvantage: the primal solution calculated might not be optimum, so the node can be fathomed only if one of the following conditions holds:

1. The problem is infeasible.
2. The dual objective function value equals the primal objective function value, meaning the optimal solution of the current node has been found.
3. The current dual objective function value is worse than the objective function value of the best primal solution found thus far.

6. Computational tests

6.1. Description of the computational experiments

The primal-dual heuristic and the branch-and-bound procedure were tested with a set of randomly generated problems. The following values for m, n and T were considered and, for each combination, five problems were generated (total of 360 problems):

n	25	50	100	200	500	1000
m	5	10	50	100		
T	5	20	50			

The test problems were generated according to the following procedure:

1. Random generation of (x, y) coordinates in the plane, according to a uniform distribution and considering a 500×500 square. These coordinates correspond to the location of the $m + n$ nodes of the network.
2. Random creation of arcs between the network nodes, considering a probability of 75%.
3. Creation of arcs (not created in step 2) between nodes such that the Euclidean distance from one another is less than 50, with probability of 80%.
4. Generation of costs associated with arcs: for the first period, the costs are randomly generated according to a uniform distribution, in the interval [100,1100]. For $t > 1$, the cost associated to the arc in period t is equal to the cost in $t - 1$ plus a changing factor randomly generated corresponding to a variation between -10% and $+10\%$.
5. For each time period, calculation of the shortest path between each client and each facility, using the Floyd–Warshall algorithm [25].
6. For each facility i and period t, consider $tend = t, \ldots, T$. For $tend = t$, random generation of fixed costs for variables a_{it}^{tend} and r_{it}^{tend} according to a uniform distribution in the interval [500,3500]. For $tend > t$, random generation of a factor between 0% and 10% that represents an increase in the fixed cost for $tend - 1$.

All test problems, as well as the source code and executable file for the generation algorithm, are available upon request from the authors. All experiments were carried out in a Pentium 4, 1.80 Ghz, running under Windows 2000 operating system, with a maximum of 2000 MB of virtual memory and 260 kb of Ram.

Both heuristic and branch-and-bound procedure were programmed using C-language and Borland-C++ compiler (version 5.0). The performance of these two algorithms was compared with the performance of Cplex, version 7.0.

The branch-and-bound procedure is terminated whenever the branch-and-bound tree reaches level 25,000, or when the execution time exceeds 175,000 s. Cplex terminates without calculating the optimal solution whenever more than 2,100,000,000 nodes of the branch-and-bound tree are explored, or when the number of simplex iterations in a node exceeds 2,100,000,000, or when there is not enough memory to read the problem.

After the execution of the primal-dual heuristic, a local search procedure was executed. Consider the following notation:

$SOL_S =$ set of solutions constituting the k-neighborhood of solution S.

$Z_S =$ primal objective function value considering solution S.

Definition 2. An admissible solution S' to DLPOCR is said to be in the k-neighborhood of the admissible solution S if and only if S' differs from S by the insertion or removal of at most k functioning continuous time periods to a facility i.

The local search procedure can be described as follows:

Local search procedure

1. $k \leftarrow 1$. $S =$ current primal solution.
2. Calculate $S^+ \in SOL_S$ such that $Z_{S^+} = min_{S' \in SOL_s}\{Z_{S'}\}$.
3. If $Z_{S^+} < Z_S$, then $S \leftarrow S^+$ and go to 2. Else go to 4.
4. $k \leftarrow k + 1$. If $k > T$ then stop. Else go to 2.

6.2. Computational results

The computational results obtained considering T equal to 5 will be omitted because the computational times spent by the primal-dual heuristics are less than 15 s, and the computational results obtained considering T equal to 20 and 50 are representative of the behavior of the heuristic when compared to Cplex and branch-and-bound. These results are, however, available from the authors upon request. Table 1 shows average results of the quality of the primal solutions obtained by the primal-dual heuristic, and after the execution of the local search procedure around the best solution found by the heuristic. For each set (n, m, T), five problems were generated and solved by the heuristic. The table shows the worst, the best and the average value of the deviations of the best primal solution found from a known lower bound on the optimal value. This lower bound is equal to the optimum value for all problems Cplex or the branch-and-bound algorithm were able to solve. For all the others, this lower bound is given by the best dual solution found by the primal-dual heuristic. The values shown are calculated, in percentage, as $(Z - Z_{\mathrm{LB}})/Z_{\mathrm{LB}}$, where Z is the objective function value of the best primal solution found and Z_{LB} is the value of the lower bound.

Table 1
Quality of the primal solution (in percentage)

T	m	n	Primal-dual heuristic			Primal-dual heuristic + local search procedure			T	m	n	Primal-dual heuristic			Primal-dual heuristic + local search procedure		
			Best	Average	Worst	Best	Average	Worst				Best	Average	Worst	Best	Average	Worst
20	5	25	0.00	0.01	0.04	0.00	0.00	0.00	50	5	25	0.00	6.34	15.89	0.00	2.70	8.06
20	5	50	0.00	0.49	1.99	0.00	0.27	1.35	50	5	50	1.98	5.74	10.75	0.13	1.53	5.19
20	5	100	0.00	1.05	3.78	0.00	0.22	1.08	50	5	100	2.33	4.40	7.02	0.00	1.12	4.37
20	5	200	0.00	0.36	0.89	0.00	0.02	0.05	50	5	200	0.22	3.26	9.83	0.00	0.39	0.90
20	5	500	0.00	0.54	0.88	0.00	0.00	0.00	50	5	500	0.33	1.79	3.34	0.00	0.57	1.79
20	5	1000	0.00	0.19	0.38	0.00	0.01	0.04	50	5	1000	0.39	1.55	3.62	0.02	0.49	1.53
20	10	25	0.00	0.54	2.70	0.00	0.01	0.03	50	10	25	6.71	9.52	11.93	0.65	1.66	2.18
20	10	50	0.00	0.44	1.44	0.00	0.16	0.79	50	10	50	0.74	9.59	27.41	0.00	0.70	1.79
20	10	100	0.00	1.26	3.30	0.00	0.00	0.00	50	10	100	0.64	3.27	4.92	0.22	1.08	2.81
20	10	200	0.00	0.77	1.81	0.00	0.05	0.09	50	10	200	4.03	8.03	19.7	0.60	2.16	5.07
20	10	500	0.00	0.08	0.27	0.00	0.02	0.10	50	10	500	0.53	2.40	4.61	0.03	0.58	2.10
20	10	1000	0.00	0.20	0.69	0.00	0.02	0.06	50	10	1000	0.24	2.12	4.38	0.06	0.90	3.02
20	50	25	0.00	1.52	4.61	0.00	0.00	0.00	50	50	25	8.32	12.47	17.01	2.33	4.64	6.09
20	50	50	0.53	1.91	3.66	0.00	1.13	3.16	50	50	50	5.01	8.15	12.81	0.40	3.04	7.66
20	50	100	0.48	1.93	4.06	0.00	0.32	1.20	50	50	100	1.30	5.70	12.46	0.88	2.53	3.52
20	50	200	0.59	2.52	5.24	0.00	1.86	3.45	50	50	200	3.41	5.30	7.26	1.16	2.21	4.72
20	50	500	0.22	0.90	2.39	0.00	0.39	1.34	50	50	500	2.36	4.19	6.99	0.17	1.17	1.81
20	50	1000	1.21	1.74	2.26	0.67	1.17	1.98	50	50	1000	1.40	2.40	2.85	0.39	1.07	1.80
20	100	25	0.39	2.09	5.40	0.00	0.23	0.93	50	100	25	4.81	14.32	40.14	2.27	3.51	6.73
20	100	50	0.90	1.99	3.57	0.00	0.72	2.48	50	100	50	4.91	9.88	22.98	1.13	2.44	4.20
20	100	100	0.42	2.33	4.51	0.00	0.32	0.54	50	100	100	1.65	5.78	9.93	0.85	2.73	7.99
20	100	200	0.15	1.51	2.79	0.00	0.22	0.53	50	100	200	3.33	7.19	8.87	0.73	3.04	7.75
20	100	500	2.04	2.58	3.49	0.80	1.50	2.24	50	100	500	2.50	4.33	5.83	1.17	2.06	2.50
20	100	1000	2.38	2.82	3.05	2.08	2.62	2.96	50	100	1000	3.44	4.83	7.06	2.03	2.53	3.16
Average results			0.39	1.24	2.63	0.15	0.47	1.02	Average results			2.52	5.94	11.57	0.63	1.87	4.03

Table 2 shows, in percentage, the quality of the lower bound on the optimal objective function value calculated by the primal-dual heuristic. The quality of the lower bound is calculated as $(Z^* - Z_{\mathrm{LB}})/Z^*$, where Z^* represents the best objective function value known.

Table 3 compares the execution times in seconds of the primal-dual heuristic with and without the execution of the local search procedure, and Table 4 shows the execution times of Cplex and of the branch-and-bound procedure.

The heuristic was able to find primal solutions for all of the test problems. Problems that neither Cplex nor the branch-and-bound procedure were able to solve are not considered in Tables 2 and 4. The symbol '–' means that the corresponding procedure was not able to solve any of the 5 problems. For instances with $(T, m, n) = (5, 100, 500)$ and $(20, 100, 50)$, Cplex was not able to solve one of the five problems. For instances with $(T, m, n) = (20, 50, 500)$, the branch-and-bound procedure was not capable of solving one of the five problems. In these cases, the values presented consider only four executiontimes.

Table 2
Quality of the lower bound

T	m	n	Lower bound		
			Best	Average	Worst
20	5	25	0.00	0.08	0.39
20	5	50	0.00	0.01	0.03
20	5	100	0.00	0.41	1.24
20	5	200	0.00	0.00	0.00
20	5	500	0.00	0.00	0.00
20	5	1000	0.00	0.00	0.00
20	10	25	0.00	0.01	0.06
20	10	50	0.00	0.13	0.63
20	10	100	0.00	0.03	0.10
20	10	200	0.00	0.07	0.28
20	10	500	0.00	0.01	0.04
20	10	1000	0.00	0.00	0.00
20	50	25	0.00	0.28	1.06
20	50	50	0.41	1.05	1.59
20	50	100	0.14	0.45	0.81
20	50	200	0.46	0.79	1.58
20	50	500	0.48	0.56	0.61
20	100	25	0.20	1.20	3.06
20	100	50	1.05	1.35	2.16
20	100	100	0.49	0.98	1.53
20	100	200	0.52	1.03	1.84
Average results			0.18	0.40	0.81
50	5	25	0.00	0.11	0.24
50	5	50	0.04	0.21	0.29
50	5	100	0.00	0.37	0.92
50	10	25	0.00	0.15	0.48
50	10	50	0.00	0.72	2.03
50	10	100	0.20	0.84	2.11
50	10	200	0.01	0.21	0.49
Average results			0.04	0.37	0.94

6.3. Conclusions

The analysis of the computational results allows drawing some conclusions:

1. The primal-dual heuristic is an efficient procedure to calculate admissible solutions to DLPOCR. The heuristic is able to calculate good-quality solutions in reasonable computational times even for very large instances of the problem. The average deviation from the best-known lower bound is 2.59%.
2. The execution of the local search procedure after the primal-dual heuristic is worthwhile, because the additional computational time needed can result in a significant improvement in the quality of the best primal solution found. The average deviation from the best-known lower bound is 0.86%.

Table 3
Primal-dual heuristic computational times in seconds

T	m	n	Primal-dual heuristic			Primal-dual heuristic + local search			T	m	n	Primal-dual heuristic			Primal-dual heuristic + local search		
			Best	Average	Worst	Best	Average	Worst				Best	Average	Worst	Best	Average	Worst
20	5	25	0.0	0.0	0.0	0.0	0.0	0.0	50	5	25	0.3	0.3	0.5	0.3	0.5	0.7
20	5	50	0.0	0.1	0.1	0.0	0.1	0.1	50	5	50	0.9	1.0	1.1	1.4	1.4	1.5
20	5	100	0.1	0.1	0.1	0.1	0.2	0.2	50	5	100	1.7	1.8	1.9	2.5	2.6	2.7
20	5	200	0.2	0.3	0.3	0.2	0.3	0.4	50	5	200	3.8	4.0	4.2	4.4	5.8	6.9
20	5	500	0.6	0.9	1.3	0.6	1.0	1.6	50	5	500	8.9	10.2	11.8	9.6	14.9	20.3
20	5	1000	1.2	1.9	2.4	1.2	2.1	2.9	50	5	1000	18.6	21.13	22.8	31.5	33.6	37.6
20	10	25	0.0	0.0	0.1	0.0	0.0	0.1	50	10	25	1.3	1.6	1.8	2.0	2.2	2.5
20	10	50	0.1	0.2	0.2	0.1	0.2	0.3	50	10	50	3.7	4.0	4.5	3.9	5.1	5.6
20	10	100	0.4	0.5	0.6	0.5	0.6	0.8	50	10	100	7.8	8.4	9.6	10.0	10.7	12.1
20	10	200	1.0	1.1	1.2	1.0	1.4	1.5	50	10	200	15.6	16.2	17.0	23.6	24.8	25.5
20	10	500	2.6	3.0	3.6	2.6	3.4	4.3	50	10	500	40.3	42.4	44.8	58.7	62.3	66.0
20	10	1000	5.3	6.0	6.7	5.3	8.0	10.6	50	10	1000	80.5	81.4	83.2	111.5	126.6	138.5
20	50	25	0.6	0.7	1.0	0.7	0.9	1.2	50	50	25	57.0	69.1	72.9	60.6	73.4	77.5
20	50	50	3.7	4.7	5.2	4.1	5.0	5.5	50	50	50	168.2	172.5	178.1	183.8	187.9	193.7
20	50	100	13.7	13.9	14.3	15.0	15.5	16.1	50	50	100	359.8	367.8	375.6	401.4	419.1	449.4
20	50	200	30.5	30.9	31.6	34.6	37.2	40.4	50	50	200	736.7	744.2	750.4	814.4	871.4	896.1
20	50	500	78.8	80.7	82.0	92.8	101.6	108.6	50	50	500	1852.8	1864.8	1876.1	2122.9	2241.6	2387.5
20	50	1000	158.6	160.8	163.0	186.1	201.5	215.6	50	50	1000	2746.2	3535.1	3756.5	3746.9	4276.2	4472.1
20	100	25	2.8	3.2	3.5	3.4	3.6	3.8	50	100	25	283.9	307.5	329.8	298.6	320.8	340.1
20	100	50	17.4	21.0	25.5	18.2	21.9	26.3	50	100	50	755.4	773.6	792.6	780.9	805.2	837.3
20	100	100	56.8	57.8	59.9	59.4	61.5	63.7	50	100	100	1611.2	1630.0	1638.5	1695.6	1739.5	1790.3
20	100	200	132.0	136.5	140.7	149.2	153.1	157.4	50	100	200	3302.0	3314.0	3323.4	3465.1	3600.6	3669.0
20	100	500	334.8	337.5	339.7	371.6	397.9	415.5	50	100	500	8298.7	8325.9	8355.6	8731.9	9026.7	9392.4
20	100	1000	680.9	705.7	721.0	730.0	774.1	848.0	50	100	1000	16691.7	16735.7	16768.7	17532.7	18177.2	18580.3

Table 4
Cplex and branch-and-bound execution times in seconds

T	m	n	Cplex			Branch-and-bound		
			Best	Average	Worst	Best	Average	Worst
20	5	25	1.1	1.2	1.4	0.0	0.3	1.7
20	5	50	2.2	2.3	2.3	0.0	1.8	8.5
20	5	100	4.8	28.7	123.0	0.1	2.5	9.0
20	5	200	13.0	15.7	22.6	0.3	3.3	11.8
20	5	500	476.0	563.6	790.4	1.5	37.0	85.2
20	5	1000	7531.3	7821.6	8479.0	1.4	26.1	95.2
20	10	25	2.6	2.9	3.9	0.0	0.0	0.1
20	10	50	5.8	6.0	6.1	0.2	2.3	6.7
20	10	100	13.5	14.2	14.6	0.6	2.5	6.4
20	10	200	174.8	271.3	541.6	1.3	8.9	17.4
20	10	500	9019.7	9379.2	9690.0	3.3	21.4	84.9
20	10	1000	—	—	—	6.6	41.8	128.7
20	50	25	48.5	118.2	345.9	2.7	22.4	98.9
20	50	50	619.6	1881.2	3163.7	6.1	79.2	240.6
20	50	100	9160.8	9657.4	9891.8	26.0	154.1	386.7
20	50	200	—	—	—	174.4	10218.9	38633.6
20	50	500	—	—	—	1335.2	24601.9	81552.6
20	100	25	618.4	1574.3	3349.5	13.0	44.5	81.1
20	100	50	9325.1	14092.7	20264.3	64.2	5202.9	20157.0
20	100	100	—	—	—	164.6	597.0	1254.8
20	100	200	—	—	—	9323.7	52291.0	173287.4
50	5	25	115.6	152.9	186.0	0.3	30.8	113.6
50	5	50	3560.2	3994.2	4898.3	1.7	2211.7	8148.3
50	5	100	—	—	—	579.1	1168.6	1819.8
50	10	25	3913.2	4070.8	4242.4	32.1	2343.9	4377.2
50	10	50	—	—	—	432.3	9585.0	21667.7
50	10	100	—	—	—	142.6	6248.5	13398.5
50	10	200	—	—	—	2391.2	21833.1	46346.8

3. The lower bounds calculated by the heuristic are, in general, very tight. The average deviation from the optimal solution is 0.43%, and the worst deviation is 4.62%.
4. Cplex is unable to solve large instances of the problem.
5. In 86% of the problems solved by Cplex, the optimal solution of DLPOCR linear relaxation was an integer solution.
6. The branch-and-bound algorithm is, in average, more efficient than Cplex. Nevertheless, it needs to build branch-and-bound trees with much more nodes than Cplex. The branch-and-bound procedure has more difficulties solving problems with large n values. This is due to the fact that larger n values lead to an increasing number of violations of complementary conditions (18). The branch-and-bound algorithm is, in general, capable of solving larger instances of the problem than Cplex.

In what concerns the characteristics of the DLPOCR solutions calculated, they can be very different, even for problems of the same size. Nevertheless, it is possible to point out some observations:

1. It is in problems with a large number of periods ($T \geqslant 20$) and few facility locations ($m \leqslant 10$) that facilities tend to be opened, closed and reopened more often. As the number of possible facility locations increases, the number of facilities that are opened more than once decreases.
2. Defining the set $I^+ = \{i \in I : \exists(i, \tau, \xi) \in I_A^+ \cup I_R^+\}$, it is possible to consider two subsets of I^+: subset I_1^+ containing facilities that are opened during most time periods, and subset I_2^+ containing facilities that are opened sporadically. With the increase in the number of possible facility locations, there is an increase in the number of facilities belonging to subset I_2^+.
3. The assignment of clients to facilities is basically stable during the planning horizon, because the assignment is generally made considering facility belonging to I_1^+. Exception is made for periods where facilities belonging to I_2^+ are in operation.

7. Conclusions and future work directions

In this paper the DLPOCR was formulated and an efficient primal-dual heuristic was described. This problem, studied here for the first time as an integer linear problem, increases the flexibility of the dynamic simple location problems. The model introduces the possibility of opening and closing a facility more than once during the planning horizon. It also considers explicitly not only installation (differentiating opening and reopening costs) and operating costs but also costs incurred by the closure of a facility. The primal-dual heuristic developed is capable of solving even large instances of the problem, generating good-quality solutions, and calculates tight lower bounds for the optimal objective function value.

The quality of the results obtained encouraged the authors to study the DLPOCR with additional restrictions, namely capacity restrictions [26]. These capacity restrictions can be dealt with in two ways: considering them explicitly in the dual problem formulation, or relaxing them in a Lagrangean way. In the latter case, the problem obtained is the DLPOCR, and a subgradient optimization method can be used. Instead of solving DLPOCR optimally, the lower bounds calculated by the primal-dual heuristic can be used.

References

[1] Beasley JE. Lagrangean heuristics for location problems. European Journal of Operational Research 1993;65:383–99.

[2] Cornuejols G, Nemhauser G, Wolsey L. The uncapacitated facility location problem. In: Mirchandani PB, Francis RL, editors. Discrete location theory. New York: Wiley Interscience; 1990. p. 119–72.

[3] Erlenkotter D. A dual-based procedure for uncapacitated facility location. Operations Research 1978;26:992–1009.

[4] Krarup J, Pruzan P. The simple plant location problem: survey and synthesis. European Journal of Operational Research 1983;12:36–81.

[5] Krarup J, Pruzan P. Ingredients of locational analysis in discrete location. In: Mirchandani PB, Francis RL, editors. Discrete location theory. New York: Wiley Interscience; 1990. p. 1–54.

[6] Morris JG. On the extent to which certain fixed-charged depot location problems can be solved by LP. Journal of the Operational Research Society 1978;29:71–6.

[7] Ross T, Soland R. Modelling facility location problems as generalized assignment problems. Management Science 1977;24:345–57.
[8] Swain R. A parametric decomposition approach for the solution of uncapacitated location problems. Management Science 1974;21:189–98.
[9] Erlenkotter D. A comparative study of approaches to dynamic location problems. European Journal of Operational Research 1981;6:133–43.
[10] Wesolowsky GO. Dynamic facility location. Management Science 1973;19:1241–8.
[11] Wesolowsky G, Truscott W. The Multiperiod location-allocation problem with relocation of factilities. Management Science 1975;22:57–65.
[12] Fong CO, Srinivasan V. The multiregion dynamic capacity expansion problem—Part I. Operations Research 1981;29: 787–99.
[13] Fong CO, Srinivasan V. The multiregion dynamic capacity expansion problem—Part II. Operations Research 1981;29: 800–16.
[14] Van Roy T, Erlenkotter D. A dual-based procedure for dynamic facility location. Management Science 1982;28: 1091–105.
[15] Laporte G, Dejax P. Dynamic location-routing problems. Journal of the Operational Research Society 1989;40:471–82.
[16] Jacobsen S. Multiperiod capacitated location models. In: Mirchandani PB, Francis RL, editors. Discrete location theory. New York: Wiley Interscience; 1990. p. 173–207.
[17] Shulman A. An algorithm for solving dynamic capacitated plant location problems with discrete expansion sizes. Operations Research 1991;39:423–36.
[18] Galvão RD, Santibañez-Gonzalez R. A Lagrangean heuristic for the P-median dynamic location problem. European Journal of Operational Research 1992;58:250–62.
[19] Melachrinoudis E, Min H, Wu X. A multiobjective model for the dynamic location of landfills. Location Science 1995;3: 143–66.
[20] Hinojosa Y, Puerto J, Fernández FR. A multiperiod two-echelon multicommodity capacitated plant location problem. European Journal of Operational Research 2000;123:271–91.
[21] Antunes A, Peeters D. On solving complex multi-period location models using simulated annealing. European Journal of Operational Research 2001;130:190–201.
[22] Chardaire P, Sutter A, Costa M-C. Solving the dynamic facility location problem. Networks 1996;28:117–24.
[23] Canel C, Khumawala B, Law J, Loh A. An algorithm for the capacitated, multi-commodity, multi-period facility location problem. Computers & Operations Research 2001;8:411–27.
[24] Saldanha da Gama F, Captivo ME. A note on a dual based procedure for dynamic facility location. Working Paper 11/96, Centro de Investigação Operacional, Faculdade de Ciências da Universidade de Lisboa, 1996.
[25] Ahuja R, Magnanti T, Orlin J. Network flows-theory, algorithms and applications. Englewood Cliffs, NJ: Prentice-Hall; 1993.
[26] Dias J, Captivo ME, Clímaco J. Capacitated dynamic location problems with opening, closure and reopening of facilities. Inescc Research Report 02/2004.