Dynamic Location Problems with Discrete Expansion and Reduction Sizes of Available Capacities

Joana Dias[†]
M. Eugénia Captivo[‡]
João Clímaco[†]

[†]Faculdade de Economia and INESC-Coimbra
Universidade de Coimbra, Av. Dias da Silva, 165, 3004 -512 Coimbra
joana@fe.uc.pt
jclimaco@inescc.pt

[‡]Universidade de Lisboa, Faculdade de Ciências
Centro de Investigação Operacional, Campo Grande, Bloco C6, Piso 4, 1749-016 Lisboa
mecaptivo@fc.ul.pt

Abstract

In this paper a dynamic location problem is formulated that considers the possibility of expanding or reducing the maximum available capacity at any given location during the planning horizon. The expansion (or reduction) of available capacity at a given location is achieved through the opening (or closure) of one or more facilities with different discrete capacities. The mixed-integer linear model developed considers fixed costs for opening the first facility at any location, plus additional fixed costs for every open facility in a location with already existing facilities. It is possible to open, close and reopen any facility at any location more than once during the planning horizon. It is also possible to consider different assignment costs depending on the size of the facility that is assigned to each client. This is important, because, in general, smaller facilities have smaller fixed costs but greater unitary operating costs. A primal-dual heuristic is developed that is able to find primal feasible solutions to the problem here described, and computational results are presented.

Resumo

Neste artigo considera-se um problema de localização dinâmica, em que é possível aumentar ou reduzir a capacidade máxima disponível numa dada localização, num determinado período de tempo, através da abertura ou fecho de um ou mais equipamentos, de iguais ou diferentes capacidades. O modelo de programação linear inteira mista desenvolvido considera os custos fixos de abrir o primeiro equipamento numa dada localização, e custos fixos associados à localização de outros equipamentos, para além do primeiro. É possível abrir, fechar e reabrir qualquer equipamento em qualquer localização, mais do que uma vez durante o horizonte temporal considerado. É também possível considerar custos de afectação que podem variar consoante a dimensão do equipamento. Foi desenvolvida uma heurística primal-dual capaz de resolver este problema, que aqui se descreve, e apresentam-se alguns resultados computacionais.

Keywords: dynamic location problems, primal-dual heuristics, capacity planning

1 Introduction

Capacitated location problems have been widely studied in the literature (see, for instance, Guignard and Spielberg, 1979, Jacobsen, 1983; Christofides and Beasley, 1983; Van Roy, 1986; Beasley, 1988; Cornuejols *et al.*, 1991; Sridharan, 1995). Dynamic location problems have also been studied (see, for instance, Van Roy and Erlenkotter, 1982; Saldanha da Gama and Captivo, 2002). It is interesting to note that a capacitated dynamic location problem is, in essence, a capacity expansion problem: facilities are open in different time periods, increasing the total available capacity, in order to serve a (generally) increasing demand. In this paper, the authors study a problem where the expansion of capacity is explicitly considered and is achieved not only through the location of facilities at new sites but also through the location of facilities that will increase the already existing capacity at a given site (as in Shulman, 1991). Each facility capacity has to be chosen from a finite (small) set of feasible capacities, similar to what is described in Lee (1991) and Mazzola and Neebe (1999). Lee extends the classical capacitated location problem and considers a multiproduct capacitated facility location problem in which each facility capacity has to be chosen from a given set of admissible capacities. The author solves the problem using an algorithm based on a Benders' decomposition. Sridharan (1991) studies the problem of locating and choosing the size of the facility, by solving a capacitated location problem with side constraints (guaranteeing that at most one facility is located at each site). The problem is solved using a lagrangean heuristic. Mazzola and Neebe (1999) study a similar problem and develop a branch and bound algorithm. Ghiani *et al.* (2002) study the problem of locating capacitated facilities, allowing several identical facilities to be located at the same site. The problem was motivated by a polling station location problem in an Italian Municipality. The authors solve the problem using a Lagrangean Heuristic.
As far as we know, in most of the references dealing explicitly with capacity expansions, they can be continuously incremented.
Hinomoto (1965) studies the problem of capacity expansion of a productive system, assuming the capacity can be expanded by the addition of facilities in discrete steps, and the size of a facility can be treated as a continuous factor. Erlenkotter (1975) develops two approaches to deal with capacity planning for large multilocation systems: an approximate approach based on an equivalent cost measure and an incomplete dynamic programming approach to systematically improve the approximate solution. The author describes an application to a real problem (India's nitrogenous fertilizer industry). Fong and Srinivasan (1981a) formulate the problem of continuous capacity expansion as a dynamic discrete time location mixed integer programming problem. The authors develop a heuristic to tackle the problem. In the sequel of this paper (Fong and Srinivasan, 1981b), the authors extend the problem considering a fixed cost if a capacity expansion takes place at a given location plus a cost proportional to the size of the expansion. Freindenfelds (1981a) considers the capacity expansion problem in which there are two types of demand and two types of facilities. The author considers that the capacity can be increased in a continuum of sizes, at a cost that does not depend either on time or on previous expansion decisions. In his book (Freindenfelds, 1981b), the author introduces a series of capacity expansion analytical models and applications, emphasizing the real capital investment decisions involved in the establishment of new productive capacity.

Smith (1981) generalizes the work of Manne[1], presenting an efficient algorithm that solves the deterministic capacity problem considering a finite planning horizon[2].

In 1982, Luss publishes a survey of the existing literature on capacity expansion problems. The author calls the readers attention to the lack of existing literature dealing with dynamic capacity expansion problems. In this survey the author considers both single and multi-facility location problems, with finite or infinite horizon time planning[2].

Min (1988) studies the problem of dynamic expansion and relocation of capacitated public facilities, considering multiple objectives. The author formulates the problem as a mixed integer goal programming model in a fuzzy decision environment. The problem is illustrated by considering the real case of expanding and relocating public libraries in the Columbus metropolitan public library system.

Shulman (1991) formulates the problem of dynamically locating and expanding the capacity of facilities. The author considers a small set of feasible expansion sizes (capacity expansion is achieved by dynamically locating more than one facility at a given location), and develops two algorithms: one deals with the more general problem that allows several facilities of different capacities to be located at the same site; the other solves the special case where it is only possible to locate facilities of the same size at each location.

The problem studied in this paper considers the situation where capacity expansion is achieved by locating additional facilities and the reduction of capacity is achieved by closing existing facilities. There is a finite (small) set of feasible capacities for the facilities to be located. The major differences between the problem here presented and the problems studied in the literature are the possibility of reducing the capacity at any time period (most of the problems studied only consider the possibility of capacity increasing), the possibility of locating several facilities of different sizes in the same location and also the possibility of a facility being open, closed and reopen more than once during the planning horizon. Canel *et al.* (2001) consider the possibility of a service 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. In the model here presented it is also possible to differentiate the operating costs of the different facilities.

According to Luss (1982) the major decisions in capacity expansion problems are: expansion sizes, expansion times and expansion locations. In this problem, one can say that the major decisions in capacity expansion and reduction are: expansion and reduction sizes, times and locations.

In this paper, the problem is formulated as a mixed-integer linear problem, and a primal-dual heuristic is described that can find primal feasible solutions. The behaviour of the primal-dual heuristic is compared with the behaviour of a general solver (Cplex). This work was motivated by the problem of locating transfer stations in a solid waste treatment system (see, for instance, Wirasinghe and Waters, 1983). Most transfer stations are composed by one or more equipments that can take one of a small set of different sizes. Each equipment has fixed and operating costs that are, usually, directly and inversely proportional (respectively) to its capacity.

In the next section the model developed is presented. In section 3, the dual problem of its linear relaxation is formulated. In section 4 the primal-dual heuristic (based on the work of Erlenkotter (1978) and Van Roy and Erlenkotter (1982)) is described. In section 5 some computational results are shown and, in the last section, some conclusions and future work directions are drawn.

2 The Proposed Mathematical Model

---

[1] A. S. Manne (1961), *Capacity Expansion and Probabilistic Growth*, Econometrics, 29

[2] In the paper's context, planning horizon refers to the time period during which the additional capacity can be used. It can be interpreted as the lifetime of a facility. In this research report, the planning horizon is interpreted as the time interval, explicitly considered, in which it is possible to change the configuration of facility locations.

Consider the following definitions:
$J = \{1,\ldots,n\}$ set of indices corresponding to the clients' locations;
$I = \{1,\ldots,m\}$ set of indices corresponding to facilities' possible locations;
$S = \{1,\ldots,q\}$ set of indices corresponding to facilities' possible dimensions, ordered by ascending order of the corresponding capacities;
$T =$ number of time periods considered in the planning horizon;
$c_{ijs}^{t} =$ cost of fully assigning client j to a facility of dimension s located at i in period t;
$FA_{ist}^{\xi} =$ fixed cost of opening a facility of dimension s at 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), knowing that this is the first facility located at i;
$FR_{ist}^{\xi} =$ unitary fixed cost of locating a facility of dimension s at 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), knowing that this facility is not the first to be located at i.[3]
$d_j^t =$ demand of client j at period t;
$Q_s =$ maximum capacity of a facility of dimension s;
$Nmax =$ maximum number of facilities that can be operational at one location at the same time.

Let us define the variables:

$$a_{ist}^{\xi} = \begin{cases} 1 & \text{if a facility of dimension } s \text{ located at } i \text{ is opened at the beginning of period } t \\ & \text{and stays open until the end of period } \xi, \text{ knowing that this is the first} \\ & \text{facility to be located at } i \\ 0 & \text{otherwise} \end{cases}$$

$r_{ist}^{\xi} =$ number of facilities of dimension s located at i at the beginning of period t and staying open until the end of period ξ, knowing that this is not the first facility to be located at i.
$x_{ijs}^{t} =$ fraction of customer j's demand that is served by a facility of dimension s located at i during period t.

The first facility to be located at i will be called *i-first* facility. All the other facilities that are located at i will be called *i-follow* facilities.
The dynamic location problem of expansion and reduction of available capacities, considering that it is possible to reconfigure one location more than once during the planning horizon can be formulated as:

DLPER

$$Min \sum_t \sum_i \sum_j \sum_s c_{ijs}^t x_{ijs}^t + \sum_t \sum_i \sum_s \sum_{\xi=t}^{T} FA_{ist}^{\xi} a_{ist}^{\xi} + \sum_t \sum_i \sum_s \sum_{\xi=t}^{T} FR_{ist}^{\xi} r_{ist}^{\xi} \tag{1}$$

subject to:

$$\sum_i \sum_s x_{ijs}^t = 1, \qquad \forall\, j,t \tag{2}$$

$$\sum_{\tau=1}^{t} \sum_{\xi=t}^{T} \left(a_{is\tau}^{\xi} + r_{is\tau}^{\xi} \right) - x_{ijs}^t \geq 0, \qquad \forall\, i,j,s,t \tag{3}$$

$$Nmax \sum_{s'} \sum_{\tau=1}^{t} \sum_{\psi=\tau}^{T} a_{is'\tau}^{\psi} - r_{ist}^{\xi} \geq 0, \qquad \forall\, i,s,t,\xi \geq t \tag{4}$$

[3] The fixed cost FA_{ist}^{ξ} should be equal to FR_{ist}^{ξ} plus the additional cost of installing for the first time a facility at location i. This additional cost may represent costs of land acquisition, development of infra-structures, etc. Let us define this additional cost as f_i^t. Then $FA_{ist}^{\xi} = FR_{ist}^{\xi} + f_i^t, \forall i,s,t,\xi \geq t$.

$$\sum_{s}\sum_{t=1}^{T}\sum_{\xi=t}^{T} a_{ist}^{\xi} \le 1, \qquad \forall\ i \qquad (5)$$

$$\sum_{s}\sum_{\tau=1}^{t}\sum_{\xi=t}^{T}\left(a_{is\tau}^{\xi} + r_{is\tau}^{\xi}\right) \le Nmax, \qquad \forall\ i,t \qquad (6)$$

$$Q_s\sum_{\tau=1}^{t}\sum_{\xi=t}^{T}\left(a_{is\tau}^{\xi} + r_{is\tau}^{\xi}\right) - \sum_{j} d_j^t x_{ijs}^t \ge 0, \qquad \forall i,s,t \qquad (7)$$

$$\begin{array}{ll} a_{ist}^{\xi} \in \{0,1\}, & \\ r_{ist}^{\xi} \ge 0 \text{ and integer}, & \forall i,s,t,\xi \ge t \\ x_{ist}^{t} \ge 0, & \forall i,s,t,\xi \ge t \end{array} \qquad (8)$$

These constraints guarantee that:
(2): Each client's demand will be fully satisfied in each time period;
(3): A client will be assigned to open facilities only;
(4): A facility of dimension *s* that is opened at *i* at the beginning of period *t* can be considered as an *i-follow* facility only if there is an *i-first* facility that has been open at the beginning of a time period $t' \le t$ (an *i-follow* facility and the *i-first* facility can be located simultaneously);
(5): For each location *i*, there can be at most one *i-first* facility during the whole planning horizon;
(6): There is an upper limit on the number of operating facilities at location *i*, in each time period;
(7): The facilities' maximum capacity will not be exceeded in any time period.
The proposed model allows that, in each time period and in each location, every mix of facilities of different or equal dimensions is feasible, up to a maximum of *Nmax* number of facilities in each location.

3 The Dual Problem and Complementary Conditions

3.1 Formulation of the Dual Problem

Multiplying constraints (5) and (6) by -1 and associating dual variables v_j^t with constraints (2), dual variables w_{ijs}^t with constraints (3), dual variables u_{ist}^{ξ} with constraints (4), dual variables ρ_i with constraints (5), dual variables π_i^t with constraints (6), and dual variables λ_{is}^t with constraints (7), the dual problem of DLPER can be formulated as D-DLPER:

D-DLPER

$$Max\sum_{t}\sum_{j} v_j^t - \sum_{i}\rho_i - Nmax\sum_{t}\sum_{i}\pi_i^t \qquad (9)$$

subject to:

$$v_j^t - w_{ijs}^t - d_j^t\lambda_{is}^t \le c_{ijs}^t, \qquad \forall i,j,s,t \qquad (10)$$

$$\sum_{j}\sum_{\tau=t}^{\xi} w_{ijs}^{\tau} + Nmax\sum_{s'}\sum_{\tau=t}^{T}\sum_{\psi=\tau}^{T} u_{is'\tau}^{\psi} - \rho_i - \sum_{\tau=t}^{\xi}\pi_i^{\tau} + Q_s\sum_{\tau=t}^{\xi}\lambda_{is}^{\tau} \le FA_{ist}^{\xi}, \qquad \forall i,s,t,\xi = t,\cdots,T \qquad (11)$$

$$\sum_{j}\sum_{\tau=t}^{\xi} w_{ijs}^{\tau} - u_{ist}^{\xi} - \sum_{\tau=t}^{\xi}\pi_i^{\tau} + Q_s\sum_{\tau=t}^{\xi}\lambda_{is}^{\tau} \le FR_{ist}^{\xi}, \qquad \forall i,s,t,\xi = t,\cdots,T \qquad (12)$$

$$w_{ijs}^t, u_{ist}^{\xi}, \rho_i, \pi_i^t, \lambda_{is}^t \ge 0, \qquad \forall i,j,s,t,\xi = t,\cdots,T$$

An equivalent condensed formulation is obtained by considering $w_{ijs}^t = \max\left\{0, v_j^t - c_{ijs}^t - d_j^t \lambda_{is}^t\right\}, \forall i, j, s, t$.

CD-DLPER

$$Max \sum_t \sum_j v_j^t - \sum_i \rho_i - N\max \sum_t \sum_i \pi_i^t$$

subject to:

$$\sum_j \sum_{\tau=t}^{\xi} \max\left\{0, v_j^\tau - c_{ijs}^\tau - d_j^\tau \lambda_{is}^\tau\right\} \le FA_{ist}^\xi - Nmax \sum_{s'} \sum_{\tau=t}^{T} \sum_{\psi=\tau}^{T} u_{is'\tau}^\psi + \rho_i + \sum_{\tau=t}^{\xi} \pi_i^\tau - Q_s \sum_{\tau=t}^{\xi} \lambda_{is}^\tau, \quad \forall i, s, t, \xi = t, \cdots, T \qquad (13)$$

$$\sum_j \sum_{\tau=t}^{\xi} \max\left\{0, v_j^\tau - c_{ijs}^\tau - d_j^\tau \lambda_{is}^\tau\right\} \le FR_{ist}^\xi + u_{ist}^\xi + \sum_{\tau=t}^{\xi} \pi_i^\tau - Q_s \sum_{\tau=t}^{\xi} \lambda_{is}^\tau, \quad \forall i, s, t, \xi = t, \cdots, T \qquad (14)$$

$$u_{ist}^\xi, \rho_i, \pi_i^t, \lambda_{is}^t \ge 0, \quad \forall i, j, s, t, \xi = t, \cdots, T$$

3.2 Complementary Conditions

Let us define:

$$SA_{ist}^\xi = FA_{ist}^\xi - Nmax \sum_{s'} \sum_{\tau=t}^{T} \sum_{\psi=\tau}^{T} u_{is'\tau}^\psi + \rho_i + \sum_{\tau=t}^{\xi} \pi_i^\tau - \sum_j \sum_{\tau=t}^{\xi} \max\left\{0, v_j^\tau - c_{ijs}^\tau - d_j^\tau \lambda_{is}^\tau\right\} - Q_s \sum_{\tau=t}^{\xi} \lambda_{is}^\tau, \quad \forall i, s, t, \xi = t, \cdots, T \qquad (15)$$

$$SR_{ist}^\xi = FR_{ist}^\xi + u_{ist}^\xi + \sum_{\tau=t}^{\xi} \pi_i^\tau - \sum_j \sum_{\tau=t}^{\xi} \max\left\{0, v_j^\tau - c_{ijs}^\tau - d_j^\tau \lambda_{is}^\tau\right\} - Q_s \sum_{\tau=t}^{\xi} \lambda_{is}^\tau, \quad \forall i, s, t, \xi = t, \cdots, T \qquad (16)$$

$$S_{ist}^\xi = \min\left\{SA_{ist}^\xi, SR_{ist}^\xi\right\}, \quad \forall i, s, t, \xi = t, \cdots, T \qquad (17)$$

The following complementary conditions hold if in presence of optimal primal and dual solutions to the respective problems (when there is no duality gap).

$$\left(\sum_{\tau=1}^{t} \sum_{\xi=t}^{T} \left(a_{is\tau}^\xi + r_{is\tau}^\xi\right) - x_{ijs}^t\right) w_{ijs}^t = 0, \quad \forall i, j, t \qquad (18)$$

$$\left(Nmax \sum_{s'} \sum_{\tau=1}^{t} \sum_{\psi=\tau}^{T} a_{is'\tau}^\psi - r_{ist}^\xi\right) u_{ist}^\xi = 0, \quad \forall\ i, s, t, \xi \ge t \qquad (19)$$

$$\left(\sum_s \sum_{t=1}^{T} \sum_{\xi=t}^{T} a_{ist}^\xi - 1\right) \rho_i = 0, \quad \forall i \qquad (20)$$

$$\left(\sum_s \sum_{\tau=1}^{t} \sum_{\xi=\tau}^{T} \left(a_{is\tau}^\xi + r_{is\tau}^\xi\right) - Nmax\right) \pi_i^t = 0, \quad \forall i, t \qquad (21)$$

$$SA_{ist}^\xi \cdot a_{ist}^\xi = 0, \quad \forall i, s, t, \xi = t, \cdots, T \qquad (22)$$

$$SR_{ist}^\xi \cdot r_{ist}^\xi = 0, \quad \forall i, s, t, \xi = t, \cdots, T \qquad (23)$$

$$\left(Q_s \sum_{\tau=1}^{t} \sum_{\xi=t}^{T} \left(a_{is\tau}^\xi + r_{is\tau}^\xi\right) - \sum_j d_j^t x_{ijs}^t\right) \lambda_{is}^t = 0, \quad \forall i, s, t \qquad (24)$$

4 Primal-Dual Heuristic

The primal-dual heuristic here proposed follows the work of Erlenkotter (1978), Guignard and Spielberg (1979) and Van Roy and Erlenkotter (1982).
The heuristic builds primal admissible solutions based on dual admissible solutions, trying to force the satisfaction of the complementary conditions. If a pair of primal and dual admissible solutions is found that satisfies conditions (18) – (24), then the optimal solution has been calculated. When this is not achieved, the best dual solution gives a valid lower bound for the optimal solution and may be used to assess the quality of the best admissible primal solution found.
The heuristic operating 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 Primal-dual 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_{ist}^{ξ}. If the dual solution is changed go to 2;
8. Dual Descent Procedure for dual variables u_{ist}^{ξ}. If the dual solution is changed go to 2;
9. If *Nmax* =1, then execute the Dual Adjustment Procedure for variables π_i^t. If the dual solution is changed go to 2.
10. Dual Ascent Procedure for dual variables λ_{is}^t. If the dual solution is changed go to 2;
11. Dual Descent Procedure for dual variables λ_{is}^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 either the primal or the dual objective function values.
The Dual Ascent Procedure for dual variables v_j^t, the Primal-dual Adjustment Procedure for variables v_j^t, the Dual Adjustment Procedures for variables π_i^t and ρ_i are exactly the same as those developed by the authors for the resolution of the Dynamic Location Problem with Opening, Closure and Reopening of Facilities - DLPOCR (Dias *et al.*, 2007a). Instead of considering the set *I* of possible locations, it should be considered the set $I \times S$. In the Dual Ascent Procedure for dual variables v_j^t, and in the Primal-dual Adjustment Procedure the assignment costs should be considered equal to $c_{ijs}^t + d_j^t \lambda_{is}^t$. For this reason, these procedures will not be described here. The Dual Adjustment Procedure for variables π_i^t is executed only when *Nmax* equals 1. The computational experiments put in evidence that, in every other situation, the change of this dual variable does not increase the value of the dual objective function due to the variable's dual objective function coefficient –*Nmax*.
The Dual Ascent and Descent Procedures for variables λ_{is}^t are also similar to the ones already developed by the authors for the DLPOCR with maximum capacity restrictions

(Dias *et al.*, 2006). It is sufficient to consider facilities (i,s) with maximum capacities equal to Q_s, instead of facilities i with maximum capacities equal to Q_i. The two procedures referred will not be repeated here.

4.1 Initialization of dual variables

The dual variables are initialised as follows:

1.	$v_j^t \leftarrow \min_{i,s}\left\{ c_{ijs}^t \right\}, \forall j,t; \quad \pi_i^t \leftarrow 0,$	$\forall i,t$
2.	$u_{ist}^{\xi} \leftarrow \max\left\{0, -FR_{ist}^{\xi}\right\},$	$\forall i,s,t,\xi = t,\cdots,T$
3.	$\rho_i \leftarrow \max\left\{0, -\min_{\substack{t,s \\ \xi \ge t}}\left(FA_{ist}^{\xi} - N \max \sum_s \sum_{\tau=t}^{T} \sum_{\psi=\tau}^{T} u_{is\tau}^{\psi} \right)\right\},$	$\forall i$

4.2 Primal Procedure

Consider the following definitions:

$I^* = \{ (i,s,\tau,\xi): S_{is\tau}^{\xi} = 0\}$,

$I_t^* = \{(i,s) : (i,s,\tau,\xi) \in I^* \text{ and } \tau \le t \le \xi\}$,

$I_t^+ = \{(i,s)$: at least one facility of dimension s is open at i during period $t\}$,

$I^+ = \{ (i,s,\tau,\xi) : a_{is\tau}^{\xi} = 1 \text{ or } r_{is\tau}^{\xi} > 0\}$,

F_{is}^t = smallest cost incurred by opening a facility of dimension s at i during period t,

$Num[i,s,t]$ = total number of open facilities at location i of dimension s during period t[4],

$N[i,t]$ = total number of open facilities at location i during period t[5].

Primal Procedure

1. $I^+ \leftarrow \varnothing$. $I_t^+ \leftarrow \varnothing, \forall t$. Build sets I^* and I_t^*. $Num[i,s,t] \leftarrow 0, \forall i,s,t$. $Num[i,t] \leftarrow 0, \forall i,t$.
2. For t=1 until T, include in set I_t^+ all pairs $(i,s) \in I_t^*$ such that $\exists j : v_j^t \ge c_{ijs}^t$ and $v_j^t < c_{i's'j}^t, \forall (i',s') \ne (i,s)$ (essential facilities as in Van Roy and Erlenkotter (1982) and Dias *et al.* (2007a)).
3. For each client j such that $v_j^t < c_{ijs}^t, \forall (i,s) \in I_t^+$, include in set I_t^+ the pair (i,s) such that $c_{ijs}^t = \min_{v_j^t \ge c_{i'js'}^t} c_{i'js'}^t$.
4. Build set I^+. Update $I_t^+, \forall$t.
5. $t \leftarrow 1$;
6. $N[i,t] \leftarrow \sum_s Num[i,s,t], \forall i \in I$.
7. $D \leftarrow \sum_j d_j^t$; $C \leftarrow \sum_{(i,s)\in I_t^+} \left(Q_s \cdot Num[i,s,t]\right)$. If $D \le C$ then go to 17.
8. Calculate $F_{is}^t, \forall\ i \in I,\ s \in S$.
9. If $F_{is}^t = +\infty,\ \forall\ i \in I,\ s \in S$, then go to 13.
10. Calculate $F'^t_{is} = \frac{F_{is}^t}{Q_s}\left\lceil \frac{\phi_s}{Q_s} \right\rceil, \forall\ i \in I,\ s \in S$, where $\phi_s = \begin{cases} D - C, & \text{if } C + Q_s < D \\ Q_s, & \text{otherwise} \end{cases}$.
11. Consider the pair (i',s') such that $F'^t_{i's'} = \min_{i \in I, s \in S}\left\{F'^t_{is}\right\}$;

[4] Represents the total number of elements $(i,s,\tau,\xi) \in I^+$ such that $\tau \le t \le \xi$.

[5] Represents the total number of elements $(i,s,\tau,\xi) \in I^+$ such that $\tau \le t \le \xi$, $\forall\ s \in S$.

12. $I_t^+ \leftarrow I_t^+ \cup \{(i', s')\}$; Rebuild sets I^+ and $I_t^+, \forall t$; $C \leftarrow C + Q_s$; $Num[i,s,t] \leftarrow Num[i,s,t]+1$; $N[i,t] \leftarrow N[i,t]+1$. If $D \leq C$ then go to 17. Else go to 8.
13. If $s = q$ for every $(i,s,\tau,\xi) \in I^+$ with $\tau \leq t \leq \xi$, then Stop. The procedure cannot find a feasible solution.
14. If $D > C$ then, for every $(i,s,\tau,\xi) \in I^+$ with $s<q$ and $\tau \leq t \leq \xi$, calculate $H_{is\tau}^{\xi} = \left(FR_{i(s+1)\tau}^{\xi} - FR_{is\tau}^{\xi}\right) \cdot \left\lceil \dfrac{D-C}{Q_{s+1}-Q_s} \right\rceil$.
15. Choose $(i,s,\tau,\xi) \in I^+$ with $s<q$ and $\tau \leq t \leq \xi$ that corresponds to the smallest $H_{is\tau}^{\xi}$.
    $I^+ \leftarrow I^+ \setminus (i,s,\tau,\xi)$; $I^+ \leftarrow I^+ \cup \{(i,s+1,\tau,\xi)\}$; $Num[i,s,t] \leftarrow Num[i,s,t] - 1$; $Num[i,s+1,t] \leftarrow Num[i,s+1,t] + 1$; $C \leftarrow C + Q_{s+1} - Q_s$.
16. If $D >$ go to 13.
17. $t \leftarrow t + 1$; If $t \leq T$ go to 6.
18. $t \leftarrow 1$;
19. Solve one transportation problem considering as sources the set J of clients (with supplies d_j^t), as destinations all pairs $(i,s) \in I_t^+$ (with demands $Q_s.Num[i,s,t]$), and transportation costs (per unit) given by $\dfrac{c_{ijs}^t}{d_j^t}$.
20. $t \leftarrow t + 1$; If $t \leq T$ go to 19.
21. Calculate the values of primal variables $a_{is\tau}^{\xi}$ and $r_{is\tau}^{\xi}$.
22. Execute a local exchange search procedure.

There are several steps in this primal procedure that deserve further explanations.
Step 4 of the primal procedure (building set I^+) can be described as follows:

Step 4 of the Primal Procedure

1. $i \leftarrow 1$.
2. $s \leftarrow 1$.
3. If $\exists\, t$: $(i,s) \in I_t^+$, go to 4; else go to 9.
4. $t_1 \leftarrow min\{\tau : (i,s) \in I_\tau^+\}$; $t_2 \leftarrow max\{\tau : (i,s) \in I_\tau^+\}$.
5. Calculate $Num[i,s,t]$ and $N[i,t]$, $\forall t$. $I1^+ \leftarrow I^+$. Execute Procedure 1.
6. Calculate $Num[i,s,t]$ and $N[i,t]$, $\forall t$. $I2^+ \leftarrow I^+$. Execute Procedure 2.
7. $sum1 \leftarrow \sum_{(i,s,\tau,\xi) \in I1^+} FR_{is\tau}^{\xi}$; $sum2 \leftarrow \sum_{(i,s,\tau,\xi) \in I2^+} FR_{is\tau}^{\xi}$.
8. If $(sum1 < sum2)$ $I^+ \leftarrow I1^+$; else $I^+ \leftarrow I2^+$. Calculate $Num[i,s,t]$, $\forall t$.
9. $s \leftarrow s + 1$; If $s \leq q$ then go to 3.
10. $i \leftarrow i + 1$; if $i > m$ stop. Else go to 2.

After the execution of step 4, set I^+ has as many (i,s,τ,ξ) elements as the number of facilities of dimension s that are operating at i from the beginning of time period τ to the end of time period ξ.
Procedures 1 and 2 are based on similar procedures described in Dias *et al.*, 2007a. The main differences are due to the fact that in DLPOCR an admissible solution has, at most, one facility open in each location during each time period. In DLPER, it is admissible to have more than one facility simultaneously open. As described in Dias *et al.*, 2007a, procedure 1 builds a solution from period t_1 forward, while procedure 2 builds a solution from period t_2 backwards.
Step 7 of the primal procedure tests the admissibility of the primal solution constructed in terms of total available capacity.
In step 8 of the primal procedure, the calculation of F_{is}^t accounts for all the hypotheses of having a facility of dimension s open at i during time period t. There are two possibilities:

a new facility is placed or the operating upper and/or lower time limits of an already existing facility are changed. The calculation of F_{is}^{t} tries to find the best choice in terms of fixed costs incurred.
Let us define:

$$F_i^{\tau} = \begin{cases} FA_{is\tau}^{\xi} - FR_{is\tau}^{\xi}, \forall s, \xi, & \text{if } \nexists (i,s') \in I_t^+ : t < \tau, \forall s' \in S \\ 0, & \text{otherwise} \end{cases}.$$

This value represents the fixed cost incurred if the first equipment is placed at i at the beginning of time period τ.
Consider that facility i is not open during period t but is open in time periods before and after t, as depicted in figure 1.

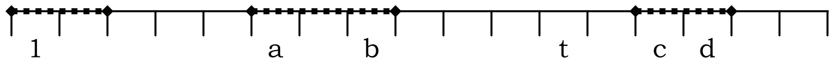

Figure 1: represents facility i operating time periods

Time periods a, b, c, d can be defined formally as: $b = \max\left\{0, \max_{t'<t}\left\{t' : i \in I_{t'}^{+}\right\}\right\}$; $a = t' : (i, t', b) \in I_A^+ \cup I_R^+$; $c = \min\left\{T+1, \min_{t'>t}\left\{t' : i \in I_{t'}^{+}\right\}\right\}$; $d = t' : (i, c, t') \in I_A^+ \cup I_R^+$. Time period b represents the time period before and nearest to t such that facility i is operating. Time period c represents the time period after and nearest to t such that facility i is operating. Let us also define $F_i^{'}$ as the fixed cost of installing for the first time a facility at i, considering the set I^+: $F_i^{'} = \begin{cases} F_i^{\tau} & \text{if } \exists (i,s') \in I_{\tau}^{+}, s' \in S \wedge \nexists (i,s') \in I_t^+ : t < \tau, \forall s' \in S \\ 0, & \text{otherwise} \end{cases}$

Calculation of F_{is}^{t}

1. $F_{is}^{t} \leftarrow +\infty$. If $N[i,t]$ = $Nmax$ then stop.
2. $F_{is}^{t} \leftarrow \min\left\{ F_{is}^{t}, \begin{cases} \min\left\{ FA_{is\tau}^{\xi} - F_i^{'} : \tau \le t \le \xi \text{ and } N[i,k] < \text{Nmax}, k \in [\tau, \xi] \right\}, & \text{if } \nexists (i,s') \in I_{t'}^{+}, t' \le t, s' \in S \\ \min\left\{ FR_{is\tau}^{\xi} : \tau \le t \le \xi \text{ and } N[i,k] < \text{Nmax}, k \in [\tau, \xi] \right\}, & \text{otherwise} \end{cases} \right\}$ [6]
3. Calculate $b = \max\left\{0, \max_{t'<t}\left\{t' : (i,s) \in I_{t'}^{+}\right\}\right\}$; $a = \left\{t' : (i,s,t',b) \in I^+\right\}$;
   $c = \min\left\{T+1, \min_{t'>t}\left\{t' : (i,s) \in I_{t'}^{+}\right\}\right\}$; $d = \left\{t' : (i,s,c,t') \in I^+\right\}$.
4. If $b = 0$ and $c \le T$ then
   $F_{is}^{t} \leftarrow \min\left\{F_{is}^{t}, \min_{\tau}\left\{FR_{is\tau}^{d} + F_i^{\tau} - FR_{isc}^{d} - F_i^{c} : \tau \le t, N[i,k] < \text{Nmax}, k \in [\tau, c[\right\}\right\}$ [7], stop.
5. If $c = T + 1$ and $b > 0$ then
   $F_{is}^{t} \leftarrow \min\left\{F_{is}^{t}, \min_{\xi}\left\{FR_{isa}^{\xi} - FR_{isa}^{b} : \xi \ge t, N[i,k] < Nmax, k \in]b, \xi]\right\}\right\}$, stop.

---

[6] If there is any facility operating in i during a time period before t, then the procedure considers only reopening fixed costs. Otherwise, it considers the cost of opening a facility for the first time, but discounts the fixed cost of installing for the first time a facility at time period $t' > \tau$ (if that is the case).

[7] At this step the procedure tries to calculate the changes in fixed costs by merging two time intervals: (τ,ξ) and (c,d). If period c corresponds to the period of the first installation of a facility in i then $\nexists (i,s') \in I_t^+, t < \tau, \forall s'$, so F_i^{τ} corresponds to the fixed cost of installing the first facility in i. If the facility was open for the first time before c then $F_i^{c} = 0$. The same holds for $F_i^{\tau} = 0$.

6. If $c \leq T$ and $b > 0$ and $N[i,k] < Nmax$, $k \in]b,c[$ then $F_{is}^{t} \leftarrow \min\left\{F_{is}^{t}, FR_{isa}^{d} - FR_{isa}^{b} - FR_{isc}^{d}\right\}$.

Step 10 penalizes facilities with a dimension that is not sufficient to cover the difference between the clients' total demand and the total available capacity (in this case it would be necessary to open more than one facility). The set I^{+} is updated, at step 12, considering the upper and lower limits that correspond to the smallest $F_{is}^{\prime t}$ value. It is important to note that, in set I^{+}, it is possible to have several identical elements (i,s,τ,ξ). This corresponds to the situation where several identical facilities are installed at the same location and are open and closed in the same time periods.

If it is not possible to find a feasible solution by opening more facilities during period t, steps (13)-(16) try to build a feasible solution by changing the dimension of the already located facilities. If all the facilities installed are of dimension q, then the procedure will not be able to build a feasible solution. Otherwise, for every $(i,s,\tau,\xi) \in I^{+}$ with $s<q$ and $\tau \leq t \leq \xi$, the procedure calculates the cost of changing dimension s to $s+1$ (penalizing those changes that are not sufficient to guarantee primal admissibility).

Procedure 1:

```
begin ← 1;time ← t1;
WHILE time ≤ t2
    IF N[i,time]=Nmax THEN time ← time +1; CONTINUE; ENDIF
    IF (i,s) ∈ 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, s,τ, ξ ) ∈ I+ THEN
                    (i, s, τ, ξ )→ I1+
                    FOR k = τ TO k = ξ DO
                        Num[i,s,k] ← Num[i,s,k] + 1
                        N[i,k] ← N[i,k] + 1
                        IF N[i,k]=Nmax and begin ≤ k THEN
                            begin ← k +1
                        ENDIF
                    ENDFOR
                    time ← ξ +1
                    stop ← true
                ENDIF
                ELSE ξ ← ξ -1 ENDELSE
            ENDWHILE
            τ ← τ +1; ξ ← T
        ENDWHILE
        IF not stop THEN
            ξ ← t2
            FOR k = time TO k = t2 DO
                IF N[i,k] = Nmax THEN
                    ξ ← k-1
                    BREAK
                ENDIF
            ENDFOR
            IF (time ≤ ξ ) THEN
                (i, s, time, ξ )→ I1+
```

```
                        FOR k = time TO k = ξ DO
                            N[i,k] ← N[i,k]+1; Num[i,s,k] ← Num[i,s,k] +1
                            IF N[i,k]=Nmax and begin ≤ k THEN
                                begin ← k +1
                            ENDIF
                        ENDFOR
                        time ← ξ +1
                    ENDIF
                    ELSE time ← time +1; begin ← ξ +1; ENDELSE
                ENDIF
            ENDIF
            ELSE time ← time +1 ENDELSE
    ENDWHILE
```

Procedure 2

```
end ← T; time ← t2 ;
WHILE time ≥ t1
      IF N[i,time]=Nmax THEN time ← time – 1; CONTINUE; ENDIF
      IF (i,s) ∈ Γ+time THEN
            τ ← 1; ξ ← end; t ← time; stop ← false;
            WHILE ξ ≥ t and not stop
                  WHILE τ ≤ t and not stop
                        IF ∃ (i, s, τ ,ξ ) ∈ Γ*  THEN
                              (i, s, τ,ξ )→ I2+
                              FOR k = τ  TO k = ξ  DO
                                    N[i,k] ← N[i,k]+1; Num[i,s,k] ← Num[i,s,k] +1
                                    IF N[i,k]=Nmax and end ≥ k THEN
                                          end ← k –1
                                    ENDIF
                              ENDFOR
                              time ← τ -1
                              stop ← true
                        ENDIF
                        ELSE τ  ← τ +1 ENDELSE
                  ENDWHILE
                  ξ ← ξ -1; τ ←1
            ENDWHILE
            IF not stop THEN
                  τ ← t1
                  FOR k = time DOWNTO k = τ  DO
                        IF N[i,k] = Nmax THEN
                              τ  ← k +1
                              BREAK
                        ENDIF
                  ENDFOR
                  IF (time ≥ τ  ) THEN
                        (i, s, τ, time)→ I2+
                        FOR k = τ  TO k = time  DO
                              N[i,k] ← N[i,k]+1; Num[i,s,k] ← Num[i,s,k] +1
                              IF N[i,k]=Nmax and end ≥ k THEN
                                    end ← k –1
                              ENDIF
                        ENDFOR
                        time ← τ –1
                  ENDIF
                  ELSE time ← time –1; end ← τ –1; ENDELSE
            ENDIF
      ENDIF
      ELSE time ← time –1; ENDELSE
ENDWHILE
```

Steps (18)-(20) solve the allocation problem, through the resolution of T transportation problems.

Step 21 calculates the values of the primal variables $a_{is\tau}^{\xi}$ and $r_{is\tau}^{\xi}$. This is done in a straightforward manner. As costs $FA_{is\tau}^{\xi}$ are equal to $FR_{is\tau}^{\xi}$ plus the additional cost of opening a facility in i for the first time (see footnote 1), the following procedure is used for each location i:

1. If $N[i,t] = 0$, $\forall t$, then stop.
2. Choose arbitrarily one element $(i,s,\tau,\xi) \in I^{+}$, such that $\tau = min\{t: \exists\ (i,s) \in I_t^{+}\}$.
3. Set $a_{is\tau}^{\xi} = 1$. Eliminate (i,s,τ,ξ) from set I^{+}.
4. All variables $r_{is\tau}^{\xi}$ are set to the number of elements $(i,s,\tau,\xi) \in I^{+}$.

In step 22, the procedure tries to improve the feasible primal solution found by executing a very simple local exchange heuristic. The local exchange heuristic tries to change variable $a_{is\tau}^{\xi}$ (and $r_{is\tau}^{\xi}$), to $a_{is'\tau}^{\xi}$ (and $r_{is'\tau}^{\xi}$), $\forall$ $s' \neq s$ and $s' \in S$. It chooses the admissible change that corresponds to the greatest improvement in the primal objective function. The process is repeated until there are no improvements in the primal objective function value.

4.3 Dual Ascent Procedure for dual variables u_{ist}^{ξ}

As can be seen by expressions (15) and (16), the increase in the dual variable u_{ist}^{ξ} can increase slacks SR_{ist}^{ξ}, but decreases slacks $SA_{is'\tau}^{\xi}$, $\tau \leq t$, $\forall s' \in S$. It is only worth trying to increase slacks SR_{ist}^{ξ} such that $SR_{ist}^{\xi} = 0$ and $SA_{ist}^{\xi} > 0$, otherwise the value S_{ist}^{ξ} would not be changed and would not be possible to increase the value of dual variables v_j^t.

Dual Ascent Procedure for variables u_{ist}^{ξ}

1. $i \leftarrow 1$.
2. $s \leftarrow 1$.
3. $t \leftarrow 1$.
4. $\xi \leftarrow t$.
5. $\Delta u_{ist}^{\xi} \leftarrow 0$.
6. If $SR_{ist}^{\xi} = 0$ and $SA_{ist}^{\xi} > 0$, then $\Delta u_{ist}^{\xi} \leftarrow \dfrac{SA_{ist}^{\xi}}{2}$. Else go to 10.
7. $\Delta u_{ist}^{\xi} \leftarrow \min\left\{\Delta u_{ist}^{\xi}, \min\limits_{\substack{s' \in S \\ \tau \leq t \\ \psi \geq \tau}} SA_{is'\tau}^{\psi}\right\}$. $\Delta u_{ist}^{\xi} \leftarrow \dfrac{\Delta u_{ist}^{\xi}}{\text{Nmax}}$. If $\Delta u_{ist}^{\xi} = 0$ then go to 10.
8. $SR_{ist}^{\xi} \leftarrow SR_{ist}^{\xi} + \Delta u_{ist}^{\xi}$; $SA_{is'\tau}^{\xi} \leftarrow SA_{is'\tau}^{\xi} - \Delta u_{ist}^{\xi} \cdot Nmax$, $\tau \leq t$, $\forall s' \in S$; $u_{ist}^{\xi} \leftarrow u_{ist}^{\xi} + \Delta u_{ist}^{\xi}$.
9. Execute the Dual Ascent Procedure for Variables v_j^t.
10. $\xi \leftarrow \xi + 1$. If $\xi > T$ then $t \leftarrow t + 1$ and go to 11. Else go to 5.
11. If $t > T$ then $s \leftarrow s + 1$ and go to 12. Else go to 4.
12. If $s > q$ then $i \leftarrow i + 1$ and go to 13. Else go to 3.
13. If $i > m$ then stop. Else go to 2.

In step 6 of this procedure, Δu_{ist}^{ξ} takes the value $\dfrac{SA_{ist}^{\xi}}{2}$ because if it would take the value SA_{ist}^{ξ} this slack could become equal to zero in step 8 of this procedure, decreasing the possibilities of improving the dual objective function value. If $SA_{ist}^{\xi} = \min\limits_{\substack{s' \in S \\ \tau \leq t \\ \psi \geq \tau}}\left\{SA_{is'\tau}^{\psi}\right\}$, then SA_{ist}^{ξ} and SR_{ist}^{ξ} will end up with the same value in step 8 of the procedure.

4.4 Dual Descent Procedure for dual variables u_{ist}^{ξ}

Decreasing the value of the dual variable u_{ist}^{ξ} will decrease the value of slack SR_{ist}^{ξ}, but will increase the value of all slacks $SA_{is'\tau}^{\xi}$, $\tau \le t$, $\forall s' \in S$. If the procedure increases the value of a slack that was blocking dual variables v_j^t, it is possible to increase the dual objective function value.

Dual Descent Procedure for variables u_{ist}^{ξ}

1. $i \leftarrow 1$.
2. $s \leftarrow 1$.
3. $t \leftarrow 1$.
4. $\xi \leftarrow t$.
5. $\Delta u_{ist}^{\xi} \leftarrow min\{ u_{ist}^{\xi}, \frac{SR_{ist}^{\xi}}{2} \}$. If $\Delta u_{ist}^{\xi} = 0$, go to 8.
6. $SR_{ist}^{\xi} \leftarrow SR_{ist}^{\xi} - \Delta u_{ist}^{\xi}$; $SA_{is'\tau}^{\psi} \leftarrow SA_{is'\tau}^{\psi} + \Delta u_{ist}^{\xi} \cdot Nmax$, $\tau \le t$ and $\psi \ge \tau$, $\forall s' \in S$; $u_{ist}^{\xi} \leftarrow u_{ist}^{\xi} - \Delta u_{ist}^{\xi}$.
7. Execute the Dual Ascent Procedure for Variables v_j^t.
8. $\xi \leftarrow \xi + 1$. If $\xi > T$ then $t \leftarrow t + 1$ and go to 11. Else go to 5.
9. If $t > T$ then $s \leftarrow s + 1$ and go to 12. Else go to 4.
10. If $s > q$ then $i \leftarrow i + 1$ and go to 13. Else go to 3.
11. If $i > m$ then stop. Else go to 2.

In step 5 of the procedure Δu_{ist}^{ξ} takes the minimum value between u_{ist}^{ξ} and $\frac{SR_{ist}^{\xi}}{2}$ for the same reasons already pointed out for the dual ascent procedure.

5 Computational Results

The primal-dual heuristic was tested with a set of randomly generated problems. Clients and possible locations for facilities are randomly generated, as well as capacities and demands. The data for the test problems were generated according to the following procedure:

1. Random generation of (x,y) coordinates in the plane of the $m+n$ nodes of the network according to a uniform distribution and considering a 500×500 square.
2. Random creation of arcs between the network nodes, with 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 a probability of 80%.
4. For the first period, the arcs' cost are randomly generated according to a uniform distribution, in the interval [100,1100]. For $t>1$, the arc's cost in period t is equal to its value in period $t-1$ 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.
6. For each facility location i and period t, consider $tend=t,\ldots,T$. For $tend=t$, the fixed costs for variables a_{i1t}^{tend} and r_{i1t}^{tend} are randomly generated according to a uniform distribution in the interval [500,3500]. For dimensions $s > 1$, the fixed costs are randomly generated considering that they are between 20% and 80% greater than the fixed costs associated with dimension $s-1$. For dimension $s = 1$, the unitary operating costs were randomly generated within the interval [1,11], using a uniform distribution. The unitary operation cost for a facility of dimension $s > 1$ is 20% to 80% less than the

unitary cost corresponding to dimension $s-1$. For *tend*>*t*, a factor between 0% and 10%, that represents an increase in the fixed cost for *tend*–1, is randomly generated.

7. The maximum capacities and the clients' demands in each time period are randomly generated, guaranteeing that the sum of the maximum capacities of all possible facilities is greater than the total demand. From one time period to the next the clients' demands can be changed by a percentage of ±10%.

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 on a Pentium 4, 1.80 Ghz, running under Windows 2000 operating system, with a maximum of 2000 MB of virtual memory and 260Mb of Ram. The heuristic was programmed using the C-language and Microsoft Visual C++ compiler. The performance of the algorithm was compared with the performance of CPLEX, version 9.0.

The test problems generated had the following dimensions: T =5 or 10; m = 10 or 20; n = 50 or 100; q = 2 or 3; *Nmax* = 1 or 2. For each set of parameters, five test problems were generated, in a total of 160 problems.

In each of the tables presented, the best, average and worst columns show the best, average and worst values obtained for each of the computational procedures tested. The summary line shows the best of the best values, the average of all the average values, and the worst of all the worst values.

Table 1 shows the quality of the primal solution found by the primal-dual heuristic, with and without the execution of a local search procedure. This local search procedure visits the neighbours of the current solution that are in its k-neighbourhood, where the k-neighbourhood is defined as follows: a feasible solution *FS*' is said to be in the k-neighbourhood of the feasible solution *FS* if and only if *FS*' differs from *FS* by the insertion or removal of at most k continuous functioning time periods to a facility i. Notice that this definition of neighbourhood does not consider as neighbours solutions that differ only in the dimensions of the facilities located. The quality of a solution is calculated as $(Z–Z_{LB})/Z_{LB}$, where Z is the objective function value of the best primal solution found and Z_{LB} is the value of the best lower bound known. Table 1 also shows the results obtained with the execution of a lagrangean heuristic. In this case, the capacity restrictions were relaxed in a lagrangean way, and the problem was solved using a dual heuristic procedure described in Dias *et al.*, 2007a, for uncapacitated problems, embedded into a subgradient iterative method.

Table 2 shows the quality of the dual solution obtained by the primal-dual heuristic and by the lagrangean heuristic. These values are calculated as $(Z^{*}-Z_{LB})/Z^{*}$, where Z^{*} represents the best primal objective function value known. Table 3 shows the computational times spent by all the procedures tested. Table 4 shows the results when Cplex is used, limiting its execution time to at most 600 seconds (about three times greater than the greatest computational time spent by the heuristic), and stopping the execution whenever Cplex finds a solution with a deviation from the best lower bound known equal to the average deviation obtained with the primal-dual heuristic, for that group of problems. In this table there is a column that shows, for each combination of parameters, the number of test problems for which Cplex was not able to find any primal solution within the time limit imposed. The last column represents the relation between the average computational times:

$$\frac{\textit{Cplex}\text{ average computational time}}{\text{Primal-dual heuristic + local search average computational time}}$$

From the computational results shown, we can conclude that the primal-dual heuristic is capable of handling efficiently the tested problems. It is able to find good primal solutions, with a better performance when compared to the lagrangean heuristic. The primal-dual heuristic is much faster than the general solver, even when Cplex is stopped after finding a primal solution of identical quality when compared to the solution build by the heuristic. The dual solutions are, in general, of bad quality.

Table 1 – Quality of the primal solution (in percentage)

					Primal-Dual Heuristic			**Primal-Dual Heuristic + Local search**			**Lagrangean Heuristic**			**Lagrangean Heuristic + Local search**		
T	*m*	*n*	*q*	*Nmax*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*
5	10	50	2	1	0.00	1.29	4.25	0.00	1.29	4.25	0.00	2.10	5.24	0.00	1.09	5.24
5	10	50	2	2	0.31	3.83	8.17	0.31	2.67	4.94	0.31	1.83	4.51	0.31	1.83	4.51
5	10	50	3	1	2.33	3.14	3.68	0.00	2.57	3.68	0.00	3.00	5.92	0.00	2.28	3.91
5	10	50	3	2	2.02	3.62	7.06	1.84	3.55	7.06	0.46	3.79	7.69	0.46	3.79	7.69
5	10	100	2	1	0.00	3.03	7.72	0.00	0.93	2.02	0.00	4.08	10.27	0.00	1.89	5.03
5	10	100	2	2	0.75	2.58	4.20	0.71	2.57	4.20	0.40	3.02	5.15	0.40	3.02	5.15
5	10	100	3	1	0.00	5.85	9.72	0.00	3.93	7.11	0.04	5.85	12.79	0.04	3.78	6.76
5	10	100	3	2	0.68	2.89	5.57	0.48	2.76	5.23	0.96	3.58	5.58	0.96	3.58	5.58
5	20	50	2	1	0.89	2.59	6.15	0.89	2.59	6.15	0.02	2.50	9.08	0.02	2.50	9.08
5	20	50	2	2	0.55	2.85	5.21	0.55	2.61	4.01	2.14	3.96	5.59	2.14	3.96	5.59
5	20	50	3	1	2.43	5.02	7.61	2.07	4.12	5.98	0.44	3.95	8.98	0.44	3.00	4.42
5	20	50	3	2	1.71	3.65	5.84	1.71	3.55	5.84	4.19	6.22	9.36	4.19	6.22	9.36
5	20	100	2	1	2.27	4.23	6.33	0.35	2.79	6.33	3.58	4.47	6.64	1.75	3.17	4.46
5	20	100	2	2	0.66	1.92	2.72	0.66	1.78	2.72	1.76	2.20	3.01	1.76	2.20	3.01
5	20	100	3	1	1.31	4.27	7.89	0.00	3.49	5.60	1.03	3.76	9.56	0.00	2.55	4.83
5	20	100	3	2	2.40	4.59	6.73	1.41	4.24	6.73	3.32	5.85	10.10	3.32	5.85	10.10
10	10	50	2	1	0.01	2.35	4.34	0.01	2.34	4.34	0.01	1.84	3.59	0.01	1.83	3.59
10	10	50	2	2	0.49	1.90	3.70	0.49	1.41	3.42	0.70	4.36	9.20	0.70	4.36	9.20
10	10	50	3	1	2.01	4.37	5.69	1.03	2.10	3.27	0.57	2.84	5.26	0.57	1.91	2.63
10	10	50	3	2	1.15	3.13	5.72	0.99	2.71	4.64	1.79	3.27	7.12	1.79	3.13	6.44
10	10	100	2	1	0.01	1.57	2.69	0.01	1.00	2.20	1.45	2.32	2.76	0.44	1.39	2.76
10	10	100	2	2	0.81	2.09	2.72	0.81	1.79	2.59	2.45	3.87	7.67	2.03	3.66	7.67
10	10	100	3	1	2.69	4.07	5.91	0.49	1.79	3.60	1.74	4.09	5.91	0.49	1.85	2.91
10	10	100	3	2	0.95	3.91	6.97	0.95	3.24	6.97	1.32	6.10	12.75	1.32	6.10	12.75
10	20	50	2	1	1.45	2.60	3.27	0.60	2.19	3.27	1.45	3.35	5.30	0.60	2.70	5.30
10	20	50	2	2	1.42	5.42	15.91	0.53	5.10	15.91	1.14	6.33	18.69	1.14	6.33	18.69
10	20	50	3	1	0.76	4.12	7.57	0.18	3.35	7.21	0.76	5.45	8.36	0.76	5.22	8.36
10	20	50	3	2	0.86	4.37	7.48	0.71	3.39	7.48	1.37	5.39	12.82	1.37	4.76	12.82
10	20	100	2	1	3.04	4.14	5.15	2.11	3.03	4.64	2.75	3.99	6.41	1.25	3.13	6.41
10	20	100	2	2	1.73	3.70	5.48	1.40	2.98	5.48	2.97	5.27	11.11	2.97	5.27	11.11
10	20	100	3	1	2.22	4.84	6.68	0.06	1.54	2.70	1.96	6.87	11.44	0.47	2.89	5.30
10	20	100	3	2	2.18	4.68	7.00	1.04	3.75	7.00	3.57	6.63	12.19	1.14	6.15	12.19
				Summary	0.00	3.52	15.91	0.00	2.72	15.91	0.00	4.13	18.69	0.00	3.48	18.69

Table 2 – Quality of the dual solution (in percentage)

					Primal-Dual Heuristic			**Lagrangean Heuristic**		
T	*m*	*n*	*q*	*Nmax*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*
5	10	50	2	1	3.61	10.68	19.37	4.03	9.87	20.88
5	10	50	2	2	11.98	15.91	21.91	2.44	14.29	31.30
5	10	50	3	1	11.48	15.05	26.47	10.10	13.81	16.10
5	10	50	3	2	7.36	16.07	23.48	1.94	14.28	24.24
5	10	100	2	1	5.27	11.70	23.05	7.09	15.46	30.29
5	10	100	2	2	9.28	17.63	27.62	11.24	24.44	38.13
5	10	100	3	1	6.29	13.10	19.41	8.57	14.05	23.29
5	10	100	3	2	8.89	11.85	16.63	7.63	13.48	19.63
5	20	50	2	1	9.57	14.97	20.49	1.48	13.13	26.60
5	20	50	2	2	8.55	14.59	19.47	1.54	18.11	30.05
5	20	50	3	1	8.11	15.35	20.44	11.06	16.80	26.21
5	20	50	3	2	9.76	11.78	14.95	12.50	16.06	23.22
5	20	100	2	1	6.26	11.06	17.86	4.88	10.66	15.84
5	20	100	2	2	8.62	12.17	15.97	11.12	15.80	20.50
5	20	100	3	1	7.62	16.06	21.12	11.53	16.19	24.43
5	20	100	3	2	11.85	15.86	20.25	14.19	21.83	31.02
10	10	50	2	1	5.77	10.94	14.75	6.15	8.17	11.51
10	10	50	2	2	5.43	9.68	19.14	1.95	10.77	24.18
10	10	50	3	1	5.43	9.86	15.57	7.74	10.31	14.38
10	10	50	3	2	4.54	6.84	8.83	1.30	4.77	8.00
10	10	100	2	1	5.84	8.75	15.43	6.63	10.92	18.11
10	10	100	2	2	2.64	13.35	34.46	0.07	6.77	12.54
10	10	100	3	1	6.28	8.28	12.20	7.61	9.53	10.72
10	10	100	3	2	5.82	10.49	19.55	0.60	5.21	7.69
10	20	50	2	1	5.32	10.64	19.08	1.03	7.09	9.76
10	20	50	2	2	6.49	10.92	13.73	3.72	10.13	17.66
10	20	50	3	1	10.52	14.18	18.19	0.28	7.17	10.74
10	20	50	3	2	5.17	7.59	14.47	0.10	7.06	17.64
10	20	100	2	1	5.01	9.82	12.64	7.39	10.40	12.85
10	20	100	2	2	4.44	11.14	25.54	0.12	9.31	31.54
10	20	100	3	1	5.63	8.72	11.32	7.45	10.64	13.81
10	20	100	3	2	7.14	11.11	16.92	4.65	12.22	21.10
				Summary	2.64	12.07	34.46	0.07	12.15	38.13

Table 3 – Computational times (in seconds)

					Primal-Dual Heuristic			**Primal-Dual Heuristic + Local search**			**Lagrangean Heuristic**			**Lagrangean Heuristic + Local search**			**Cplex**		
T	*m*	*n*	*q*	*Nmax*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*
5	10	50	2	1	0.13	0.36	0.84	0.41	0.57	0.97	0.44	1.09	1.73	0.88	1.73	2.25	0.44	46.22	191.45
5	10	50	2	2	0.36	0.67	1.19	0.70	1.05	1.30	0.95	2.62	3.47	1.91	3.47	4.39	21.48	2260.93	10983.00
5	10	50	3	1	0.22	0.49	1.05	0.59	0.87	1.28	2.38	5.25	11.02	3.73	6.51	12.48	9.91	56.02	135.92
5	10	50	3	2	0.16	1.17	2.70	1.00	1.82	3.11	3.31	8.85	24.80	5.56	10.81	26.48	113.06	267.60	693.34
5	10	100	2	1	0.50	0.68	0.99	1.17	1.49	1.91	2.08	3.82	6.30	4.17	5.63	7.14	24.98	60.58	105.72
5	10	100	2	2	1.42	2.65	4.34	2.39	3.26	4.44	3.28	7.17	10.44	6.08	8.83	10.77	30.73	726.09	2767.20
5	10	100	3	1	0.55	1.23	3.55	1.45	2.97	4.16	6.06	7.85	11.08	8.77	12.75	17.25	30.67	554.94	866.27
5	10	100	3	2	0.30	1.60	4.81	1.83	4.13	6.42	8.50	14.83	22.19	15.31	21.00	27.27	2259.44	6472.31	12276.30
5	20	50	2	1	0.28	1.20	2.19	0.98	1.89	2.83	3.41	9.92	25.80	6.30	12.50	28.17	41.25	587.75	2714.19
5	20	50	2	2	0.53	1.93	4.61	1.53	4.38	10.95	5.67	13.08	34.30	9.89	17.24	39.36	204.48	3782.21	10249.77
5	20	50	3	1	0.81	2.08	3.89	3.39	5.37	9.48	4.06	12.25	19.06	12.11	20.34	26.73	58.09	428.30	792.27
5	20	50	3	2	1.27	2.86	4.78	3.78	5.72	7.08	8.00	12.90	18.61	17.41	22.06	29.06	4633.76	15678.92	28898.99
5	20	100	2	1	1.02	1.73	2.36	2.36	5.33	9.41	5.84	13.70	17.77	11.97	22.17	29.06	85.83	104.20	159.76
5	20	100	2	2	1.44	4.21	8.19	4.42	7.89	12.97	12.83	17.64	22.88	23.92	28.28	31.92	2593.66	37562.57	137829.75
5	20	100	3	1	2.50	6.15	10.81	9.13	11.51	13.75	3.97	30.95	65.36	22.61	50.13	76.23	97.13	501.49	1943.77
5	20	100	3	2	4.78	16.17	38.41	10.69	22.12	41.91	25.53	43.03	69.56	45.00	64.46	84.14	2544.06	30310.08	69891.59
10	10	50	2	1	0.24	1.19	4.34	0.75	1.97	4.61	1.81	9.88	18.55	7.13	13.72	20.17	7.67	191.02	1330.44
10	10	50	2	2	0.16	1.02	2.42	1.44	3.10	6.38	3.34	8.79	13.17	11.92	15.59	19.53	124.61	7810.15	18430.33
10	10	50	3	1	0.97	1.68	3.02	4.09	7.08	10.97	7.58	17.13	31.53	25.00	30.50	42.14	62.70	693.62	2195.17
10	10	50	3	2	0.23	2.21	5.75	2.53	6.56	12.99	5.66	25.47	51.92	18.81	45.88	68.67	1563.01	32558.67	98431.45
10	10	100	2	1	0.58	1.82	3.05	1.95	6.87	9.94	7.00	8.80	12.17	19.50	23.65	27.44	214.78	374.89	796.45
10	10	100	2	2	0.47	3.80	9.42	4.53	9.03	18.84	14.99	24.93	43.89	30.64	45.75	63.55	29.14	30588.27	132207.80
10	10	100	3	1	2.42	5.25	9.38	10.69	16.83	27.58	20.16	34.74	59.44	53.45	71.04	90.50	204.14	837.88	3237.26
10	10	100	3	2	3.03	8.19	13.31	7.84	24.36	57.11	37.81	73.96	122.17	80.74	115.02	158.83	3455.34	125059.55	224544.61
10	20	50	2	1	0.97	3.04	6.22	7.36	11.80	20.81	10.52	31.05	91.94	28.59	56.49	116.31	131.97	974.98	2910.52

Table 3 – Computational times (in seconds) (cont.)

					Primal-Dual Heuristic			**Primal-Dual Heuristic + Local search**			**Lagrangean Heuristic**			**Lagrangean Heuristic + Local search**			**Cplex**		
T	*m*	*n*	*q*	*Nmax*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*	*Best*	*Average*	*Worst*
10	20	50	2	2	0.22	3.92	7.09	7.14	10.96	14.55	12.56	36.69	92.50	39.72	64.49	121.05	0.00	3445.21	6486.00
10	20	50	3	1	2.42	9.91	21.92	8.31	26.66	60.92	35.38	126.08	315.61	74.63	169.03	357.94	319.56	1284.30	7694.13
10	20	50	3	2	1.39	6.32	12.42	17.63	32.17	65.58	33.69	53.67	78.74	106.81	147.20	257.77	43718.31	87540.41	175002.00
10	20	100	2	1	2.17	7.34	12.66	22.45	39.63	58.19	14.98	43.13	72.67	90.56	126.27	193.39	240.38	53154.97	103354.49
10	20	100	2	2	2.56	12.26	24.94	36.14	63.64	121.58	37.03	92.88	135.97	125.77	173.30	224.84	904.22	43921.56	175005.77
10	20	100	3	1	7.66	27.85	69.72	49.09	131.36	192.66	43.33	82.96	163.58	212.74	310.83	503.59	1343.20	11208.79	39835.94
10	20	100	3	2	6.11	33.07	84.38	62.22	132.34	203.91	104.20	124.02	155.75	262.97	321.72	453.52	15039.22	80031.52	175004.94
Summary					0.13	5.44	84.38	0.41	18.90	203.91	0.44	31.22	315.61	0.88	63.70	503.59	0.00	18096.13	224544.61

Table 4 – Computational times when Cplex is used considering as termination criteria a maximum time of 600 seconds or finding a primal solution of quality equal to the average deviation obtained by the primal-dual heuristic

T	*m*	*n*	*q*	*Nmax*	*Average Computational Times*	*Number of Unsolved Problems*	*Relation*
5	10	50	2	1	40.50	0	71.57
5	10	50	2	2	118.87	0	113.17
5	10	50	3	1	34.08	0	39.23
5	10	50	3	2	83.92	0	46.22
5	10	100	2	1	36.44	0	24.45
5	10	100	2	2	31.76	1	9.74
5	10	100	3	1	149.49	1	50.25
5	10	100	3	2	---	5	---
5	20	50	2	1	12.89	0	6.81
5	20	50	2	2	127.76	0	29.18
5	20	50	3	1	70.94	0	13.21
5	20	50	3	2	---	5	---
5	20	100	2	1	38.37	0	7.21
5	20	100	2	2	121.12	3	15.34
5	20	100	3	1	124.14	0	10.79
5	20	100	3	2	318.12	2	14.38
10	10	50	2	1	56.13	0	28.56
10	10	50	2	2	126.73	0	40.92
10	10	50	3	1	34.61	0	4.89
10	10	50	3	2	367.44	4	56.02
10	10	100	2	1	234.29	0	34.09
10	10	100	2	2	112.66	2	12.47
10	10	100	3	1	172.40	0	10.24
10	10	100	3	2	378.72	4	15.55
10	20	50	2	1	47.33	0	4.01
10	20	50	2	2	54.56	0	4.98
10	20	50	3	1	200.13	1	7.51
10	20	50	3	2	273.56	3	8.50
10	20	100	2	1	113.11	0	2.85
10	20	100	2	2	284.71	1	4.47
10	20	100	3	1	249.03	4	1.90
10	20	100	3	2	---	5	---
Global Average Values					138.41	1.28	23.74

We have also tested the use of memetic algorithms to this problem, namely by directly adapting the algorithm developed in Dias *et al.*, 2007b, by creating dummy facilities. This approach showed to be unfit, because the memetic algorithm is capable of finding solutions that are, on average, 6% far from the optimal, but needs huge computational times (most of the times greater than Cplex computational times).

6 Conclusions

The mathematical model presented in this paper describes a dynamic location problem where the total available capacity can increase or decrease from one time period to the next, by opening or closing facilities of equal or different dimensions.
The computational tests performed showed that the primal-dual heuristic developed is capable of solving efficiently the problem formulated.
It would be interesting to apply other optimization techniques, namely metaheuristics, to the problem formulated, because it is unlikely that the primal-dual heuristic developed is capable of handling problems where the number of possible dimensions for facilities' capacities is big. As the algorithm developed in Dias *et al.*, 2007b, successfully used in solving other dynamic location problems, produced unsatisfactory results, a different approach is being thought, being the configuration of the chromosomes composition the biggest challenge.

7 References

Beasley, J. E. (1988) An Algorithm for Solving Large Capacitated Warehouse Location Problems, European Journal of Operational Research, Vol. 33, pp 314-325.

Canel, C., Khumawala, B., Law, J., Loh, A. (2001) An Algorithm for the Capacitated, Multi-Commodity, Multi-Period Facility Location Problem, Computers & Operations Research, Vol. 28, pp 411-427.

Christofides, N., Beasley, J. E. (1983) Extensions to a Lagrangean Relaxation Approach for the Capacitated Warehouse Location Problem, European Journal of Operational Research, Vol. 12, pp 19-28.

Cornuejols, G., Sridharan, R., Thizy, J. M. (1991) A Comparison of Heuristics and Relaxations for the Capacitated Plant Location Problem, European Journal of Operational Research, Vol. 50, pp 280-297.

Dias, J., Captivo, M.E., Clímaco, J. (2006) Capacitated Dynamic Location Problems with Opening, Closure and Reopening of Facilities, IMA Journal of Mathematical Management, Special Issue on Location Analysis: Applications and Models, S. Salhi e Zvi Drezner (eds), Vol. 17, No 4, pp 317-348.

Dias, J., Captivo, M.E., Clímaco, J. (2007a) Efficient Primal-Dual Heuristic for a Dynamic Location Problem, Computers & Operations Research, Vol. 34, pp1800-1823.

Dias, J., Captivo, M.E., Clímaco, J. (2007b) A Memetic Algorithm for Dynamic Location Problems, Meta heuristics: Progress in Complex Systems Optimization, Operations Research/Computer Science Interfaces Series, Vol. 39, Karl F. Doerner, Michel Gendreau, Peter Greistorfer, Walter J. Gutjahr, Richard F. Hartl, Marc Reimann (edts), pp 225-244.

Erlenkotter, D. (1975) Capacity Planning for Large Multilocation Systems: Approximate and Incomplete Dynamic Programming Approaches, Management Science, Vol. 22, pp 274-285.

Erlenkotter, D. (1978) A Dual-based Procedure for Uncapacitated Facility Location, Operations Research, Vol. 26, pp 992-1009.

Fong, C. O., Srinivasan, V. (1981a) The Multiregion Dynamic Capacity Expansion Problem- Part I, Operations Research, Vol. 29, pp 787-799.

Fong, C. O., Srinivasan, V. (1981b) The Multiregion Dynamic Capacity Expansion Problem- Part II, Operations Research, Vol. 29, pp 800-816.

Freindenfelds, J. (1981a) Near-Optimal Solution of a Two-Type Capacity Expansion Problem, Computers & Operations Research, Vol. 8, pp 221-239.

Freindenfelds, J. (1981b) Capacity Expansion, Analysis of Simple Models with Applications, Elsevier North Holland.

Ghiani, G., Guerriero, F., Musmanno, R. (2002) The Capacitated Plant Location Problem with Multiple Facilities in the Same Site, Computers & Operations Research, Vol. 29, pp 1903-1912

Guignard, M., Spielberg, K. (1979) A Direct Dual Method for the Mixed Plant Location Problem with Some Side Constraints, Mathematical Programming, Vol. 17, pp 198-228.

Hinomoto, H. (1965) Capacity Expansion with Facilities under Technological Improvement, Management Science, Vol. 11, pp 581-592.

Jacobsen, S. K. (1983) Heuristics for the Capacitated Plant Location Model", European Journal of Operational Research, Vol. 12, pp 253-261.

Lee, C. (1991) An Optimal Algorithm for the Multiproduct Capacitated Facility Location Problem with a Choice of Facility Type, Computers & Operations Research, Vol. 18, pp 167-182.

Luss, H. (1982) Operations Research and Capacity Expansion Problems, Operations Research, Vol. 30, pp 907-947.

Mazzola, J., Neebe, A. (1999) Lagrangian - Relaxation - Based Solution Procedures for a Multiproduct Capacitated Facility Location Problem with Choice of Facility Type, European Journal of Operational Research, Vol. 115, pp 285-299.

Min, H. (1988) The Dynamic Expansion and Relocation of Capacitated Public Facilities: a Multi-Objective Approach, Computers & Operations Research, Vol. 15, pp 243-252.

Saldanha-da-Gama, F., Captivo, M.E. (2002) A Branch-and-Bound Procedure for the Multi-Period Capacitated Location Problem, Working Paper 8/2002, Centro de Investigação Operacional - Faculdade de Ciências da Universidade de Lisboa.

Shulman, A. (1991) An Algorithm for Solving Dynamic Capacitated Plant Location Problems with Discrete Expansion Sizes, Operations Research, Vol. 39, pp 423-436.

Smith, R. (1981) Planning Horizons for the Deterministic Capacity Problem, Computers & Operations Research, Vol. 8, pp 209-220.

Sridharan, R., (1991) A Lagrangian Heuristic for the Capacitated Plant Location Problem with Side Constraints, Journal of the Operational Research Society, Vol. 42, pp 579-585.

Sridharan, R., (1995) The Capacitated Plant Location Problem, European Journal of Operational Research, Vol. 87, pp 203-213.

Van Roy, T. J. (1986) A Cross Decomposition Algorithm for Capacitated Facility Location, Operations Research, Vol. 34, pp 145-163.

Van Roy, T., Erlenkotter, D. (1982) A Dual-Based Procedure for Dynamic Facility Location, Management Science, Vol. 28, pp 1091-1105.

Wirasinghe, S. C, Waters, N. M. (1983) An Approximate Procedure for Determining the Number, Capacities and Locations of Solid Waste Transfer-Stations in an Urban Region, European Journal of Operational Research, Vol. 12, pp 105-111.