
Volume 44
April 2014
ISSN 0305-0548
ELSEVIER
computers & operations research
Editor: Stefan Nickel
Available online at www.sciencedirect.com
ScienceDirect

Computers & Operations Research 44 (2014) 193–205

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

A dual bounding scheme for a territory design problem

Mónica G. Elizondo-Amaya [a,*], Roger Z. Ríos-Mercado [a], Juan A. Díaz [b]

[a] Graduate Program in Systems Engineering, Universidad Autónoma de Nuevo León, AP 111-F, Cd. Universitaria, San Nicolás de los Garza, Nuevo León 66450, Mexico
[b] Department of Industrial and Mechanical Engineering, Universidad de las Américas Puebla, Sta. Catarina Mártir, San Andrés Cholula, Puebla 72820, Mexico

ARTICLE INFO

Available online 16 November 2013

Keywords:
Commercial territory design
Discrete location
Lagrangian relaxation
Dual bounding scheme

ABSTRACT

In this work, we present a dual bounding scheme for a commercial territory design problem. This problem consists of finding a p-partition of a set of geographic units that minimizes a measure of territory dispersion, subject to multiple balance constraints. Dual bounds are obtained using binary search over a range of coverage distances. For each coverage distance a Lagrangian relaxation of a maximal covering model is used effectively. Empirical evidence shows that the bounding scheme provides tighter lower bounds than those obtained by the linear programming relaxation. To the best of our knowledge, this is the first study about dual bounds ever derived for a commercial territory design problem.

1. Introduction

Territory design can be viewed as the problem of grouping small geographical areas, called *basic areas*, into larger geographic clusters called *territories* according to specific planning criteria. These problems arise in different applications such as political districting [17,23,24,35,30,2,3] and sales territory design [43,45,46,14,9,22] to name the most relevant. An extensive survey on general territory design problems and their approaches can be found in Kalcsics et al. [26] and Duque et al. [10].

The problem addressed in this paper is motivated by a concrete practical application from a local beverage firm. To improve customer supply, the company needs to divide the set of city blocks (or basic units) in the city area into a specific number of disjoint territories. In particular, the planning requirements considered in this problem are territory compactness and territory balancing with respect to two activity measures present at every basic unit. The former criterion means that customers within a territory are relatively close to each other while the latter requirement refers to creating territories of about equal size in terms of both number of customers and product demand. This problem can be classified as a commercial territory design problem (TDP) for which related versions under different requirements have been addressed in literature from both exact and heuristic approaches.

Typically, the problem is modeled as minimizing a dispersion measure subject to some planning requirements such as connectivity and territory balancing. The connectivity requirement implies that basic units (BUs) that are assigned to the same territory must reach each other by traveling within the territory. Depending on how the dispersion measure objective is chosen, we can further classify these TDP models as p-median TDPs (PMTDP) and p-center TDPs (PCTDP). Heuristic methods have been developed for both different versions PCTDPs and PMTDPs. Ríos-Mercado and Fernández [36] introduced the PCTDP subject to connectivity and multiple balance constraints. They propose a Reactive GRASP to solve the problem. Their proposed approach obtained solutions of much better quality (in terms of dispersion measure and the balancing requirements) than those found by the company method in relatively fast computation times.

Later, Caballero-Hernández et al. [4] study other version of the commercial PCTDP model that includes additional joint assignment constraints which means that some units are required to belong to the same territory. In that work, the authors develop a metaheuristic solution approach based on GRASP. Experimental results show the effectiveness of their method in finding good-quality solutions for instances up to 500 BUs and 10 territories in reasonably short computation times. Particularly, a very good performance is observed within the local search procedure, which produces an improvement of about 90% in solution quality.

Ríos-Mercado and Salazar-Acosta [38] address an extension of the TDP that considers requirements about design and routing in territories. In contrast to the TDP variations described above, the authors use network-based distances between BUs (instead of Euclidean distances) and a diameter-based function to measure territory dispersion. To solve this problem, the authors proposed a GRASP that incorporates advanced features such as adaptive memory and strategic oscillation. Empirical evidence shows that the incorporation of these two components into the procedure had a very positive impact on both obtaining feasible solutions and improving solution quality.

* Corresponding author. Tel.: +52 18186877231.
E-mail addresses: moni@yalma.fime.uanl.mx (M.G. Elizondo-Amaya), roger.rios@uanl.edu.mx (R.Z. Ríos-Mercado), juana.diaz@udlap.mx (J.A. Díaz).

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.11.006

Salazar-Aguilar et al. [39] present an exact optimization framework based on branch and bound and cut generation for tackling relatively small instances of several TDP models. Particularly, they studied both, the PCTDP and PMTDP models. They successfully solved instances of up to 100 BUs for the PCTDP and up to 150 BUs for the PMTDP. The authors also propose new integer quadratic programming models that allowed to efficiently solve larger instances by commercial MINLP solvers. For IQPs models, they obtained locally optimal solutions for instances with up to 500 BUs and 12 territories.

Ríos-Mercado and López-Pérez [37] and López-Pérez and Ríos-Mercado [28] address a commercial TDP with additional side constraints such as disjoint assignment requirements and similarity with existing plan. In their work, they assume a fixed set of centers, and present several heuristic algorithmic strategies for solving the allocation phase.

Recently, a bi-objective TDP model was introduced by Salazar-Aguilar et al. [40], where an ε−constraint method is developed for tackling small- to medium-scale instances from an exact optimization perspective. In that work, two different measures of dispersion are studied, one based on the p-center problem objective and the other based on the p-median objective model. It was shown how the latter had a tighter LP relaxation that allowed to solve larger instances. The proposed method was successful for finding optimal Pareto frontiers on instances from 60 up to 150 BUs and 6 territories. It was also clear that larger instances were indeed intractable, thus justifying the use of heuristic approaches proposed by Salazar-Aguilar et al. in [41,42]. In these works, the authors address the development of GRASP and Scatter Search (SS) strategies to handle considerably large instances. These proposed heuristic procedures outperformed two of the well-known and most successful multiobjective algorithms in the field, the Non-dominated Sorting Genetic Algorithm (NSGA-II) by Deb et al. [8] and the Scatter Tabu Search Procedure for Multiobjective Optimization (SSPMO) by Molina et al. [32].

As it can be seen, from literature, practically all of the work on commercial territory design has focused on developing heuristics for finding good feasible solutions to large instances in reasonable times due to the well established NP-completeness of both PCTDP and PMTDP [36,39]. However, thus far, the quality of the solutions obtained by these heuristic methods has not been properly assessed since the quality of the lower bound provided by the linear programming relaxation of TDP models is very poor. To the best of our knowledge, no dual bounding schemes have been developed for any of the commercial TDP models found in the literature. It is worth mentioning that besides being useful in evaluating the quality of heuristic solutions, dual bounds are also the foundations in the development of exact solution methods.

Therefore, the main contribution of this work is the introduction and development of the first dual bounding scheme for a commercial territory design problem. The TDP addressed here considers balance and compactness requirements. This scheme is motivated by exact solution methodologies already found in literature for related location problems, where the main idea is to generate and solve a set of auxiliary problems. Particularly, Albareda-Sambola et al. [1] propose a successful exact solution method for the capacitated p-center problem (CpCP) that involves a procedure for obtaining lower bounds for this problem. The bounding procedure developed in [1] is not quite applicable for our problem; however, given the strong similarities, one of the goals of this paper is to extend this bounding procedure to handle multiple balance constraints.

The proposed algorithm performs a binary search over a specific set of covering radii extracted from the distances matrix and solves for each of them a Lagrangian dual problem based on a maximal demand covering problem. The evaluation of this dual problem for a given radius δ can determine, under certain conditions, when such covering radius is a dual bound for TDP. An empirical study was carried out on a collection of data instances. The results show the effectiveness of the developed scheme as it considerably outperforms the linear programming relaxation dual bound.

The paper is structured as follows. Section 2 defines the problem formally and describes the mathematical formulation. Section 3 presents the dual bounding scheme and each of its components. Experimental work is included in Section 4. Finally, conclusions and some final remarks are drawn in Section 5.

2. Problem description

Let V be a set of nodes or BUs representing city blocks. Let w_i^a be the measure of activity a in block i, $a \in A = \{1,2\}$ where $a=1$ denotes number of customers and $a=2$ denotes product demand. Let d_{ij} be the Euclidean distance between each pair of basic units i and j. The number of territories is given by p. A territory design configuration is a p-partition of the set V. Let $w^a(V_k) = \sum_{i \in V_k} w_i^a$ be the size of territory $V_k \subseteq V$ with respect to activity a. A solution to this problem must have balanced territories with respect to each activity. Due to the discrete nature of the problem and to the unique assignment constraints, it is practically impossible to get perfectly balanced territories. Thus, in order to address this issue, a tolerance parameter τ^a for each activity a is introduced. This tolerance parameter is user specified and it represents a limit on the maximum deviation allowed from an ideal target. This target value is given by the average size $\mu^a = w^a(V)/p$. Finally, in each of the territories, basic units must be relatively close to each other. To account for this, in this work we use a dispersion function based on the p-center problem objective.

All parameters are assumed to be known with certainty. Therefore, the problem can be formally described as finding a p-partition of a set V of basic units that meets multiple balance constraints and minimizes a dispersion measure.

2.1. Integer programming formulation

To state the model mathematically, we define the following notation:

Indices and sets

V	set of BUs,
A	set of BUs activities,
i,j	BUs indices; $i,j \in V = \{1,2,\ldots,n\}$,
a	activity index; $a \in A = \{1,2\}$.

Parameters

n	number of BUs,
p	number of territories,
w_i^a	value of activity a in node i; $i \in V$, $a \in A$,
d_{ij}	Euclidean distance between i and j; $i,j \in V$,
τ^a	relative tolerance with respect to activity a; $a \in A$, $\tau^a \in [0,1]$.
μ^a	$w^a(V)/p$, average (target) value of activity a; $a \in A$.

Although the practical decision does not require to place facilities on centers as it is done in location problems, we used binary decision variables based on centers because they allowed to model territory dispersion appropriately.

Decision variables:

$$x_{ij} = \begin{cases} 1 & \text{if BU } j \text{ is assigned to territory with center in BU } i, \\ 0 & \text{otherwise.} \end{cases}$$

With this notation our commercial TDP can be formulated as the following MILP:

(TDP)

$$\text{Minimize} \quad f(x) = \max_{i,j \in V}\{d_{ij}x_{ij}\} \tag{1}$$

$$\text{subject to} \quad \sum_{i \in V} x_{ij} = 1, \quad j \in V, \tag{2}$$

$$\sum_{i \in V} x_{ii} = p, \tag{3}$$

$$\sum_{j \in V} w_j^a x_{ij} \geq (1-\tau^a)\mu^a x_{ii}, \quad i \in V; \ \ a \in A, \tag{4}$$

$$\sum_{j \in V} w_j^a x_{ij} \leq (1+\tau^a)\mu^a x_{ii}, \quad i \in V; \ \ a \in A, \tag{5}$$

$$x_{ij} \in \{0,1\}, \quad i,j \in V. \tag{6}$$

Objective (1) measures territory dispersion. Constraints (2) guarantee that each basic unit j is assigned to only one territory. Constraint (3) assures the creation of exactly p territories. Constraints (4) and (5) represent the territory balance with respect to each activity measure as they establish that the size of each territory must lie within a range (measured by a tolerance parameter τ^a) around its average size (μ^a). Moreover, the upper bound balance constraints (5) also ensure that if no center is placed at i, no customer can be assigned to it (i.e., $x_{ii}=0 \Rightarrow x_{ij}=0$, $\forall i,j \in V$). Finally, constraints (6) define the binary nature of the decision variables.

The model can be viewed in terms of integer programming as a vertex p-center problem with multiple capacity constraints (5) and with additional constraints (4). Given that even the uncapacitated vertex p-center problem is NP-hard [27], it follows that our commercial TDP is also NP-hard. Our model is derived from the model introduced by Ríos-Mercado and Fernández [36] that includes additional planning requirements.

3. The dual bounding scheme

The bounding framework proposed in this work follows the methodology that underlies a wide range of successful exact and approximate solution approaches for p-center problems. These problems are most often solved through generation and solution of a sequence of *auxiliary problems* that keep a strong structural relation with the p-center problem and assure an optimal solution to the original problem. In this case, the use of an auxiliary problem allows achieving the same goal through simplest equivalent formulations. Different auxiliary problems have been proposed, mostly related to coverage problems such as the *location set covering problem* [44] and the *maximal covering location problem* [5]. Successful techniques for the p-center problem use a common principle to perform an iterative search over a range of coverage distances searching for the smallest radius such that the optimal solution to the associated auxiliary problem provides a feasible solution to the p-center problem. Representative works for uncapacitated p-center problem can be found in Minieka [31], Daskin [6,7] and Elloumi et al. [11]. For the capacitated version (CpCP), which has been less studied, Özsoy and Pinar [34] and Albareda-Sambola et al. [1] propose exact solution algorithms where the latter presents the best results so far. In [1], they addressed two auxiliary problems (arising from both set and maximal covering problems) and analyzed two different strategies for solving exactly CpCP, based on binary search and sequential search. Given that the CpCP is a substructure of the TDP model, this paper exploits the knowledge generated in [1] for deriving dual bounds for the TDP.

In order to introduce the proposed scheme, we highlight the following remarks from the TDP formulation discussed in the previous section.

Remark 1.

- Let $\overline{D} = \{d_0, d_1, \ldots, d_{k_{\max}}\}$ be the set of the $k_{\max}$ different values of the distance matrix $D=(d_{ij})$ sorted by non-decreasing values ($d_0 < d_1 < \cdots < d_{k_{\max}}$), and let $K=\{0,1,\ldots,k_{\max}\}$ be the corresponding index set in $\overline{D}$. Given the nature of the objective function, which minimizes the maximum distance between a basic unit and the territory center to which it is assigned, it can be seen that the optimal value of TDP is an element of $\overline{D}$.
- If d_{k^*} is the optimal value of TDP for some index $k^* \in K$, note that any $d_k \in \overline{D}$ with $k \leq k^*$ ($k \geq k^*$) is a lower (upper) bound on the optimal value d_{k^*}.

Therefore, the algorithm relies on an iterative search procedure that attempts to find the best lower (dual) bound by exploring the set of distances in $\overline{D}$. At each iteration, it sets a threshold distance which is used as the coverage radius of an associated covering problem. This auxiliary problem allows to determine when it is not possible to assign all basic units into p or less territories within such radius, yielding therefore a valid dual bound on the optimal value of TDP. In this section we detail the components of this dual bounding procedure.

3.1. The maximum demand covering problem

From the TDP, we derive an auxiliary problem which gives an answer as to whether we can assign all basic units within a certain radius δ into at most p territories, the *maximum demand covering problem*. This problem operates with a fixed maximum distance δ known as *covering radius* and considers the objective of maximizing the total amount of covered demand when at most p territory centers are located. This auxiliary problem can be seen as an extension of a well-known problem from location optimization literature, the maximal covering location problem (MCLP) [5], as we consider additional capacity constraints (4)–(5).

To formulate the model we will use the following additional notation:

$I_\delta(j) = \{i \in V : d_{ij} \leq \delta\}$,

$J_\delta(i) = \{j \in V : d_{ij} \leq \delta\}$,

$$b_i^{(\delta,a)} = \min\left\{(1+\tau^a)\mu^a, \sum_{j \in J_\delta(i)} w_j^a\right\},$$

where $I_\delta(j)$ denotes the set of territory centers whose distance to basic unit j does not exceed the radius δ. Similarly, for a given territory center i, $J_\delta(i)$ denotes the set of basic units whose distance to i does not exceed the radius δ. Additionally, the parameter $b_i^{(\delta,a)}$ has the purpose of strengthening the model since it fits the upper limit of activity measures for territory balance constraints (5). The maximum demand covering problem henceforth denoted as MDCP_δ can be formulated as follows:

(MDCP_δ)

$$W(\delta) = \text{Maximize} \quad f(x) = \sum_{i \in V} \sum_{j \in J_\delta(i)} w_j^1 x_{ij} \tag{7}$$

$$\text{subject to} \quad \sum_{i \in I_\delta(j)} x_{ij} \leq 1, \quad j \in V, \tag{8}$$

$$\sum_{i \in V} x_{ii} \leq p, \tag{9}$$

$$\sum_{j \in J_\delta(i)} w_j^a x_{ij} \geq (1-\tau^a)\mu^a x_{ii}, \quad i \in V; \quad a \in A, \tag{10}$$

$$\sum_{j \in J_\delta(i)} w_j^a x_{ij} \le b_i^{(\delta,a)} x_{ii}, \quad i \in V; \ a \in A, \tag{11}$$

$$x_{ij} \in \{0,1\}, \quad i \in V; \ j \in J_\delta(i). \tag{12}$$

The objective function (7) maximizes the total amount of demand or product demand (i.e., activity measure $a=1$) that can be covered. By constraints (8) each customer is assigned to at most one territory. Constraints (10) and (11) conform the territory balance constraints, which are referred to as *minimum and maximum territory capacity constraints*, respectively. In particular, constraints (11) also guarantee that if no center is placed at i, no customer can be assigned to it. Finally, constraint (9) assures the creation of at most p territories. Then, the maximum demand covering problem consists of maximizing the total demand of BUs that can be satisfied with at most p territories within a given maximum distance δ.

We investigate now the relation between TDP and MDCP$_\delta$. Let $W_{\text{tot}} = \sum_{j \in V} w_j^1$ be the sum of demand corresponding to activity measure $a=1$ (i.e., product demand) over all basic units. When solving MDCP$_\delta$ we have the following cases.

Case 1: If for some $k \in K$, the total demand that can be satisfied within a radius $d_k \in \overline{D}$ is W_{tot} and p territory centers are selected, then all BUs have been assigned and the assignment obtained from MDCP$_\delta$ is a feasible solution for TDP. Therefore, the radius d_k is a valid upper bound on the optimal value of TDP.

Case 2: The optimal solution to TDP can be obtained through the auxiliary problem MDCP$_\delta$ by finding the smallest coverage radius where all the BUs can be assigned (i.e., the smallest index $k \in K$ such that $W(d_k) = W_{\text{tot}}$) and exactly p territory centers are selected. Note that an optimal solution to MDCP$_\delta$, $\delta \in \overline{D}$, that has all the BUs assigned with strictly less than p territories is possible. However, the number of territories required to cover the maximum amount of demand increases when the coverage radius decreases. Thus, a smaller radius $\delta^* \in \overline{D}$ can always be found such that the optimal solution to MDCP$_{\delta^*}$ is still covering the total demand W_{tot} using exactly p territories (otherwise the TDP would be unfeasible).

Case 3: If for some $k \in K$, $W(d_k) < W_{\text{tot}}$, it can be seen that it is not possible to assign all BUs within such covering radius and therefore the radius d_k is a valid lower bound on the optimal value of TDP.

An advantage of MDCP$_\delta$ is that its objective function $W(\delta)$ determines when δ is either a bound (dual or primal) or the optimal value for the TDP, depending on the number of BUs that were assigned in the MDCP$_\delta$ optimal solution. Also note that, without loss of generality, activity 2 can be alternatively used instead of activity 1 in the objective function $W(\delta)$ and by using $W_{\text{tot}}^2 = \sum_{j \in V} w_j^2$ the just described cases still apply.

Given that MCLP is NP-hard [29], it follows that MDCP$_\delta$ is also NP-hard. Exact solution methods developed for MCLP are not applicable to MDCP$_\delta$ unless they are adapted to handle its specific features. Moreover, even medium size instances of the problem addressed in this work are practically intractable by such solution techniques. Therefore, instead of solving MDCP$_\delta$ exactly, a Lagrangian relaxation to obtain a valid upper bound for MDCP$_\delta$ is derived.

Proposition 3.1. *Let* $\overline{W}(\delta)$ *be an upper bound for* MDCP$_\delta$, *if* $\overline{W}(\delta) < W_{\text{tot}}$, *then the coverage radius* δ *is a valid lower bound on the optimal value of TDP.*

Proof. Let X_δ be the optimal solution to MDCP$_\delta$ with corresponding optimal objective function value given by $W(\delta)$. It is easy to check that $W(d_0) \le W(d_1) \cdots \le W(d_{k_{\max}})$, where $d_k \in \overline{D}$, $k \in K$. Now we establish a more precise relationship between the optimal solutions of problems MDCP$_\delta$ and TDP.

Let k^* be the smallest index $k \in K$ such that $W(d_{k^*}) = W_{\text{tot}}$ and exactly p territories are created. Note that territory balance constraints are also present in the MDCP$_\delta$ formulation. On the other hand, for $X_{d_{k^*}} = (x_{ij}^*)$, the optimal solution to MDCP$_{d_{k^*}}$, TDP constraints of unique assignment (2) are satisfied since

$$\sum_{i \in V} \sum_{j \in J_{d_{k^*}}(i)} w_j^1 x_{ij}^* = \sum_{j \in V} \left(w_j^1 \sum_{i \in I_{d_{k^*}}(j)} x_{ij}^* \right) = W_{\text{tot}},$$

$$\Rightarrow \sum_{j \in V} \left(w_j^1 \sum_{i \in I_{d_{k^*}}(j)} x_{ij}^* \right) = \sum_{j \in V} w_j^1,$$

$$\Rightarrow \sum_{i \in I_{d_{k^*}}(j)} x_{ij}^* = 1.$$

Notice that $x_{ij} = 0$, $\forall i \notin I_{d_{k^*}}(j)$, then we have that

$$\sum_{i \in V} x_{ij}^* = \sum_{i \in I_{d_{k^*}}(j)} x_{ij}^* = 1,$$

$$\Rightarrow \sum_{i \in V} x_{ij}^* = 1.$$

Therefore, as d_{k^*} is the smallest coverage radius in $\overline{D}$ such that MDCP$_{d_{k^*}}$ solution satisfies all TDP constraints, it follows that d_{k^*} is the optimal value of TDP and $X_{d_{k^*}}$ its optimal solution.

Finally, it can be noticed that for all $k \in K$ such that $k \le k^*$, the radius d_k is a valid lower bound for TDP and further, $W(d_k) \le W_{\text{tot}}, \forall k \le k^*, k \in K$. Notice that in the general case, as $\overline{W}(\delta)$ is an upper bound on the optimal value of MDCP$_\delta$, $\overline{W}(\delta) \le W_{\text{tot}}$ implies that $W(\delta) \le W_{\text{tot}}$ and case 3 holds for any TDP relaxation. □

Next, we detail the relaxation of MDCP$_\delta$ used in order to obtain the upper bound $\overline{W}(\delta)$, for a given coverage radius δ.

3.2. *Lagrangian relaxation of* MDCP$_\delta$

In this section we propose a relaxation of MDCP$_\delta$ which consists of relaxing the assignment constraints (8) in a Lagrangian fashion, i.e., incorporating them into the objective function with the corresponding multipliers $\lambda \in \mathbb{R}_+^{|V|}$. For surveys on Lagrangian relaxation, the reader is referred to Guignard [19], Geoffrion [18] and Fisher [12,13]. The resulting model is

$(L_\delta(\lambda))$

$$\text{Maximize} \quad Z_{LR}(\lambda) = \sum_{i \in V} \sum_{j \in J(\delta)} w_j^{a^*} x_{ij} + \sum_{j \in V} \lambda_j \left(1 - \sum_{i \in I(\delta)} x_{ij}\right)$$

$$= \sum_{j \in V} \lambda_j + \max\left\{ \sum_{i \in V} \sum_{j \in J(\delta)} (w_j^{a^*} - \lambda_j) x_{ij} \right\}$$

subject to

$$\sum_{i \in V} x_{ii} \le p,$$

$$\sum_{j \in J_\delta(i)} w_j^a x_{ij} \ge (1-\tau^a)\mu^a x_{ii}, \quad i \in V; \ a \in A,$$

$$\sum_{j \in J_\delta(i)} w_j^a x_{ij} \le b_i^{(\delta,a)} x_{ii}, \quad i \in V; \ a \in A,$$

$$x_{ij} \in \{0,1\}, \quad i \in V; \ j \in J_\delta(i).$$

The Lagrangian problem $L_\delta(\lambda)$ consists of maximizing a weighted sum over the variables $x_{ij}, i, j \in V$, under constraints of minimum and maximum territory capacity and the selection of p territory centers. Notice that the model $L_\delta(\lambda)$ can be decomposed into $|V|$ independent subproblems, one for each $i \in V$, as follows:

(TSKP_i)

$$\text{Maximize} \quad \upsilon_i(\lambda, x) = \sum_{j \in J_\delta(i)} (w_j^{a^*} - \lambda_j) x_{ij} \tag{13}$$

subject to $\sum_{j \in J_\delta(i)} w_j^a x_{ij} \geq (1-\tau^a)\mu^a x_{ii}, \quad a \in A,$ (14)

$$\sum_{j \in J_\delta(i)} w_j^a x_{ij} \leq b_i^{(\delta,a)} x_{ii}, \quad a \in A, \tag{15}$$

$$x_{ij} \in \{0,1\}, \quad j \in J_\delta(i). \tag{16}$$

Each of these subproblems can be seen as a knapsack problem with double constraints of minimum and maximum capacity, or as a bidimensional knapsack problem with additional constraints (14). We denote this subproblem as *Two-Sided Knapsack Problem* (TSKP). Hence, to solve $L_\delta(\lambda)$, for each $i \in V$ its corresponding subproblem TSKP$_i$ is solved. Then, in order to meet constraint (9), the indices in the set V are sorted in non-increasing values of TSKP$_i$, that is,

$$v_{i_1}(\lambda,x) \geq v_{i_2}(\lambda,x) \geq \cdots \geq v_{i_{|V|}}(\lambda,x).$$

Then, the first p^* indices are chosen as territory centers, where p^* is given as follows:

$$p^* = \min\{p, \max\{r : v_{j_r}(\lambda,x) > 0\}\}.$$

The idea behind this is to choose the indices with the best evaluation of its corresponding subproblem TSKP$_i(\lambda)$. Therefore, the optimal solution to $L_\delta(\lambda)$ consists of the territories with center in $\{i_1, i_2, \ldots, i_{p^*}\}$ and the assignments of the BUs to these territories given by the solution of the p^* associated subproblems TSKP$_i(\lambda)$.

Thus, for a given vector of multipliers $\lambda \in \mathbb{R}_+^{|V|}$, an upper bound for MDCP$_\delta$ is computed by means of the procedure described above. As it is well known, the best Lagrangian bound is obtained by solving the Lagrangian dual problem:

(LD$_\delta$)

$$\overline{W}(\delta) = \min_{\lambda \in \mathbb{R}_+^{|V|}} L_\delta(\lambda),$$

which is solved by subgradient optimization.

3.3. Subgradient optimization algorithm

In this phase a classical subgradient optimization is performed [20,21]. Given an initial vector λ^0, a sequence $\{\lambda^k\}$ is generated by the rule:

$$\lambda_j^{k+1} = \max\{0, \lambda_j^k - \theta_k s_j^k\}, \quad j = 1, \ldots, n,$$

where s^k is a subgradient at $\lambda = \lambda^k$ and $\theta_k > 0$ is the step size, calculated through the commonly used formula:

$$\theta_k = \frac{\alpha_k(\eta^k - \eta_{lb})}{\| s^k \|^2},$$

with α_k being a scalar satisfying $0 < \alpha_k \leq 2$. In practice, this parameter is initialized to $\alpha_0 = 2$ and its value is halved if the upper bound fails to improve after a certain number of consecutive iterations; η^k is the upper bound at iteration k; η_{lb} is the lower bound available at iteration k usually obtained by applying a primal heuristic for MDCP$_\delta$.

The subgradient vector at iteration k is given by $s^k = [s_j^k]$, with

$$s_j^k = 1 - \sum_{i \in I_\delta(j)} x_{ij}^*, \quad j \in V,$$

where x_{ij}^* is the solution of the Lagrangian problem $L_\delta(\lambda)$.

In practice [15,16,33], multipliers vector $\lambda \in \mathbb{R}_+^{|V|}$ is commonly initialized with random values in the range $[0,10]$, while the stopping criteria are the following:

- $\theta \leq 0.00001$
- $\alpha \leq 0.00001$
- $\eta_k - \eta_{lb} < 1$
- If $\lfloor \eta_k \rfloor$ fails to improve after m consecutive iterations.
- Maximum number of iterations.

A summary of the subgradient procedure implemented is depicted in Algorithm 1.

Algorithm 1. Subgradient optimization procedure.

Input: P:=A TDP instance;
δ:=Covering radius;
T:=Stopping criteria;
t:=Number of iterations without improvement after which the parameter α is halved;
Output: $\overline{W}(\delta)$:=Best upper bound for MDCP$_\delta$;

```
η_lb ← −∞   η_k ← +∞ ;
λ_j^0 ← random[0, 10];   j ∈ V;
k ← 0 ;
count ← 0 ;
Terminate ← false;
while (not Terminate) do
    Solve L_δ(λ^k);
    if (L_δ(λ^k) < η_k) then
        η_k ← L_δ(λ^k);
    else
        count ← count + 1;
        if (count = t) then
            α ← α/2;
            count ← 0;
        end if
    end if
    Apply the primal heuristic to obtain a lower bound lb;
    if (lb > η_lb) then
        η_lb ← lb;
    end if
    s_j^k ← (1 − Σ_{i∈I_δ(j)} x_ij^k);   j ∈ V;
    θ_k ← α_k(η_k − η_lb) / ||s^k||^2;
    λ_j^{k+1} ← max{0, λ_j^k − θ_k s_j^k};   j ∈ V;
    k ← k + 1;
    if (Stopping criteria T is not satisfied) then
        Terminate ← true;
    end if
end while
W̄(δ) ← η_(k−1);
return W̄(δ);
```

3.3.1. Primal heuristic

Note that, given a vector of multipliers $\lambda \in \mathbb{R}_+^{|V|}$, the solution of $L_\delta(\lambda)$ may not be feasible for MDCP$_\delta$. Since single assignment constraints are relaxed, an $L_\delta(\lambda)$ solution may present multiple assignments of the BUs to the territories whereas there might be BUs that were not assigned to any territory. Therefore, at the inner iterations of subgradient optimization, primal bounds for MDCP$_\delta$ are heuristically built from $L_\delta(\lambda)$ by repairing infeasibility through the following steps:

1. This stage eliminates the multiple assignments of BUs (if they exist) by considering the unbalances (with respect to each activity measure) that produce the removal of BUs from the territories. Let $X^L = (X_{c(i_1)}, X_{c(i_2)} \ldots, X_{c(i_{p^*})})$ be the optimal solution to $L_\delta(\lambda)$, $\lambda \in \mathbb{R}_+^{|V|}$, where $X_{c(i)}$ represents the set of BUs that belong to

territory with center in $i \in V$ and let $I^L = (i_1, i_2, \ldots, i_{p^*})$ be the set of territory centers selected in the Lagrangian solution.
For each $j \in V$, the set I_j^L denotes the territory centers associated to basic unit j, i.e., $I_j^L = \{i \in I^L : x_{ij} = 1\}$. If $|I_j^L| > 1$, which means that basic unit j has been assigned to more than one territory, a function f_i is evaluated for each $i \in I_j^L$. This function quantifies the impact on the feasibility with respect to constraints (10), when the basic unit j is subtracted from the territory i and is calculated as follows:

$$f_i = \min_{a \in A}\{w^a(X_{c(i)}) - w_j^a - (1-\tau^a)\mu^a\}, \tag{17}$$

where $w^a(X_{c(i)}) = \sum_{j \in X_{c(i)}} w_j^a$ is the size of the territory $X_{c(i)}$ with respect to the activity $a \in A$, while w_j^a, τ^a and μ^a are parameters of TDP model described in Section 2. The territory that keeps the basic unit j is selected under the following criteria:

- If $\min_{i \in I_j^L}\{f_i\} \geq 0$, it means that each territory is feasible when BU $j \in V$ is eliminated from $X_{c(i)}$ and therefore, the territory with the lowest evaluation in function (17) is selected to keep the basic unit j, i.e., $i^* = \arg\min_{i \in I_j^L}\{f_i\}$.
- If $\min_{i \in I_j^L}\{f_i\} < 0$, it means that at least one territory becomes infeasible with respect to the minimum activity size $(1-\tau^a)\mu^a$ for some activity $a \in A$. Notice that when assigning basic unit j to a single territory center from I_j^L, those territories that do not satisfy balance constraints (10) are not considered in the primal solution of MDCP$_\delta$ since they become infeasible when j is removed from them. Then, it is convenient to select the territory that provides the greatest covered demand among those territories for which $f_i > 0$ to keep the basic unit j, i.e., $i^* = \arg\max_{i \in I_j^L}\{f_i : f_i < 0\}$.

2. Once that multiple assignment has been eliminated, we have a feasible solution for MDCP$_\delta$ by considering only those territories that meet balance constraints (10). It can be noticed that subtracting basic units from territories in the previous phase may lead to unbalanced territories with respect to some activity measure and therefore, such territories could not be included in a feasible solution for MDCP$_\delta$. Additionally, the Lagrangian problem solution may have unassigned BUs. Hence, a second phase that improves the actual feasible solution is applied as follows:

- Let U be the set of unassigned basic units in the Lagrangian solution. The idea at each iteration of this stage is to assign each $j \in U$ to that territory with the highest residual capacity among both activity measures. This is performed through these steps:
  - Territories in X^L are ranked by non-increasing order according to their residual capacity denoted as r_i, $i \in I^L$, which is calculated as follows:

    $$r_i = \max_{a \in A}\{b_i^{(\delta,a)} - w^a(X_{c(i)})\},$$

    being $\{X_{c(i_1)}, X_{c(i_2)}\ldots, X_{c(i_{p^*})}\}$ the ordered set in such a way that $r_{(i_1)} \geq r_{(i_2)} \geq \cdots \geq r_{(i_{p^*})}$.
  - Basic unit j is assigned to the territory in the ordered set with the lowest index $i^* \in I^L$ such that $d_{i^*j} < \delta$ and satisfies

    $$w_j^a \leq b_{i^*}^{(\delta,a)} - w^a(X_{c(i^*)}), \quad a \in A. \tag{18}$$

    Relationship (18) assures the compliance of constraints (11). If there is no territory with these characteristics, the basic unit j is not assigned.

At the end of the primal heuristic, we have a feasible solution and therefore, a primal bound for the MDCP$_\delta$, which may sometimes be feasible even for the TDP in the case that all BUs are assigned to exactly p territories which satisfy balance constraints (10). Algorithm 2 summarizes the primal heuristic.

Algorithm 2. Primal heuristic.

Input: P:=A TDP instance;
δ:=Covering radius;
$I^L = \{i_1, i_2, \ldots, i_{p^*}\}$:=Set of territory centers selected in the Lagrangian solution;
$X^L = \{X_{c(i_1)}, X_{c(i_2)}, \ldots X_{c(i_{p^*})}\}$:=Solution of Lagrangian problem $L_\delta(\lambda)$;
U:=Set of unassigned BUs in the solution of $L_\delta(\lambda)$;
Output: X^f: =Feasible solution (lower bound) for MDCP$_\delta$;
$X^f \leftarrow \phi$;
$X_{c(i)}$:=Territory with center in $i \in V$;
for all $j \in V$ **do**
  $I_j^L \leftarrow \{i \in V : x_{ij} = 1\}$;
  if $(|I_j^L| > 1)$ **then**
    for all $i \in I_j$ **do**
      $f_i \leftarrow \min_{a \in A}\{w^a(X_{c(i)}) - w_j^a - (1-\tau^a)\mu^a\}$;
    end for
    if $(\min_{i \in I_j^L}\{f_i\} > 0)$ **then**
      $i^* \leftarrow \arg\min_{i \in I_j^L}\{f_i\}$;
      for all $i \in I_j^L$ such that $i \neq i^*$ **do**
        $X_{c(i)} \leftarrow X_{c(i)} \backslash \{j\}$;
      end for
    else
      $i^* \leftarrow \arg\max_{i \in I_j^L}\{f_i : f_i < 0\}$;
      for all $i \in I_j^L$ such that $i \neq i^*$ **do**
        $X_{c(i)} \leftarrow X_{c(i)} \backslash \{j\}$;
      end for
    end if
  end if
end for
for all $j \in U$ **do**
  for all $i \in I^L$**do**
    $r_i \leftarrow \max_{a \in A}\{b_i^{(\delta,a)} - w^a(X_{c(i)})\}$;
  end for
  $i^* \leftarrow \arg\max_{i \in I^L}\{r_i : w_j^a \leq b_i^{(\delta,a)} - w^a(X_{c(i)}) \wedge d_{ij} \leq \delta, \ a \in A\}$;
  $X_{c(i^*)} \leftarrow X_{c(i^*)} \cup \{j\}$;
end for
for all $i \in I^L$ **do**
  if $(w^a(X_{c(i)}) \geq (1-\tau^a)\mu^a, \ a \in A)$ **then**
    $X^f \leftarrow X^f \cup X_{c(i)}$;
  end if
end for
return X^f;

3.4. The dual bounding scheme

In this section we present the bounding scheme for the TDP. The idea underlying this procedure is to carry out a search among the elements of the set $\overline{D}$ associated with the distance matrix in order to find the best lower (dual) bound on the optimal value of TDP. The procedure solves a series of Lagrangian duals $\overline{W}(d_k)$ and seeks for the maximum coverage radius d_{k^*} that satisfy the conditions of Proposition 3.1, thus obtaining the best dual bound from the covering radii candidates.

The proposed LB scheme is based on a binary search over the set $\overline{D}$. As a preprocessing step, this set $\overline{D}$ can be further reduced by the following test:

- *Elimination by lower bound*: If LB is a valid lower bound for TDP, then the set $\{d_0, d_1, \ldots, d_{k_l}\}$, where $k_l \in K$ is the largest index such that $d_{k_l} < LB$ can be discarded.
- *Elimination by upper bound*: If UB is a valid upper bound for TDP, then the set $\{d_{k_u}, d_{k_u+1}, \ldots, d_{k_{\max}}\}$, where $k_u \in K$ is the smallest index such that $d_{k_u} > UB$ can be discarded.

Algorithm 3 summarizes the dual bounding scheme for TDP.

Algorithm 3. Dual bounding scheme (DBS).

Input: P:=A TDP instance;
$\overline{D} = \{d_0, d_1, \ldots, d_{k_{\max}}\}$:=Ordered set of covering radii;
Output LB:=Lower (dual) bound on the optimal value of TDP;
$a \leftarrow 1$;
$b \leftarrow k_{\max}$;
while $(a < b)$ **do**
$k \leftarrow \lfloor \frac{(a+b)}{2} \rfloor$;
Solve LD_{d_k} and evaluate $\overline{W}(d_k)$;
if $(\overline{W}(d_k) < W_{\text{tot}})$ **then**
$a \leftarrow k+1$;
else
$b \leftarrow k-1$;
end if
end while
$\text{LB} \leftarrow d_a$;
return LB;

3.5. Pre-processing for DBS

In this section, a pre-processing phase which significantly reduces the computational effort of the binary search by obtaining both initial lower and upper bounds is developed. In addition to this, a relative tolerance ε for the size of the exploring interval is used.

To obtain an initial lower bound, a sequential search among the set $\overline{D}$ is performed which solves, at each iteration, the following relaxation of MDCP_δ:

$(\text{MDCP}_\delta - \text{R})$

$$\phi(\delta, x) = \text{Maximize} \quad f(x) = \sum_{i \in V} \sum_{j \in J_\delta(i)} w_j^1 x_{ij}, \tag{19}$$

$$\text{subject to} \quad \sum_{i \in V} x_{ii} \le p, \tag{20}$$

$$x_{ij} \in \{0, 1\}, \quad i \in V; j \in J_\delta(i). \tag{21}$$

Once again, it can be noticed that $\text{MDCP}_\delta - \text{R}$ is separable in the set V and it can be easily solved by calculating for each $i \in V$ the maximum demand $c_i(\delta)$ that can be covered from i within a radius δ as follows:

$$c_i(\delta) = \sum_{j \in J_\delta(i)} w_j^1.$$

Finally, to satisfy constraint (20), the indices in V are sorted by non-increasing order of the values $c_i(\delta)$ and the first p indices are chosen to calculate the amount of effective demand $C_{ef}(\delta)$ that can be covered by p territories within a maximum distance δ:

$$C_{ef}(\delta) = \sum_{r=0}^{p} c_{i_r}.$$

Therefore, the optimal value of $\text{MDCP}_\delta - \text{R}$ is given by $C_{ef}(\delta)$ which, at the same time, is an upper bound for MDCP_δ. Then, using Proposition 3.1 we determine if δ is a valid lower bound for the TDP. The purpose of the sequential search is therefore to find the best initial lower bound (i.e., the largest covering radius for which $C_{ef}(\delta) \le W_{\text{tot}}$). The procedure for solving $\text{MDCP}_\delta - \text{R}$ is outlined in Algorithm 4.

Algorithm 4. pre_processing $(P, \overline{D})$.

Input: P:=A TDP instance;
$\overline{D} = \{d_0, d_1, \ldots, d_{k_{\max}}\}$:=Ordered set of covering radii;
Output: k_1:=Index of the initial upper bound d_{k_1};
$t \leftarrow 0$;
$\delta \leftarrow d_t$;
$C_{ef}(\delta) \leftarrow 0$;
$c_i(\delta) \leftarrow 0; \quad \forall i \in V$;
while $(C_{ef}(\delta) \le W_{\text{tot}})$ **do**
for all $i \in V$ **do**
$c_i(\delta) \leftarrow \sum_{j \in J_\delta(i)} w_j^1$;
end for
Order the indices in V in such a way that $c_{i_1}(\delta) \ge \cdots \ge c_{i_{|V|}}(\delta)$;
$C_{ef}(\delta) \leftarrow \sum_{r=0}^{p} c_{i_r}(\delta)$;
$t \leftarrow t+1$;
$\delta \leftarrow d_t$;
end while
$k_1 \leftarrow t-1$;
return k_1;

A valid initial upper bound for TDP is obtained from a known heuristic developed by Ríos-Mercado and Fernández [36]. Algorithm 5 states the dual bounding scheme DBS_P.

Algorithm 5. DBS_P $(P, \overline{D})$.

Input: P:=A TDP instance;
$\overline{D} = \{d_0, d_1, \ldots, d_{k_{\max}}\}$:=Ordered set of covering radii;
Output LB:=Lower (dual) bound on the optimal value of TDP;
k_1:=**pre_processing**(); {Compute initial lower bound d_{k_1}}
k_2:=**R – GRASP**(); {Compute initial upper bound d_{k_2}}
$a \leftarrow k_1$;
$b \leftarrow k_2$;
while $\left(\frac{d_b - d_a}{d_a} \ge \varepsilon\right)$ **do**
$k \leftarrow \lfloor \frac{(a+b)}{2} \rfloor$;
Solve LD_{d_k} and evaluate $\overline{W}(d_k)$;
if $(\overline{W}(d_k) < W_{\text{tot}})$ **then**
$a \leftarrow k+1$;
else
$b \leftarrow k$;
end if
end while
$\text{LB} \leftarrow d_a$;
return LB;

4. Computational evaluation

In this section, we provide computational results for the dual bounding scheme we developed for the TDP. Our overall objective is to assess if DBS is a promising methodology for TDP. More specifically, the following issues are studied:

(1) The effect of the pre-processing stage (providing both dual and primal bounds).
(2) A comparison of the proposed bounding scheme with the LP relaxation.
(3) The assessment of quality of the DBS_P bounds when tried on medium size instances, for which optimal solutions are known.

All the procedures have been coded in C++ and compiled with the Sun C++ 8.0 compiler. The experimental work was carried out on a SunFire V440 computer under Solaris 9 operating system. CPLEX 11.2 callable libraries [25] were used to solve subproblems $TSKP_i$.

Randomly generated instances based on real-world data on planar graphs provided by the industrial partner were used. This data set is taken from [36]. In that work, full details on how the instances are generated can be found. Each instance topology was randomly generated as a planar graph in the [0, 500] × [0, 500] plane. The set $\overline{D}$ has then n^2 different distances within the range [0, 500]. A tolerance $\tau^a = 0.05, a \in A$, with respect to each activity measure was considered. The particular characteristics of the instances used are described in each experiment.

With regard to the subgradient procedure for solving $L_\delta(\lambda)$, the algorithmic rules that were considered are the following:

- Start with $\alpha = 2$ and halve its value if the dual bound fails to improve after 15 consecutive iterations.
- Stopping criteria:
  - Maximum iteration number (600 iterations).
  - If the current absolute value of the current difference between the upper and lower bounds is less than one unit (i.e., $ub - lb < 1$). As $MDCP_\delta$ is an integer programming problem, a difference less than one indicates that optimality has been achieved since the decision variables coefficients in the objective function are integer-valued. The optimal solution to the problem is given by the current lower bound.
  - If $\lambda_i = 0, \forall i \in V$. The optimal solution to LD_δ has been obtained, but a duality gap may exist. The best available solution is given by the current lower bound.
  - If $lb = W_{tot}$. The total assignment of BUs has been achieved by a primal solution of $MDCP_\delta$ then a feasible solution or upper bound for TDP has been found.
  - If $ub < W_{tot}$. Proposition 3.1 is met and a valid lower bound for TDP has been found.
  - If $\lfloor ub \rfloor$ fails to improve after 30 consecutive iterations.
  - If $\theta \leq 0.00001$. A duality gap exists and the best available solution is given by the current lower bound which is provided by the primal heuristic.

4.1. Comparing DBS and DBS_P

The improvement produced when a pre-processing is applied to DBS is first addressed. As stated in Section 3.5, initial upper and lower bounds are easily generated to reduce the initial set of coverage radii to be explored. In addition, the binary search procedure is executed until a relative gap ε (i.e., percentage difference) between the lowest and greatest values in the set of candidate radii is reached. In order to balance the trade-off between solution time and quality we set $\varepsilon = 0.001$ (i.e., 0.1%) in our computational study.

Three instance sets defined by $(n,p) \in \{(60,4),(100,6),(500,10)\}$ were generated. For each of these sets, 15 different instances were generated and tested using both binary search schemes. Table 1 compares DBS and DBS_P. The first column indicates the instance size tested. The second and third columns display the average CPU time required per instance under each scheme (time required for obtaining initial (lower and upper) bounds for TDP is also included). The fourth column shows the percentage reduction by DBS_P on the total execution time. Similarly, the last three columns show the information about the number of radii that were tested.

Results in Table 1 indicate that modified binary search DBS_P has a significant impact in the execution times, which are reduced up to 74.1%. It can be noticed that this improvement relies on the number of explored radii, which reaches a decrease of over 50% using pre-processing on tested instances. It can be concluded that providing initial upper and lower bounds as a preprocessing strategy pays off in terms of computational effort.

4.2. Evaluation of DBS_P bounds

This part of the work focuses on the study of the quality of the obtained bounds. As it was mentioned before, this dual bounding scheme is the first known to date for commercial territory design. For this reason, we make a comparison with bounds based on the LP relaxation. Additionally, the DBS_P bounds are compared with respect to optimal solutions for medium size instances (60 and 100 BUs instances).

4.2.1. Comparison with the LP relaxation

A comparison between DBS_P and the LP relaxation (LPR) lower bounds for TDP is carried out. A set of 30 instances of each size $(n,p) \in \{(500,10),(1000,20),(2000,20)\}$ was tested. There are several methods available through CPLEX for solving the LP relaxation. We made some preliminary testing and found that Sifting Algorithm [25] was the most efficient method in this case.

Results of the empirical comparison are summarized in Table 2 where the first column indicates the instance size, the second column displays the average relative deviation (RD) between the DBS_P and LPR bounds, and the third and fourth columns show the average running times for both LPR and DBS_P bounding schemes, respectively. This gap represents the relative improvement of the bound provided by the dual bounding scheme (lb(DBS_P)) with respect to the bound obtained by the linear

Table 1
Performance of DBS and DBS_P procedures.

Size (n,p)	Time (s)			Explored radii		
	DBS	DBS_P	Improvement (%)	DBS	DBS_P	Improvement (%)
(60, 4)	1306.39	513.67	60.7	12	6	50.0
(100, 6)	2812.29	694.75	63.3	12	6	50.0
(500, 10)	11811.97	3058.41	74.1	17	6	64.7

Table 2
Comparison of LPR and DBS_P bounding schemes.

Size (n,p)	RD (%)	Time (s)	
		LPR	DBS_P
(500, 10)	252.46	148.9	2352.4
(1000, 20)	259.03	1028.1	5719.8
(2000, 20)	346.16	6728.1	13 548.3

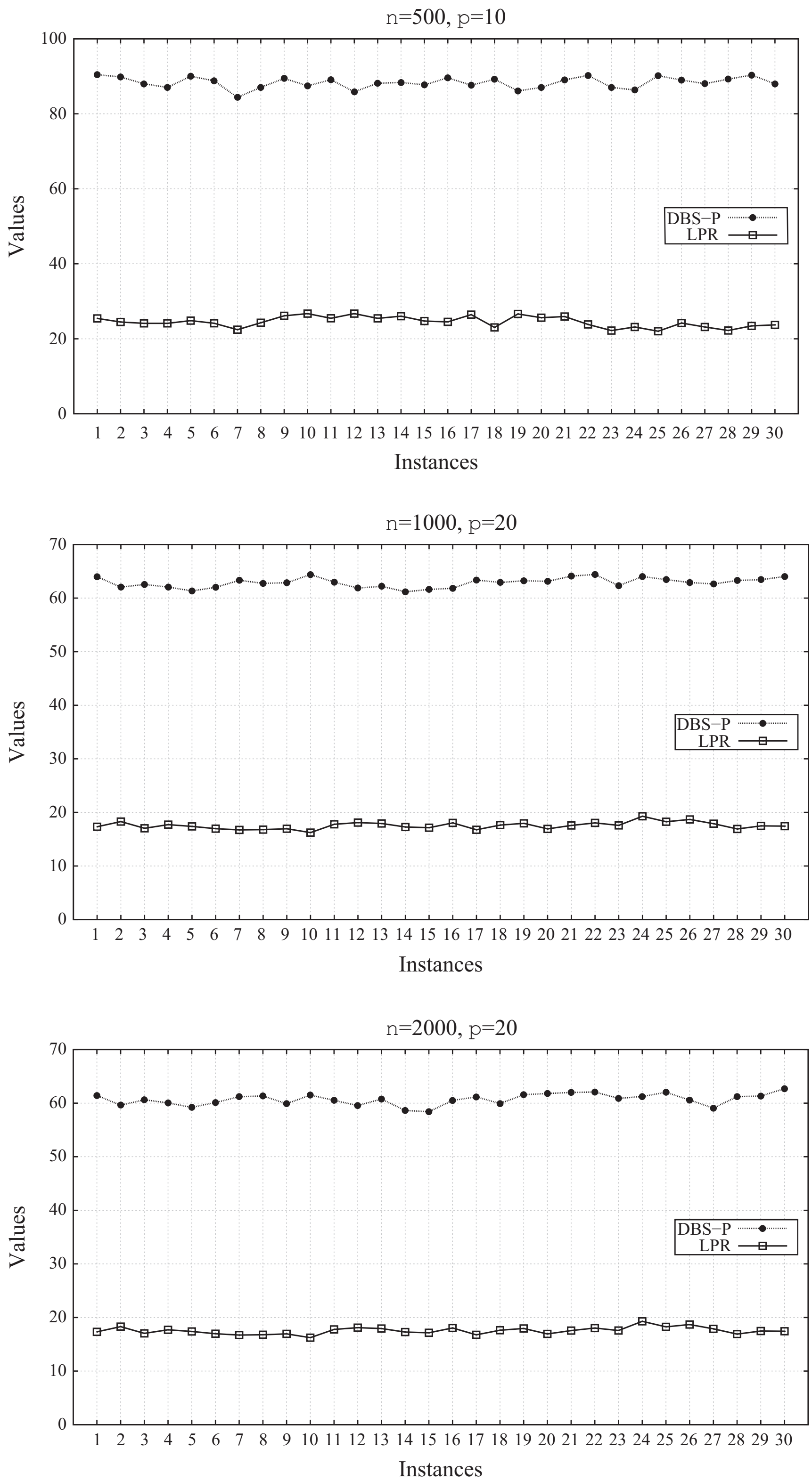

Fig. 1. Comparison of LPR and DBS_P lower bounds.

programming relaxation (lb(LPR)). It is computed as

$$\text{RD} = 100\left(\frac{lb(\text{DBS\_}P) - lb(\text{LPR})}{lb(\text{LPR})}\right).$$

As it can be observed, the average computation times of DBS_P are significantly larger than those reported by the resolution of the linear problem. However, the effort invested by DBS yields a significant improvement over the quality of the LPR bound. The average RD ranges from 252.46% to 346.16% which is remarkably high. This superiority in the quality of the bounds generated by LPR and DBS_P is better depicted in Fig. 1 where the values of both bounds per instance and size configuration are shown.

Table 3
Relative improvement of DBS_P with respect to ILPR.

Size (n, p)	RD (%)
(500, 10)	255.13
(1000, 20)	255.30
(2000, 20)	342.91

4.2.2. Comparison with an improved LP bound

From the previous experiment, it was clear that the better quality of the DBS_P bound came at a cost of a higher computational effort. Therefore, we investigate the improvement of the LPR bound when cast within a branch-and-bound (B&B) framework. As it is known, the B&B method iteratively improves its dual and primal bounds until optimality is reached. The main idea behind this experiment is to allow the B&B as much time as the

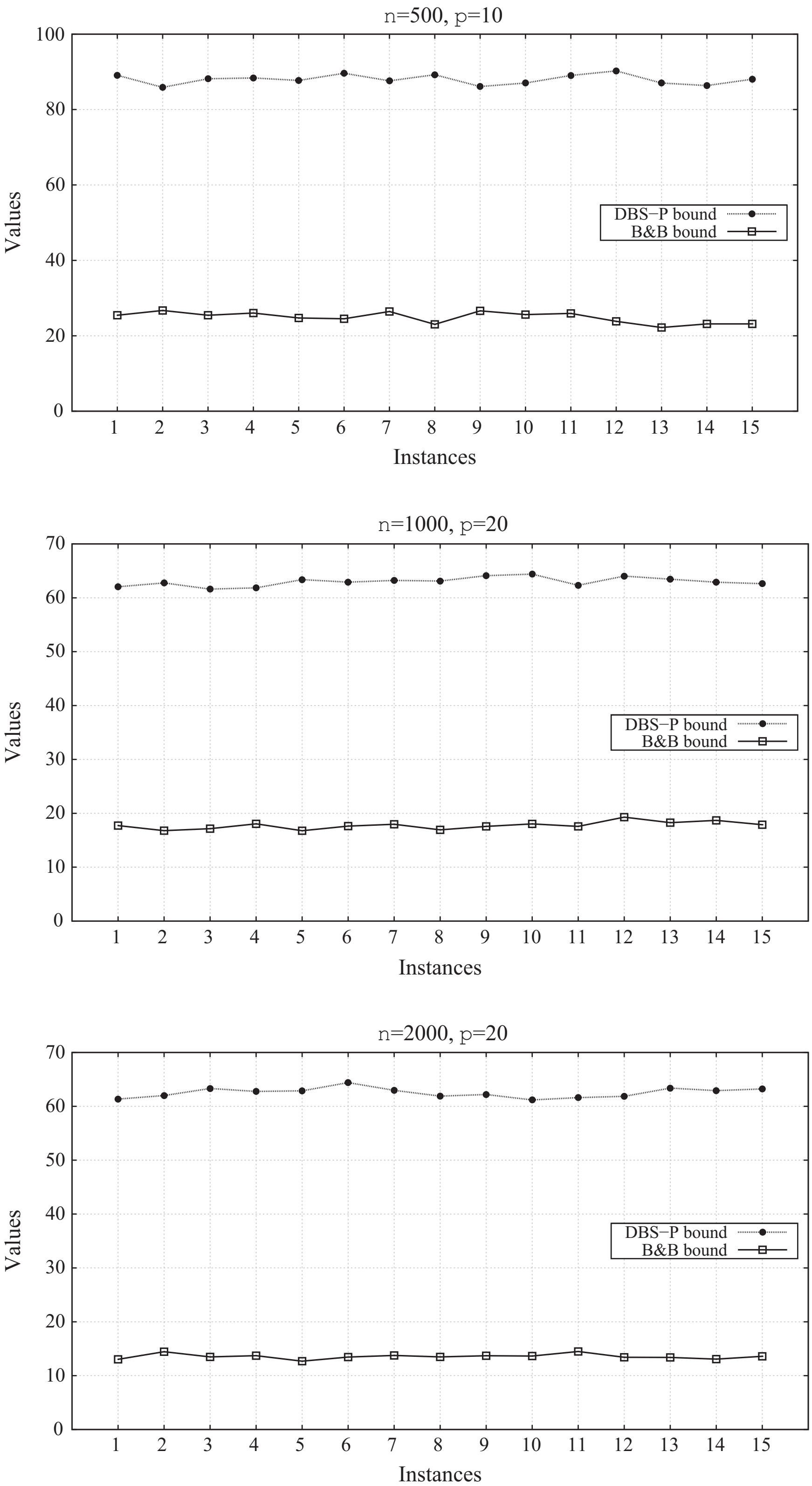

Fig. 2. Comparison of B&B and DBS_P bounds.

computation of the DBS_P bound task, and make a comparison of the DBS_P and the improved LPR bound (ILPR) under the same computational effort.

This experiment was carried out on 15 instances of each size configuration $(n,p) \in \{(500,10),(1000,20),(2000,20)\}$. Table 3 indicates the relative deviation (computed as in the previous test) between both ILPR and DBS_P bounds. The most important result in this experiment is that for all tested instances, the B&B method did not improve significantly the dual bound obtained at the root node, that is, the LPR bound. In other words, considering the same execution times for both strategies, the exact solution procedure failed to improve the linear relaxation while the proposed scheme is still better than the ILPR bound showing average relative deviations from 255.13% to 342.91%. Fig. 2 shows the individual bounds values, per instance and size configuration, for each bounding scheme.

Table 4
Comparison of DBS_P and LPR bounds vs. optimal solutions.

Size (n,p)	Gap (%)	
	DBS_P	LPR
(60,4)		
Best	0.10	59.94
Average	5.66	66.59
Worst	13.15	71.46
(100,6)		
Best	2.34	60.84
Average	10.50	67.62
Worst	16.58	72.83

4.2.3. *Comparison with optimal solutions*

Finally, the quality of the proposed DBS_P bound with respect to known optimal solutions is assessed. To this end, we solved 60- and 100-node instances by B&B implemented by CPLEX (20 instances on each set). This is the largest size that can be optimally solved in reasonable times.

Results are summarized in Table 4. For each bounding procedure a relative optimality gap is computed. This gap gives the relative deviation on how far is the lower bound (lb) from the optimal solution (opt) and is defined as $\text{Gap} = 100((opt - lb)/opt)$. As it can be seen from the table, the DBS_P scheme provides a more attractive choice than its LPR counterpart, confirming the results from previous experiments. In particular, it was observed that 90% of the 60-node instances had optimality gaps of less than 10% under the DBS_P scheme. Fig. 3 displays the LPR and DBS_P lower bound values as well as the optimal solution values of the different instances in each set (n,p).

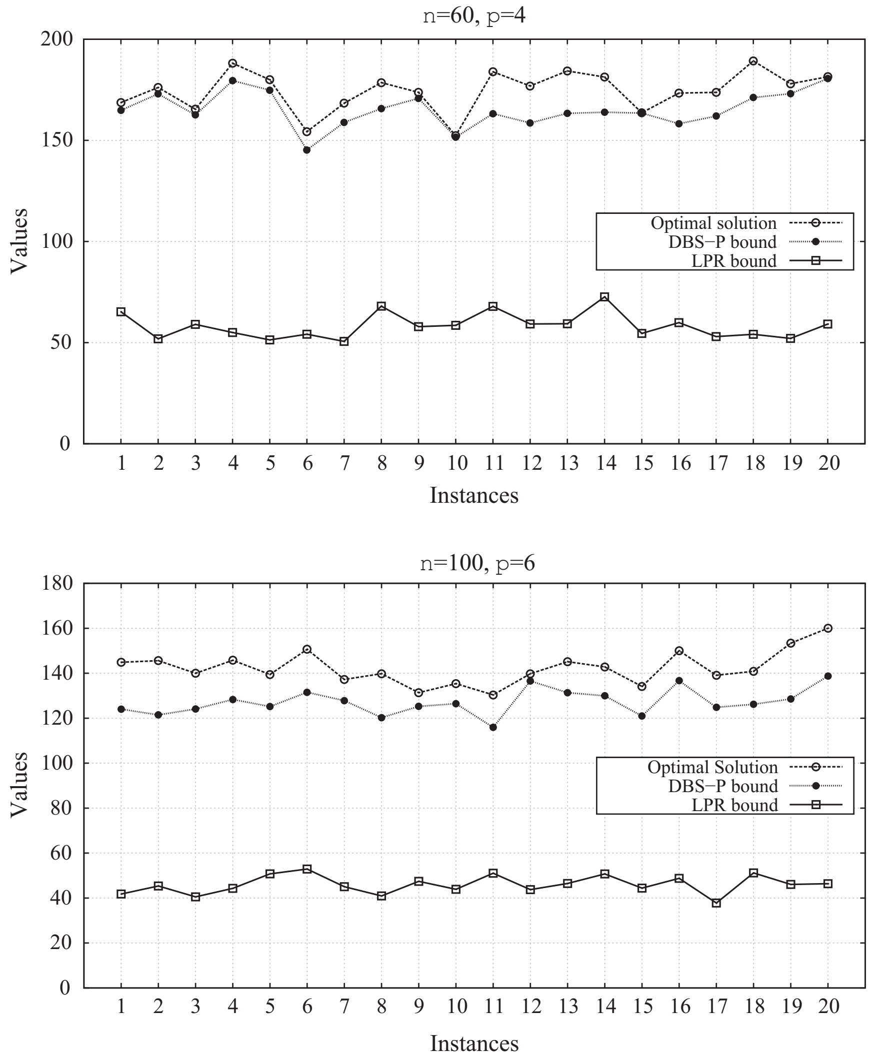

Fig. 3. Comparison of DBS_P bounds and optimal solutions.

5. Conclusions

In this paper we have presented a dual bounding scheme for a territory design problem. This problem includes compactness and balancing among territories as planning criteria. In particular, the problem addressed has been intractable through exact solution methods for real-world instance sizes, therefore different heuristic approaches have been proposed for this problem. However, to the best of our knowledge, there are no previous work on generating dual bounds for TDPs in general. As it is well known, the computation of dual bounds is important for assessing the quality of primal solutions, and moreover, dual bounds can be useful in the design of exact solution methods.

The proposed bounding procedure exploits the similarities of methodologies for solving the well known capacitated p-center problem. In this paper we extended the ideas underlying such methodologies and proposed an adaptation to handle multiple balance constraints. Lower bounds for TDP are obtained by performing a binary search on the elements on the matrix of distances between basic units. In each iteration of the procedure, the resolution of a Lagrangian dual from a coverage location problem is considered. This allows to evaluate, for a given coverage radius, if it is possible to assign all the BUs in a feasible way into p territories. When this is not met, the explored radius becomes a lower bound for the territory design problem.

In addition, a pre-processing technique to speed up the convergence of the procedure was developed. It was empirically observed that the positive impact of this simplification reducing up to 64.7% the number of explored radii during the binary search procedure yields a significant decrease in computation times. Furthermore, empirical evaluation showed that the proposed dual bound for TDP was of considerably higher quality than those provided by the linear programming relaxation of the model.

There are several extensions to this work that deserve attention. For instance, it was observed that the bottleneck in the overall execution time of the procedure is found at solving the TSKP subproblems derived from the Lagrangian relaxation of the maximum demand covering problem. Therefore, the derivation of efficient solution techniques for TSKP could greatly improve the efficiency of the proposed dual bounding scheme. To the best of our knowledge, this is a variation of the Knapsack Problem that has not been addressed before.

The study of other related location problems that can be used as auxiliary problems in the bounding scheme may also be worthwhile exploring as they could provide different dual bounds for TDP. For instance, the location minimum set covering problem ($LSCP_{\delta}$) seeks to minimize the number of territories that cover the total demand subject to assigning the BUs within a given radius δ.

A natural extension is to exploit the proposed bounding scheme for developing exact solution methods for TDP. Lagrangian heuristics form a wide family of methods that work well in finding efficient solutions for many integer programming problems. As the DBS_P procedure, these methods use a Lagrangian relaxation of the problem at hand to obtain easily solved subproblems and approximately solves the Lagrangian dual through an iterative optimization scheme. In this process, some Lagrangian (dual) information is used as an input to guide the construction of feasible solutions which are then submitted to local improvement. The Lagrangian heuristic is then embedded into a branch-and-bound scheme that yields further primal improvements. This B&B scheme can either be an exact method or a fast heuristic. Although our bounding scheme relaxes an auxiliary problem instead of the TDP, the DBS procedure can be extended to a Lagrangian heuristic framework to improve the primal solutions obtained during the subgradient optimization.

Acknowledgments

We are grateful to two anonymous referees, whose valuable comments and suggestions helped improve the presentation of this paper. This work was supported by the Mexican National Council for Science and Technology, Grants CONACYT CB-2005-01-48499Y and CB-2011-01-166397, and Universidad Auónoma de Nuevo León under its Scientific and Technological Research Support Program, Grants UANL-PAICYT CE012-09, IT511-10, and CE728-11.

References

[1] Albareda-Sambola M, Díaz JA, Fernández E. Lagrangean duals and exact solution to the capacitated p-center problem. Eur J Oper Res 2010;201(1):71–81.
[2] Bozkaya B, Erkut E, Haight D, Laporte G. Designing new electoral districts for the city of Edmonton. Interfaces 2011;41(6):534–47.
[3] Bozkaya B, Erkut E, Laporte G. A tabu search heuristic and adaptive memory procedure for political districting. Eur J Oper Res 2003;144(1):12–26.
[4] Caballero-Hernández SI, Ríos-Mercado RZ, López F, Schaeffer SE. Empirical evaluation of a metaheuristic for commercial territory design with joint assignment constraints. In: Fernandez JE, Noriega S, Mital A, Butt SE, Fredericks TK, editors. Proceedings of the 12th annual international conference on industrial engineering theory, applications, and practice IJIE, Cancun, Mexico, 2007. p. 422–7 (ISBN: 978-0-9654506-3-8).
[5] Church R, ReVelle C. The maximal covering location problem. Pap Reg Sci 1974;32(1):101–18.
[6] Daskin MS. Network and discrete location: models, algorithms and applications. New York: Wiley; 1995.
[7] Daskin MS. A new approach to solve the vertex p-center problem to optimality: algorithm an computational results. Commun Oper Res Soc Jpn 2000;45(9):428–36.
[8] Deb K, Agrawal S, Pratap A, Meyarivan T. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer M, Deb K, Rudolph G, Yao X, Lutton E, Merelo JJ, Schwefel HP, editors. Parallel problem solving from nature—PPSN VI. Lecture notes in computer science, vol. 1917. Berlin: Springer; 2000. p. 849–58.
[9] Drexl A, Haase K. Fast approximation methods for sales force deployment. Manag Sci 1999;45(10):1307–23.
[10] Duque JC, Ramos R, Suriach J. Supervised regionalization methods: a survey. Int Reg Sci Rev 2007;30(3):195–220.
[11] Elloumi S, Labbé M, Pochet Y. A new formulation and resolution method for the p-center problem. INFORMS J Comput 2004;16(1):84–94.
[12] Fisher ML. The Lagrangian relaxation method for solving integer programming problems. Manag Sci 1981;27(1):1–18.
[13] Fisher ML. An applications oriented guide to Lagrangian relaxation. Interfaces 1985;15(2):2–21.
[14] Fleischmann B, Paraschis JN. Solving a large scale districting problem: a case report. Comput Oper Res 1988;15(6):521–33.
[15] Galvão RD, ReVelle C. A Lagrangean heuristic for the maximal covering location problem. Eur J Oper Res 1996;88(1):114–23.
[16] Galvão RD, ReVelle C. A comparison of Lagrangean and surrogate relaxations for the maximal covering location problem. Eur J Oper Res 2000;124(2):377–89.
[17] Garfinkel RS, Nemhauser GL. Solving optimal political districting by implicit enumeration techniques. Manag Sci 1970;16(8):B485–508.
[18] Geoffrion AM. Lagrangean relaxation for integer programming. Math Program Stud 1974;2:82–114.
[19] Guignard M. Lagrangean relaxation. TOP 2003;11(2):151–228.
[20] Held M, Karp RM. The traveling salesman problem and minimum spanning trees. Oper Res 1970;18(6):1138–62.
[21] Held M, Karp RM. The traveling salesman problem and minimum spanning trees: Part II. Math Program 1971;1(1):6–25.
[22] Hess SW, Samuels SA. Experiences with a sales districting model: criteria and implementation. Manag Sci 1971;18(4):998–1006.
[23] Hess SW, Weaver JB, Siegfeldt HJ, Whelan JN, Zitlaur PA. Nonpartisan political redistricting by computer. Oper Res 1965;13(6):998–1006.
[24] Hojati M. Optimal political districting. Comput Oper Res 1996;23(12):1147–61.
[25] ILOG, Inc. ILOG CPLEX 11.0 user's manual. Mountain View, 2007.
[26] Kalcsics J, Nickel S, Schröder M. Toward a unified territorial design approach: applications, algorithms, and GIS integration. TOP 2005;13(1):1–56.
[27] Kariv O, Hakimi SL. An algorithmic approach to network location problems I: the p-centers. SIAM J Appl Math 1979;37(3):513–38.
[28] López-Pérez JF, Ríos-Mercado RZ. Embotelladoras ARCA uses operations research to improve territory design plans. Interfaces 2013;43(3):209–20.
[29] Megiddo N, Zemel E, Hakimi SL. The maximum coverage location problem. SIAM J Algebr Discrete Methods 1983;4(2):253–61.
[30] Mehrotra A, Johnson EL, Nemhauser GL. An optimization based heuristic for political districting. Manag Sci 1998;44(8):1100–13.
[31] Minieka E. The m-center problem. SIAM Rev 1970;12(1):138–9.

[32] Molina J, Laguna M, Martí R, Caballero R. SSPMO: a scatter tabu search procedure for non-linear multiobjective optimization. INFORMS J Comput 2007;19(1):91–100.
[33] Narciso MG, Lorena LAN. Relaxation heuristics for a generalized assignment problem. Eur J Oper Res 1996;91(3):600–10.
[34] Ozsoy FA, Pinar MÇ. An exact algorithm for the capacitated vertex p-center problem. Comput Oper Res 2006;33(5):1420–36.
[35] Ricca F, Simeone B. Local search algorithms for political districting. Eur J Oper Res 2008;189(3):1409–26.
[36] Ríos-Mercado RZ, Fernández E. A reactive GRASP for a commercial territory design problem with multiple balancing requirements. Comput Oper Res 2009;36(3):755–76.
[37] Ríos-Mercado RZ, López-Pérez JF. Commercial territory design planning with realignment and disjoint assignment requirements. Omega 2013;41(3):525–35.
[38] Ríos-Mercado RZ, Salazar-Acosta JC. A GRASP with strategic oscillation for a commercial territory design problem with a routing budget constraint. In: Batyrshin I, Sidorov G, editors. Advances in soft computing: Proceedings of the 10th Mexican international conference on artificial intelligence (MICAI 2011) Part II. Lecture notes in artificial intelligence, vol. 7095. Heidelberg, Germany: Springer; 2011. p. 307–18.
[39] Salazar-Aguilar MA, Ríos-Mercado RZ, Cabrera-Ríos M. New models for commercial territory design. Netw Spatial Econ 2011;11(3):487–507.
[40] Salazar-Aguilar MA, Ríos-Mercado RZ, González-Velarde JL. A bi-objective programming model for designing compact and balanced territories in commercial districting. Transp Res C Emerg Technol 2011;19(5):885–95.
[41] Salazar-Aguilar MA, Ríos-Mercado RZ, González-Velarde JL. GRASP strategies for a bi-objective commercial territory design problem. J Heurist 2013;19(2):179–200.
[42] Salazar-Aguilar MA, Ríos-Mercado RZ, González-Velarde JL, Molina J. Multi-objective scatter search for a commercial territory design problem. Ann Oper Res 2012;199(1):343–60.
[43] Shanker RJ, Turner RE, Zoltners AA. Sales territory design: an integrated approach. Manag Sci 1975;22(3):309–20.
[44] Toregas C, Swain R, ReVelle C, Bergman L. The location of emergency service facilities. Oper Res 1971;19(6):1363–73.
[45] Zoltners AA, Sinha P. Toward a unified territory alignment: a review and model. Manag Sci 1983;29(11):1237–56.
[46] Zoltners AA, Sinha P. Sales territory design: thirty years of modeling and implementation. Market Sci 2005;24(3):313–32.