J Heuristics (2013) 19:179–200
DOI 10.1007/s10732-011-9160-8

GRASP strategies for a bi-objective commercial territory design problem

M. Angélica Salazar-Aguilar · Roger Z. Ríos-Mercado · José Luis González-Velarde

Received: 11 March 2010 / Accepted: 7 February 2011 / Published online: 24 February 2011

Abstract A bi-objective commercial territory design problem motivated by a real-world application from the bottled beverage distribution industry is addressed. The problem considers territory compactness and balancing with respect to number of customers as optimization criteria. Previous work has focused on exact methods for small- to medium-scale instances. In this work, a GRASP framework is proposed for tackling considerably large instances. Within this framework two general schemes are developed. For each of these schemes two strategies are studied: (i) keeping connectivity as a hard constraint during construction and post-processing phases and, (ii) ignoring connectivity during the construction phase and adding this as another minimizing objective function during the post-processing phase. These strategies are empirically evaluated and compared to NSGA-II, one of the most successful evolutionary methods known in literature. Computational results show the superiority of the proposed strategies. In addition, one of the proposed GRASP strategies is successfully applied to a case study from industry.

Keywords Combinatorial optimization · Bi-objective programming · Territory design · GRASP

M.A. Salazar-Aguilar
CIRRELT, HEC Montréal, Université de Montréal, Pavillon André-Aisenstadt, Room 3485, Montréal H3T 2A7, Canada
e-mail: angelica.salazar@cirrelt.ca

R.Z. Ríos-Mercado (✉)
Graduate Program in Systems Engineering, Universidad Autónoma de Nuevo León, AP 111-F, Cd. Universitaria, San Nicolás de los Garza, NL 66450, Mexico
e-mail: roger@yalma.fime.uanl.mx

J.L. González-Velarde
Tecnológico de Monterrey, Center for Quality and Manufacturing, Monterrey, NL 64849, Mexico
e-mail: gonzalez.velarde@itesm.mx

1 Introduction

The problem addressed in this paper arises from a beverage distribution firm in Mexico. Single objective versions of the problem have been studied by Ríos-Mercado and Fernández (2009), Segura-Ramiro et al. (2007), Caballero-Hernández et al. (2007), and Salazar-Aguilar et al. (2011). The introduction of new bi-objective models and an exact solution procedure was proposed recently (Salazar-Aguilar et al. 2010). In general, commercial territory design belongs to the family of districting problems that have a broad range of applications like political districting (Bozkaya et al. 2003), school districting (Ferlang and Guénette 1990; Caro et al. 2004), and sales and service territory design (Kalcsics et al. 2005; Zoltners and Sinha 2005). In most of these applications a single objective problem is considered, however in the real world it is very common to pursue more than one criterion. In fact, looking at the literature on territory design, a few works address these problems as multiobjective problems (Tavares-Pereira et al. 2007; Ricca and Simeone 2008; Salazar-Aguilar et al. 2010). Territory design (TD) is very common in every enterprise dedicated to product sales and product distribution, specifically when the firm needs to divide the market into smaller regions to delegate responsibilities to facilitate the sales and distribution of goods. These decisions need to be constantly evaluated due to the frequent market changes such as the introduction of new products or changes in the workload, which are factors that affect the territory design. Additionally, the multiple planning requirements that the firm wants to satisfy and the large amount of customers that need to be grouped makes this difficult task even more critical. An efficient tool with capacity to provide good solutions to large problems is needed. Salazar-Aguilar et al. (2010) propose an exact solution procedure for the problem addressed in this work. In this procedure city blocks are used as basic units (BUs). They report efficient solutions for instances with up to 150 BUs and 6 territories. In the real world, it is very common to deal with instances of 500 to 2000 BUs. This fact motivates the efficient design and implementation of metaheuristic approaches for generating good solutions is short computing time. Therefore, in this work we propose some strategies based on a GRASP metaheuristic (BGRASP and TGRASP) aiming at finding a good approximation of the Pareto frontier. Each of these strategies consists of two main phases: construction and post-processing. In the construction phase a simultaneous territory creation is carried out and in the post-processing phase the neighborhood is explored in a similar way to that of the MOAMP procedure applied by Molina et al. (2007).

To the best of our knowledge no GRASP procedures have been implemented for solving multiobjective applications of the territory design problem, and moreover, no other metaheuristic technique for this particular commercial territory design problem has been developed for tackling large instances of the problem. However, GRASP procedures have been developed for other multiobjective combinatorial problems (Reynolds and de la Iglesia 2009; Higgins et al. 2008; Vianna and Arroyo 2004; Lourenço et al. 2001; Davoudpour and Ashraf 2009).

In our work, the effectiveness of GRASP as a multiobjective approach was analyzed over a set of instances based on real-world cases. The results reveal that our GRASP strategies outperform one of the best multiobjective evolutionary methods

(NSGA-II). In addition, BGRASP has better performance than TGRASP. The paper is organized as follows. In Sect. 2 the problem is described and a bi-objective optimization model is presented. In Sect. 3 the proposed solution strategies are fully detailed. Computational results, comparisons among different approaches, and a case study are discussed in Sect. 4. Finally we wrap up with the conclusions in Sect. 5.

2 Multiobjective commercial territory design

2.1 Problem description

In particular, the problem consists of finding a partition of the entire set of city blocks or basic units (BUs) into a fixed number (p) of territories, considering several planning territory requirements such as compactness, balance, and connectivity. Compactness means customers within a territory should be relatively close to each other. Balance implies territories must have similar size with respect to two attributes (number of customers and sales volume). Connectivity means BUs in the same territory can reach each other without leaving the territory. In addition, exclusive assignment from BUs to territories is required. The problem is modeled by an undirected graph $G = (V, E)$, where V is the set of nodes (BUs) and E is the set of edges representing adjacency between blocks. That is, a block or BU j is associated with a node, and an edge connecting nodes i and j exists if blocks i and j are adjacent. For each node $j \in V$ there are some associated parameters such as geographical coordinates (c_x, c_y), and two measurable attributes (number of customers and sales volume). The number of territories is given by parameter p. The company wants balanced territories with respect to each of the attribute measures. Let us define the size of territory V_k with respect to attribute a as: $w^{(a)}(V_k) = \sum_{i \in V_k}(w_i^{(a)})$, where $a \in \{1, 2\}$ and $w_i^{(a)}$ is the value associated to attribute a in node $i \in V$. Another characteristic is that all of the BUs assigned to each territory are connected by a path contained entirely within the territory. In addition, the BUs in each territory must be relatively close to each other (compactness). One way to achieve this requirement is to minimize a dispersion measure. We use a dispersion measure based on the objective of the p-median problem (p-MP). All parameters are assumed to be known with certainty. We used a bi-objective optimization model introduced by Salazar-Aguilar et al. (2010). In this model the compactness and the maximum deviation with respect to the number of customers are considered as objectives and the remaining requirements are treated as constraints. Let $N^i = \{j \in V : (i, j) \in E \vee (j, i) \in E\}$ be the set of adjacent nodes to node i; $i \in V$. The Euclidean distance between i and j is denoted by d_{ij}, $i, j \in V$.

Due to the discrete structure of the problem and to the exclusive assignment constraint, it is practically impossible to have perfectly balanced territories with respect to each attribute. Let $\tau^{(2)}$ be the specific tolerance allowed by the company to measure the relative deviation from average territory size with respect to sales volume. The average (target) value of attribute a can be computed as $\mu^{(a)} = w^{(a)}(V)/p$, $a \in A$.

2.2 Bi-objective optimization model

Decision variables

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

In that sense $x_{ii} = 1$ implies i is a territory center.

$$\text{Min} \quad f_1 = \sum_{j \in V} \sum_{i \in V} d_{ij} x_{ij}, \tag{1}$$

$$\text{Min} \quad f_2 = \max_{i \in V} \left\{ \frac{1}{\mu^{(1)}} \left| \sum_{j \in V} (w_j^{(1)} x_{ij}) - \mu^{(1)} x_{ii} \right| \right\}. \tag{2}$$

Subject to:

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

$$\sum_{i \in V} x_{ij} = 1 \quad \forall j \in V, \tag{4}$$

$$\sum_{j \in V} w_j^{(2)} x_{ij} \geq (1 - \tau^{(2)}) \mu^{(2)} x_{ii} \quad i \in V, \tag{5}$$

$$\sum_{j \in V} w_j^{(2)} x_{ij} \leq (1 + \tau^{(2)}) \mu^{(2)} x_{ii} \quad i \in V, \tag{6}$$

$$\sum_{j \in \cup_{v \in S} (N^i \setminus S)} x_{ij} - \sum_{j \in S} x_{ij} \geq 1 - |S| \quad i \in V;\ S \subset [V \setminus (N^i \cup \{i\})], \tag{7}$$

$$x_{ij} \in \{0, 1\} \quad i, j \in V. \tag{8}$$

Objective (Eq. 1) represents a dispersion measure based on a p-MP objective. In this sense, minimizing dispersion is equivalent to maximizing compactness. The second objective (Eq. 2) represents the maximum deviation with respect to the target size related to the number of customers. Thus, balanced territories should have small deviation with respect to the average number of customers. Constraint (Eq. 3) guarantees the creation of exactly p territories. Constraints (Eq. 4) guarantee that each node j is assigned to only one territory. Constraints (Eqs. 5–6) represent the territory balance with respect to the sales volume as it establishes that the size of each territory must lie within a range (measured by tolerance parameter $\tau^{(2)}$) around its average size. Constraints (Eq. 7) guarantee the connectivity of the territories. Note that, as usual, there is an exponential number of such constraints.

Single objective versions of this problem have been studied before. Ríos-Mercado and Fernández (2009) consider a version of the problem where the dispersion, modeled by a measure based on the p-center problem objective, is to be minimized. Their model considered as constraints three balancing requirements: balancing with respect number of customers, product demand, and workload. They develop a reactive

GRASP consisting of three phases: (i) A construction phase that builds q territories by using a greedy function that considers dispersion and violation with respect to the balancing constraints; (ii) an adjustment phase that converts the q-partition into a p-partition; and a local search phase that attempts to improve both infeasibility and the dispersion objective. Their empirical work shows very good results on instances of up to 500 and 1000 BUs. Caballero-Hernández et al. (2007) extend that model to account for joint assignment constraints, that is, requirements that a given set of pairs of BUs must belong to the same territory. They also proposed a GRASP for this problem with a preprocessing step that first satisfies the joint assignment constraints by solving an associated k-shortest path problem starting with n territories (one for each BUs). The construction phase of GRASP merges territories one by one until p territories are formed. The method is successfully applied to instances of 500 and 1000 nodes. In Segura-Ramiro et al. (2007), the authors consider a commercial TDP where the dispersion is now measured by a p-median problem type of function. This of course yields a model with different properties. The authors develop a location-allocation heuristic with relative good results. Salazar-Aguilar et al. (2011) address the single-objective problem from an exact optimization perspective. They present new integer quadratic programming models, and an exact optimization framework based on branch and bound and cut generation. They were able to solve instances of up to 200 BUs and 11 territories.

The bi-objective model was introduced by Salazar-Aguilar et al. (2010), 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, the one based on the p-center problem objective and the one based on the p-median objective model. It is shown how the latter has a tighter LP relaxation that allows to solve larger instances. The proposed method is shown to be successful for finding optimal Pareto frontiers on 60 to 150 node instances. It was also clear that larger instances were indeed intractable, making therefore the case for a heuristic approach. To the best of our knowledge this is the only multiobjective model for this particular application. In fact, most of the work on territory design and districting in general considers single-objective models. Among those few taking a multiobjective approach in territory design we can find the work of Guo et al. (2000) where a multiobjective zoning and aggregation tool (MOZART) is proposed. MOZART is an integration of a graph partitioning engine with a Geographic Information System (GIS) through a graphical user interface. They report a case with 577 census collection districts and 20 zones. Wei and Chai (2004) present a multiobjective hybrid metaheuristic approach for a GIS-based spatial zoning model. Their heuristic procedure is a combination of tabu search and scatter search. They show the procedure performance by solving a political districting problem with 55 basic units and 3 districts. Tavares-Pereira et al. (2007) study a multiobjective public service districting problem. They propose an evolutionary algorithm with local search and apply it to a real-world case of the Paris region public transportation. They discuss results for bi-objective cases considering different criteria combinations. Ricca and Simeone (2008) address a multiple criteria political districting problem. They transform the multiobjective model into a single-objective model using a convex combination and compare the behavior of four local search metaheuristics: descent, tabu search, simulated annealing, and old bachelor acceptance. The application is performed over a sample of five Italian regions where old bachelor acceptance produces the best results in most of the cases.

3 Proposed GRASP strategies

In general, GRASP is a metaheuristic that contains good features of both pure greedy algorithms and random construction procedures. It has been widely used for successfully solving many combinatorial optimization problems. GRASP is an iterative process in which each major iteration consists typically of two phases: construction and post-processing. The construction phase attempts to build a feasible solution and the post-processing phase attempts to improve it. The main motivation for GRASP stems from the fact that some of the good ideas developed for the single-objective versions of the problem can be exploited for this particular application, particularly in the construction and local search, where one can easily keep the connectivity constraints. As far as we are concerned no GRASP methods have been developed for this or similar multiobjective territory design problems. However, GRASP has been successfully applied to other multiobjective combinatorial optimization problems.

Lourenço et al. (2001) propose a GRASP for a bus-driver scheduling problem. However, GRASP is used as a single-objective optimizer, as a component of a multiobjective tabu search or genetic algorithm. Vianna and Arroyo (2004) present a simple GRASP for the multiobjective Knapsack problem. In each iteration of the GRASP, a linear combination of the objectives, referred to as a weighted linear utility function, is chosen. Then both the greedy randomized construction phase and the local search phase aim to maximize this weighted sum. Results presented suggest that the algorithm is competitive when compared with a number of multiobjective genetic algorithms. Higgins et al. (2008) present the problem of investing in landscapes to achieve outcomes that have multiple environmental benefits, formulating it as a multiobjective integer programming model, with objective functions representing biodiversity, water run-off and carbon sequestration. To find non-dominated solutions the authors develop a multiobjective GRASP embedded into an evolutionary method. Reynolds and de la Iglesia (2009) introduce a new multiobjective algorithm for the production and selection of classification rules for a particular class of a database (which is often referred to as partial classification), this algorithm is based on GRASP. The approach presented takes advantage of some specific characteristics of the data mining problem being solved, allowing for the very effective construction of a set of solutions that form the starting point for the local search phase of the GRASP. Davoudpour and Ashraf (2009) present a GRASP for a multiobjective flow shop scheduling problem with sequence-dependent setup times. The authors reduce the model to a single-objective model that adds the four criteria being considered, and address the problem basically as a single-objective model.

In this paper, we are introducing different GRASP schemes called BGRASP and TGRASP, each having two variants. For instance, BGRASP-I is a GRASP strategy that uses a merit function based on two components: dispersion and maximum deviation with respect to the target value in the number of customers. This method maintains connectivity as a hard constraint during the construction and post-processing phases. The BGRASP-I post-processing phase consists of optimizing three objective functions: dispersion, maximum deviation with respect to the number of customers, and total infeasibility in constraints (Eqs. 5 and 6). In contrast, BGRASP-II does not consider connectivity during the construction phase, its merit function is the same as used in BGRASP-I, but during post-processing phase, BGRASP-II adds connectivity as an objective function. Therefore, the goal in this post-processing phase is to minimize four objective functions: dispersion, maximum deviation, total infeasibility, and total number of unconnected BUs. TGRASP-I and TGRASP-II are described in a very similar way to BGRASP-I and BGRASP-II, respectively. The only difference is that the merit function in TGRASP-I and TGRASP-II has three components: dispersion, maximum deviation with respect to the number of customers, and maximum infeasibility with respect to constraints (Eq. 6). We described the GRASP strategies in a single scheme, see Procedure 1.

Procedure 1 shows the general scheme for the proposed GRASP strategies. An instance of the commercial territory design problem, the maximum number of iterations ($iter_{\max}$), the quality parameter (α), the minimum node degree (f) so that a node $i \in V$ can be selected as initial seed, the maximum number of allowed movements (max_{moves}) and the GRASP strategy (BGRASP-I, BGRASP-II, TGRASP-I or TGRASP-II) constitute the input. In order to explore the objective space in a better way, for each GRASP iteration a set of weights Λ is selected in such a way that $\lambda \in \Lambda : \lambda \in [0, 1]$. The two phases are applied for each $\lambda \in \Lambda$. Thus, for each iteration and each weight $\lambda \in \Lambda$ a construction phase and a local search phase is applied. The construction and the local search applied depends on the strategy chosen. Observe that, the merit function in BGRASP-I and BGRASP-II uses a weighted combination of the two original objectives. In contrast, in TGRASP-I and TGRASP-II the balancing constraints (Eqs. 5–6) are relaxed and added to the merit function.

Under strategies BGRASP-I and TGRASP-I, after the construction phase stops, the obtained solution may be infeasible with respect to the sales volume. Then, in order to obtain feasible solutions, during the post-processing phase infeasibility is treated as the objective to be minimized. In these strategies, this phase consists of systematically applying the local search sequentially to each of the three objectives individually. That is, first local search is applied using z_1 as the merit function in a single objective manner. After a local optimum is found, the local search is continued with z_2 as merit function, and then z_3. Finally, the initial objective z_1 is used after the local optimum is obtained for the last objective. During the search, the set of non-dominated solutions is updated at every solution. It is also clear that the order of this single objective local search strategy implies different search trajectories, that is, optimizing in the order (z_1, z_2, z_3) generates a trajectory different from (z_2, z_3, z_1), for instance. In BGRASP-II and TGRASP-II strategies, after the construction phase stops, the obtained solution may be infeasible not only with respect to sales volume balance, but with respect to the connectivity constraints as well. At the end of our GRASP strategies, an approximation of the Pareto front is reported.

3.1 BGRASP description

This strategy follows the generic scheme of GRASP. A greedy function (Eq. 9) during construction phase is a convex combination of two components weighted by λ which

Procedure 1 General framework for BGRASP and TGRASP strategies

Input:
α := GRASP RCL quality parameter
$iter_{\max}$:= GRASP iterations limit
f := Minimum node degree in the initial seeds
max_{moves} := Maximum number of movements in the post-processing phase
Obj := Number of objectives to be optimized during the post-processing phase
$strategy$:= BGRASP-I, BGRASP-II, TGRASP-I or TGRASP-II
Output: D^{eff}: set of efficient solutions

```
D^eff ← Ø
D^pot(S) ← Ø: set of potential efficients solutions
if (strategy ∈ {BGRASP-I,BGRASP-II}) then
  for (λ_1, λ_2, ..., λ_r) do
    for (l = 1, 2, ..., iter_max) do
      S ← ConstructSolutionBGRASP(α, f, λ, strategy)
    end for
  end for
else
  for (λ_1, λ_2, ..., λ_r) do
    for (l = 1, 2, ..., iter_max) do
      S ← ConstructSolutionTGRASP(α, f, λ, strategy)
    end for
  end for
end if
for (g = 1, ..., Obj) do
  D^pot(S) ←PostProcessing(S, max_moves, strategy, g, Obj)
  UpdateEfficientSolutions(D^eff, D^pot(S))
end for
return D^eff
```

are related to the original objectives: dispersion measure (Eq. 1) and maximum deviation with respect to the target number of customers (Eq. 2). The post-processing phase consists of the successive application of single-objective local search procedures (taking one objective at a time). These main BGRASP components are detailed as next.

3.1.1 BGRASP construction phase

In general, the construction phase consists of the assignment of BUs to territories keeping balanced territories with respect to the product demand while seeking good objective function values. Before the assignment process takes place p initial points are selected to open p territories. These points are the basis for the assignment process. Previous work showed that this method is very sensitive to the initial seed selection. For instance, when some seeds are relatively close to each other the growth of some territories stops way before reaching balancing. This implies some territories end up being relatively small. Hence, a better spread of the seeds is needed. In order to obtain better initial seeds, p disperse initial points with high connectivity degree are selected. Then, the construction phase starts by creating a subgraph $G' = (V', E(V'))$ where $i \in V'$ if and only if the degree of i, $d(i) \geq f$, where f is a user-given parameter. The seed selection is made by solving a p-dispersion problem (Erkut 1990) on G'. The p nodes are used as seeds for opening p territories. Let $\{i_1, i_2, \ldots, i_p\}$ be this set of disperse nodes. Then from this set, a partial solution $S = (V_1, V_2, \ldots, V_p)$ is starting by setting $V_t = \{i_t\}, t \in \{1, 2, \ldots, p\}$.

Then, at a given BGRASP construction iteration p partial territories are considered and the process attempts to allocate an unassigned node keeping balanced territories with respect to the demand. To do that, this method attempts to make assignments to the smallest territory (considering the demand). If BGRASP-I is the strategy selected by the user, the set of possible assignments is given only for those nodes that allow to preserve the connectivity. On the other hand, if the user selected BGRASP-II, the possible assignments are all those nodes that have not been assigned yet. Let V_{t^*} be the territory with smallest demand, $c(t^*)$ the center of V_{t^*} and $N(V_{t^*})$ the set of currently unassigned nodes that can be assigned to V_{t^*}. If $N(V_{t^*})$ is empty the procedure takes the next smallest territory and proceeds iteratively. The cost of assigning a node j to territory V_{t^*} is given by

$$\phi(j, t^*) = \lambda f_{\text{disp}}(j, t^*) + (1 - \lambda) f_{\text{dev}}(j, t^*), \tag{9}$$

where

$$f_{\text{disp}}(j, t^*) = \frac{1}{d_{\max}} \left(\sum_{i \in V_{t^*} \cup \{j\}} d_{ic(t^*)} \right), \tag{10}$$

$$f_{\text{dev}}(j, t^*) = \frac{1}{\mu^{(1)}} \max\left\{ w^{(1)} (V_{t^*} \cup \{j\}) - \mu^{(1)}, \mu^{(1)} - w^{(1)} (V_{t^*} \cup \{j\}) \right\}, \tag{11}$$

and the normalization parameter is

$$d_{\max} = \frac{(|V| - p)}{p} \max_{i,j \in V} \left\{ d_{ij} \right\}. \tag{12}$$

Observe that this greedy function is a weighted sum of the changes produced in the objective values.

Following the GRASP mechanism, a Restricted Candidate List (RCL) is built with the most attractive assignments which are determined by a quality parameter $\alpha \in [0, 1]$ (specified by the user). The RCL is computed as follows:

$$\phi_{\min} = \min_{j \in N(t^*)} \phi(j, t^*), \tag{13}$$

$$\phi_{\max} = \max_{j \in N(t^*)} \phi(j, t^*), \tag{14}$$

$$\text{RCL} = \{ j \in N(t^*) : \phi(j, t^*) \in [\phi_{\min}, \phi_{\min} + \alpha(\phi_{\max} - \phi_{\min})] \}. \tag{15}$$

Then, a node i is randomly chosen from the RCL. The territory V_{t^*} is updated, $V_{t^*} = V_{t^*} \cup \{i\}$ and the center $c(t^*)$ is recomputed. This is the adaptive part of GRASP.

This proceeds iteratively until all nodes are assigned. At the end of the process a p-partition $S = (V_1, V_2, \ldots, V_p)$ is obtained. This partition may be infeasible with respect to the balance of sales volume. In a few words, the proposed construction procedure tries to build territories similar in size with respect to the demand attribute. The next component of BGRASP is the post-processing or improvement phase.

3.1.2 BGRASP post-processing phase

The main idea of this local search is to successively apply a single-objective local search scheme (one objective function at a time) to avoid the cycling behavior observed in multiobjective search. This idea is motivated by its successful application in other MOCO methods (Molina et al. 2007). This process starts with the final solution obtained in the construction phase $S = \{V_1, \ldots, V_p\}$. Additionally, for each $V_t \in S$ a center $c(t) \in V_t$ is associated and a territory index $q(i) = t$ is known for $i \in V_t$. S may be infeasible with respect to the balancing constraints (Eqs. 5–6), thus in this phase BGRASP attempts to obtain feasible solutions by simultaneously searching for solutions that represent the best compromise between the objective functions. In order to obtain feasible solutions during this phase, balancing constraints (Eqs. 5–6) are dropped and are considered as an additional objective function instead. In the case of BGRASP-I, there are three objectives that are minimized:

(i) dispersion measure

$$z_1(S) = \sum_{j \in V_t, t \in T} d_{jc(t)}, \tag{16}$$

(ii) maximum deviation with respect to the number of customers

$$z_2(S) = \frac{1}{\mu^{(1)}} \max_{t \in T} \left\{ \max\{w^{(1)}(V_t) - \mu^{(1)}, \mu^{(1)} - w^{(1)}(V_t)\} \right\}, \tag{17}$$

(iii) total infeasibility

$$\begin{aligned} z_3(S) = \frac{1}{\mu^{(2)}} \sum_{t \in T} \max\bigl\{ & w^{(2)}(V_t) - (1 + \tau^{(2)})\mu^{(2)}, \\ & (1 - \tau^{(2)})\mu^{(2)} - w^{(2)}(V_t), 0 \bigr\} \end{aligned} \tag{18}$$

related to the balancing of sales volume (constraints (Eqs. 5 and 6)).

In contrast, the post-processing phase in BGRASP-II adds another minimizing objective to those three objectives used in BGRASP-I. It is given by

$$z_4(S) = \sum_{t \in T} |\eta(V_t)|, \tag{19}$$

where

$$\eta(V_t) = \bigcup_{r \in \{1, \ldots, q-1\}} B_r^t.$$

The function z_4 computes the total number of unconnected nodes. For territory $B^t = (V_t, E(V_t))$ let $B_r^t = (X_r, E(X_r))$ be the r-th connected component of B^t, for $r = 1, \dots, q$. For simplicity, let $c(t) \in X_q$. Evidently, if $q = 1$ then B^t is connected. Otherwise there are $q - 1$ sets of nodes that do not connect with the center $c(t)$ of territory V_t.

The post-processing phase attempts to find potential efficient solutions in the neighborhood of S. For doing that, a neighborhood $N(S)$ is defined. This neighborhood is formed by the solution set obtained by all possible moves such that a basic unit $i \in V_{q(i)}$ is reassigned to any adjacent territory $V_{q(j)}, q(j) \neq q(i)$, into the p-partition defined by S. When the current solution is connected, a move is allowed only if the resulting solution keeps the connectivity requirement. This means that, when BGRASP-I is used, only connected moves are allowed and when BGRASP-II is used, this condition is activated once a connected solution has been found. Each possible move *move*(i, j) deletes i from territory $q(i)$ and inserts it into territory $q(j)$, $(i, j) \in E, q(i) \neq q(j)$. For example, suppose there is a partition S with the structure $S = (\dots, V_{q(i)}, \dots, V_{q(j)}, \dots)$, if *move*$(i, j)$ is selected, the neighbor solution $\bar{S}$ is given by $\bar{S} = (\dots, V_{q(i)} \setminus \{i\}, \dots, V_{q(j)} \cup \{i\}, \dots)$. The *move*$(i, j)$ is accepted only if this improves the value of the objective function that is being optimized in that moment.

The neighborhood exploration consists of a relinked local search strategy. This is very similar to the local search proposed in MOAMP by Caballero et al. (2004) and used by Molina et al. (2007). The linking of single-objective local search schemes is made by considering different ordering of the objective functions being pursued. Suppose we select the optimization order as $(z_1(S), z_2(S), z_3(S))$, then the local search path is as follows: The first local search starts with any given solution S, typically obtained at the end of the construction phase, and attempts to find a better solution to the problem with respect to the single objective $z_1(S)$ (Eq. 16). Let S^1 be the best point visited at the end of this search. Then a local search is applied again to find the best solution to the problem with the single objective $z_2(S)$ (Eq. 17) using S^1 as initial solution. After that, a local search is applied to find the best solution to the problem considering the single objective $z_3(S)$ (Eq. 18) and the initial solution S^2 obtained in the previous optimization. At this point, we solve again the problem with the first objective $z_1(S)$ starting from S^3. This phase yields at least 3 points that approximate the best solutions to the single objective problems that result from ignoring all but one objective function. During this phase only feasible solutions are kept and a potential set of nondominated solutions is kept too. Additionally, efficient solutions may be found because all potential nondominated solutions are checked for inclusion in the efficient set E. This efficient set E is updated according to Pareto efficiency. This check is made over the original objectives: dispersion (Eq. 16) and maximum deviation with respect to the number of customers (Eq. 2).

Definition 1 Pareto efficiency. A solution $x^* \in X$ is efficient if there is no other solution $x \in X$ such that $f(x)$ is preferred to $f(x^*)$ according to Pareto order. That is, $x^* \in X$ is efficient if there is no solution $x \in X$ such that $f_i(x) \leq f_i(x^*)\ \forall i = 1, \dots, g$ and at least one $j \in \{1, \dots, g\}$ such that $f_j(x) < f_j(x^*)$.

Procedure 2 PostProcessing(S_0, g, Obj)

Input:
$S = S_0 :=$ Initial solution
$h = g :=$ objective index for starting the linked local search, $g \in \{1, 2, \ldots, Obj\}$
$Obj :=$ Number of objective functions to be optimized
Output: D: Nondominated solutions set

```
Do
  D ← ∅, count ← 0
  N(S): {Set of neighbors. In this case set of possible moves}
  A move (i, j) is represented by an arc (i, j) ∈ E such that t(i) ≠ t(j) i.e.,
  N(S) = {(i, j) ∈ E such that t(i) ≠ t(j) under the partition S}
  while (N(S) ≠ ∅) and (count < iter_max) do
    (i, j) ← select_move(N(S))
    N(S) ← N(S) \ {(i, j)}
    acceptable ← EvaluateMove(S, (i, j), h)
    if (acceptable) then
      S_t(i) ← S_t(i) \ {i}
      S_t(j) ← S_t(j) ∪ {i}
      count ← count+1
      Update(N(S))
      if (IsFeasible(S) = YES) then
        UpdateNDS(D, S)
      end if
    end if
  end while
  if (h < Obj) then
    h = h + 1
  else
    h = h − 1
  end if
While (h ≠ g)
return D
```

The relinked local search process can be repeated by using a different ordering of the objectives. In this work, different trajectories depending on the number of objectives to be optimized are explored. For instance, in BGRASP the following trajectories, starting from the same initial solution, were used: (z_1, z_2, z_3, z_1), (z_2, z_3, z_1, z_2) and (z_3, z_1, z_2, z_3). Each local search stops when the limit of iterations is reached or when the set of possible moves is empty. At the end the output is an approximate Pareto front.

3.2 TGRASP description

TGRASP-I and TGRASP-II are very similar to the BGRASP-I and BGRASP-II, respectively. The main difference is in the construction phase (see Procedure 3). During this phase the greedy function (Eq. 20) is a convex combination (Eq. 22) of

Procedure 3 ConstructSolutionTGRASP(α, f, λ, *strategy*)

Input:

α := GRASP RCL quality parameter

f := Minimum node degree which is required to consider a node as an initial seed to open a new territory

λ := weight used in the greedy function

strategy := TGRASP-I or TGRASP-II

Output: $S = (V_1, \ldots, V_p)$: Solution, p-partition of V

$T = \{1, \ldots, p\}, t \in T$:= Territory index

$c(t)$:= Center of V_t

$Flag(t)$:= 1 if a territory t is open, 0 otherwise

$B \leftarrow V$; $V_t \leftarrow \emptyset$

$H \leftarrow \{i \in V : |N^i| \geq f$ {*Subgraph of G used to select the initial seeds*}

Compute p disperse points $\{i_1, \ldots, i_p\}, i_t \in H$

for all $t \in T$ **do** $c(t) \leftarrow i_t$; $V_t \leftarrow V_t \cup \{i_t\}$; $B \leftarrow B \setminus \{i_t\}$

while $(B \neq \emptyset)$ **do**

$l \leftarrow \arg\min_{t \in T} \dfrac{w^{(2)}(V_t)}{\mu^{(2)}}$

if (*strategy* = TGRASP-I) **then**

$N(l) \leftarrow \bigcup_{i \in V_l} \{j \in N^i$ and $j \in B\}$ {*only connected nodes*}

else

$N(l) \leftarrow \bigcup \{j \in B\}$

end if

if $(N(l) \neq \emptyset)$ **then**

ComputeGreedyFunction $\gamma(j, c(l))$ **for all** $j \in N(l)$

$\gamma_{\min} \leftarrow \min_{j \in N(l)} \gamma(j, l)$

$\gamma_{\max} \leftarrow \max_{j \in N(l)} \gamma(j, l)$

RCL $\leftarrow \{j \in N(l) : \gamma(j, l) \in [\gamma_{\min}, \alpha(\gamma_{\max} - \gamma_{\min})]\}$

Random selection of $k \in$ RCL

$V_l \leftarrow V_l \cup \{k\}$

$B \leftarrow B \setminus \{k\}$

$c(l) \leftarrow \arg\min_{j \in V_l} \sum_{i \in V_l} d_{ij}$ {*Update center*}

else

flag$(t) \leftarrow 0$ {*Close territory*}

end if

end while

return $S = (V_1, \ldots, V_p)$

three components: dispersion measure (Eq. 10), maximum deviation with respect to the target number of customers (Eq. 11), and maximum infeasibility with respect to the upper bound of product demand balancing (Eq. 18). The strategy starts with p disperse points (obtained as in BGRASP construction phase) and the cost of assigning a node i to territory t with center $c(t)$ is measured by the greedy function

$$\gamma(j,t) = \lambda_1 f_{\mathrm{disp}}(j,t) + \lambda_2 f_{\mathrm{dev}}(j,t) + \lambda_3 f_{\mathrm{infeas}}(j,t), \tag{20}$$

where

$$f_{\mathrm{infeas}}(j,t) = \frac{1}{\mu^{(2)}} \max\left\{w^{(2)}(V_t \cup \{j\}) - (1+\tau^{(2)})\mu^{(2)}, 0\right\}, \tag{21}$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 1. \tag{22}$$

Note that (Eq. 21) penalizes for violations of the balancing constraint (Eq. 6) only. The post-processing phase is depicted in Procedure 2. Note that, in TGRASP-I and TGRASP-II, four objectives are minimized in the local search: (i) dispersion measure (Eq. 16), (ii) maximum deviation with respect to the number of customers (Eq. 17), (iii) infeasibility related to the balancing of product demand (Eq. 18), and (iv) total number of unconnected nodes (Eq. 19). The updating of efficient solutions is made by considering feasible solutions only.

4 Experimental results

4.1 Assessing the performance of the GRASP strategies

The proposed procedures were coded in C++, and compiled with the Sun C++ compiler workshop 8.0 under the Solaris 9 operating system and run on a SunFire V440. The test instances were taken from the library developed by Ríos-Mercado and Fernández (2009). These are based on real-world data provided by the industrial partner for each set. We test 20 large instances for each set with $(n,p) \in \{(1000,50),(500,20)\}$. Tolerance parameter was $\tau^{(2)} = 0.05$ and the input parameters for the GRASP strategies were $f = 2$, $\alpha = 0.04$, $\Lambda = \{0, 0.01, 0.02, \ldots, 1.0\}$, the total number of GRASP iterations was 2020 and 2000 was the maximum number of movements during the post-processing phase. These input parameters were set taking into account previous empirical work.

During our experimental work, we observed that the largest computational effort is during the post-processing phase. The multiple trajectories and the relinked local search on each trajectory increase the computational time dramatically. In order to find a good balance between construction and post-processing time, we made a filtering of solutions in order to apply the post-processing phase only over a set of the best solutions which were evaluated according to the merit function given by (Eq. 23). We tested other merit functions that empirically showed poor behavior. That motivated the use of this function for filtering solutions. Note that, each component is normalized.

$$\rho(S) = \frac{2 f_{\mathrm{disp}}(S)}{(|V|-p)d_{\mathrm{Max}}} + \frac{f^{(1)}_{\mathrm{Tdev}}}{p}, \tag{23}$$

where

$$f_{\mathrm{disp}}(S) = \sum_{t \in T} \sum_{j \in V_t} d_{jc(t)}, \tag{24}$$

$$f_{\mathrm{Tdev}}^{(1)} = \sum_{t \in T} \left\{ \frac{1}{\mu^{(1)}} \max\left\{ w^{(1)}(V_t) - \mu^{(1)}, \mu^{(1)} - w^{(1)}(V_t) \right\} \right\}, \tag{25}$$

and

$$d_{\mathrm{Max}} = \max_{i,j \in V} d_{ij}. \tag{26}$$

We selected 100 (out of 2020) solutions in such a way that these solutions have the smallest values in the merit function given by (Eq. 23). The post-processing phase (described in Procedure 2) was applied over the set of these filtered solutions.

This part of our experimental work was carried out to analyze the behavior of each proposed strategy. Figures 1 and 2 show examples of efficient frontiers obtained by all GRASP strategies. These results correspond to one instance on each size tested ((500, 20) and (1000, 50), respectively. In multiobjective optimization the performance comparison among different procedures is not an easy task. Observe for instance that TGRASP-II gives the best and the worst frontier in Fig. 1 and Fig. 2, respectively. Typically, the performance evaluation on multiobjective optimization is carried out using different metrics like the following:

1. *Number of points:* It is an important measure because efficient frontiers that provide more alternatives to the decision maker are preferred than those frontiers with few efficient points.
2. *k-distance:* This density-estimation technique used by Ziztler et al. (2001) in connection with the computational testing of SPEA2 is based on the k-th nearest neighbor method of Silverman (1986). This metric is simply the distance to the k-th nearest efficient point. We use $k = 4$ and calculate both the mean and the max of k-th nearest distance values. Thus, the smaller the k-distance the better in terms of the frontier density.
3. *Size of space covered (SSC):* Suggested by Ziztler and Thiele (1999), this measure computes the volume of the dominated points. Hence, larger values of SSC are preferred.
4. *C(A,B):* It is known as the coverage of two sets measure (Ziztler and Thiele 1999). This measure represents the proportion of points in the estimated efficient B that are dominated by the efficient points in the estimated frontier A.

We computed the average metric values over the tested instance sets. Tables 1 and 2 contain a summary of these results. It is important to comment that some strategies reported 4 or less efficient points for some (no more than 3) of the instances tested. These strategies are marked with (*). We do not considered these cases for computing the k-distance metrics.

Table 1 shows that TGRASP-II is an attractive strategy given that it reported the minimum values of the k-distance (mean and max) measure. That means the fronts reported by TGRASP-II are denser than those reported by the other strategies, in the set of instances (500, 20). Observe that the size of space covered (SSC) of TGRASP-II is very similar to the best value of this metric which was reported by BGRASP-II. In addition, the number of efficient points reported by TGRASP-II is very close to the best values of this metric (reported by TGRASP-I).

Table 2 shows that TGRASP-II reached the best k-distance values as in the previous set of (500, 20). However, the SSC values reported by TGRASP-II are the worst

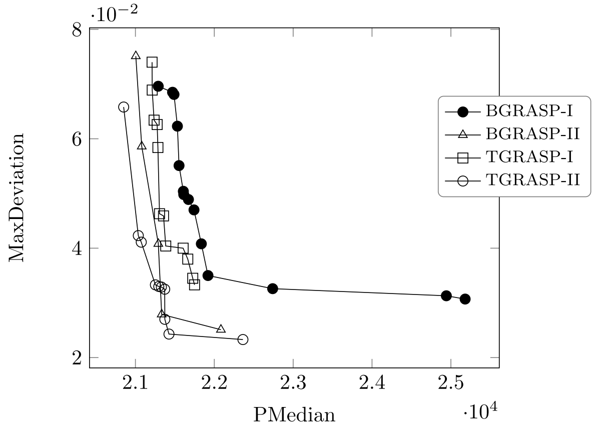

Fig. 1 Efficient frontiers for instance ds_500_20-03

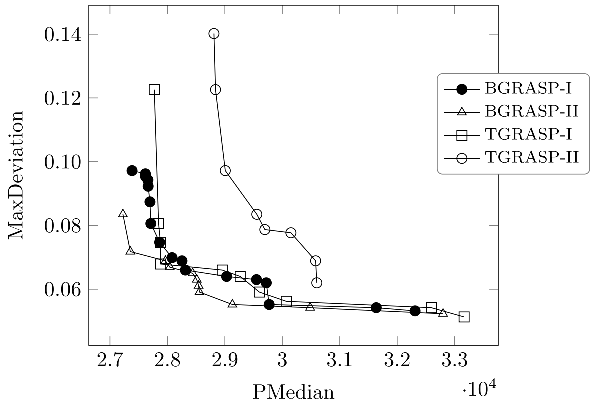

Fig. 2 Efficient frontiers for instance ds_1000_50-08

Table 1 Summary of metrics for instance set (500,20)

GRASP	*k*-distance (mean)			*k*-distance (max)			SSC			N. of points		
strategy	Min	Ave	Max	Min	Ave	Max	Min	Ave	Max	Min	Ave	Max
BGRASP-I	0.169	0.367	0.729	0.401	0.661	0.995	0.528	0.726	0.883	5.000	9.700	17.000
TGRASP-I	0.142	0.317	0.764	0.301	0.638	1.038	0.542	0.712	0.853	6.000	11.600	19.000
*BGRASP-II	0.145	0.339	0.851	0.203	0.618	0.996	0.682	0.851	0.993	4.000	8.950	16.000
TGRASP-II	0.117	0.294	0.570	0.156	0.569	0.900	0.638	0.867	0.971	5.000	9.250	16.000

for instances of (1000, 50) while BGRASP-II reported the best values of this metric. Regarding the number of efficient points both BGRASP-I and BGRASP-II had similar behavior.

A summary for the coverage of two sets measure is shown in Tables 3 and 4. Each column on these tables contains the mean proportion of points that are dominated by the strategy indicated by the row label. In Table 3, for instance, the values of the third row mean that the non-dominated points generated by BGRASP-II domi-

Table 2 Summary of metrics for instance set (1000,50)

GRASP	k-distance (mean)			k-distance (max)			SSC			N. of points		
strategy	Min	Ave	Max	Min	Ave	Max	Min	Ave	Max	Min	Ave	Max
BGRASP-I	0.173	0.326	0.926	0.372	0.659	1.093	0.542	0.786	0.905	5.000	11.850	18.000
*TGRASP-I	0.156	0.290	0.415	0.376	0.628	0.873	0.608	0.740	0.867	4.000	10.550	17.000
BGRASP-II	0.172	0.301	0.480	0.284	0.601	0.984	0.622	0.806	0.954	5.000	10.850	18.000
*TGRASP-II	0.086	0.291	0.494	0.398	0.547	0.809	0.100	0.306	0.550	3.000	9.300	25.000

Table 3 Mean value of coverage of two sets measure for instances from (500, 20)

Dominance	BGRASP-I	TGRASP-I	BGRASP-II	TGRASP-II
BGRASP-I	0.000	0.544	0.143	0.240
TGRASP-I	0.344	0.000	0.107	0.166
BGRASP-II	0.770	0.833	0.000	0.442
TGRASP-II	0.728	0.815	0.477	0.000

Table 4 Mean value of coverage of two sets measure for instances from (1000, 50)

Dominance	BGRASP-I	TGRASP-I	BGRASP-II	TGRASP-II
BGRASP-I	0.000	0.646	0.333	0.995
TGRASP-I	0.300	0.000	0.264	0.995
BGRASP-II	0.541	0.591	0.000	0.995
TGRASP-II	0.000	0.000	0.000	0.000

nate 77% of those non-dominated points obtained by BGRASP-I and 83.3% of those non-dominated points generated by TGRASP-I. In addition, Table 4 shows that for instances from (1000, 50) the non-dominated solutions obtained by BGRASP-II tend to dominate 99.5% of those non-dominated points generated by TGRASP-II. In all instances tested, BGRASP-II strategy presents the best compromise for this performance measure.

Taking into account the behavior of the proposed solution strategies, we conclude that BGRASP-II is the most robust strategy given that overall instances tested it reported the best mean values of the space size covered (SSC) and the coverage of two sets measures. Moreover, this procedure reported average values (close to the best values) for the number of points and k-distance metrics.

4.2 Comparison of GRASP strategies and NSGA-II

To asses the quality of our proposed GRASP strategies. We implemented one of the most successful multiobjective technique called NSGA-II (see Deb et al. 2002 for a detailed description). Four objective functions are minimized: (i) dispersion (Eq. 16), (ii) maximum deviation with respect to the average number of customers (Eq. 17), (iii) total infeasibility with respect to the balancing constraints of sales volume (Eq. 18), and iv) total number of unconnected nodes (Eq. 19). When the convergence criterion is reached, the best nondominated solutions are filtered to obtain

Table 5 Summary of mean metrics for two instances from set (500,20)

Procedure	N. of points	k-distance (mean)	k-distance (max)	SSC
BGRASP-I	6.000	0.368	0.636	0.738
TGRASP-I	15.000	0.187	0.779	0.760
BGRASP-II	7.500	0.219	0.507	0.795
TGRASP-II	9.000	0.214	0.293	0.855
NSGA-II	2.500	–	–	0.390

Table 6 Mean C(A,B) measure for two instances from (500, 20)

Dominance	BGRASP-I	TGRASP-I	BGRASP-II	TGRASP-II	NSGA-II
BGRASP-I	0.000	0.333	0.242	0.231	0.000
TGRASP-I	0.286	0.000	0.333	0.256	0.000
BGRASP-II	0.429	0.333	0.000	0.379	0.333
TGRASP-II	0.429	0.378	0.439	0.000	0.333
NSGA-II	0.095	0.089	0.030	0.179	0.000

those feasible solutions that are efficient with respect to the dispersion measure and the maximum deviation with respect to the average number of customers.

The same data sets described in the previous section were used for feeding the NSGA-II approach. This means that 20 instances of each size ($(500, 20)$ and $(1000, 50)$) were tested. Experimental results showed that NSGA-II significantly struggles when attempting to generate feasible solutions to the problem. Feasible solutions were found just for 2 out of 20 instances tested from (500, 20) and no feasible solutions were reported for those 20 instances from (1000, 50). This poor behavior is explained mainly by the presence of the connectivity constraints, which are very hard to satisfy under this scheme. Thus, this is one of these highly constrained combinatorial problems for which a method, such as the proposed GRASP, that better exploits the problem structure makes a tremendous difference.

An analysis of performance over those 2 instances solved by NSGA-II was carried out. Table 5 shows the metric average values for these instances. Observe that NSGA-II reported the lowest number of efficient points and the lowest value of SSC. The k-distance (mean and max) metric could not be computed given that we used a $k = 4$ and the NSGA-II did not report more than 3 efficient solutions for these instances.

The metric of coverage of two sets is showed in Table 6. In this table both BGRASP-I and TGRASP-I do not cover any of the efficient points generated by NSGA-II. In contrast, both BGRASP-II and TGRASP-II have a mean coverage of 33.3% over those efficient points reported by NSGA-II. And the highest coverage of NSGA-II is that one with 17.9% over those efficient points generated by TGRASP-II.

Figure 3 shows one of the instances that were solved by NSGA-II. The number of efficient points reported by NSGA-II are dramatically shorter than those reported by our proposed strategies (Table 5 shows this fact too). It is evident that the proposed

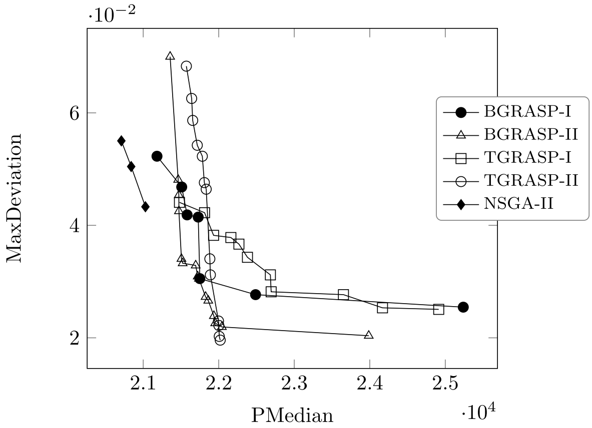

Fig. 3 GRASP strategies vs. NSGA-II, instance ds_500_20-04

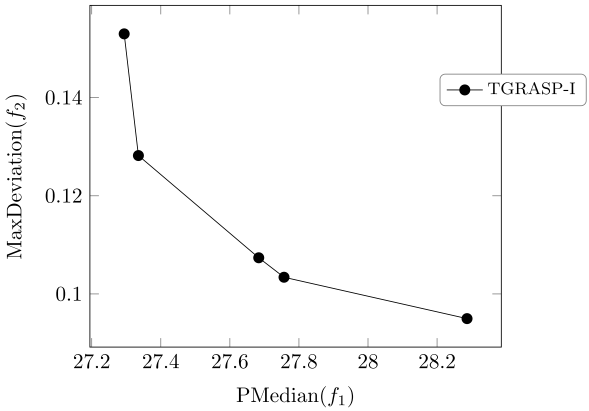

Fig. 4 Pareto front (TGRASP-I) for a real-world case with 1999 BUs and 50 territories

GRASP strategies clearly outperform the NSGA-II approach for this territory design problem.

4.3 Case study

A real-world instance with 1999 BUs and 50 territories was solved by applying one of our proposed procedure (TGRASP-I). The maximum number of moves in the post-processing phase was set to 3000, and the quality parameter in the RCL was $\alpha = 0.05$. During the fine-tuning this α value was better for this particular instance size. The rest of the input parameters are the same as in Sect. 4.1. Figure 4 shows the approximated front reported by TGRASP-I. It is important to mention that the firm solves the territory design problem around twice a year. Moreover, the planning department has tried to generate a territory design plan by minimizing a single-objective function, specifically the dispersion measure. For the instance tested in this work, they did not obtain feasible solutions even for the single-objective problem. In contrast, TGRASP-I reported approximate efficient solutions in less than 3 hours. Note that all of them are feasible solutions. Therefore, the proposed strategies are a very good alternative for the firm since these strategies are able to generate more than one attractive solution for the decision maker.

5 Conclusions

In this paper we have presented a GRASP approach for a bi-objective territory design problem with territory dispersion and customer balancing minimizing criteria. Two different variants (BGRASP and TGRASP) are implemented within the GRASP framework. For each variant, two different strategies for BGRASP and TGRASP are studied. Strategy I means that the connectivity requirement is holding as a hard constraint during all GRASP procedure. In contrast, strategy II means that the connectivity requirement is put away during the construction phase and incorporated as an objective function during the post-processing phase.

We carried out an evaluation of these GRASP strategies based on performance measures used in multiobjective optimization. These measures are: number of points, size of the space cover (SSC), k-distance, and coverage of two sets measure. The strategies were applied to two different instance sets of $(n, p) \in \{(500, 20), (1000, 50)\}$. For each of these sets, 20 instances based on real-world data provided by the industrial partner were used.

We observed that BGRASP-II presents the best mean performance over all multi-objective metrics. That means it is important to take control of the territories growth during the construction phase and diversify the search allowing unconnected territories. Therefore, these solutions have small infeasibility with respect to the sales volume at the beginning of the post-processing phase. Even thought these are unconnected, the relinked local search allows to reach feasibility for both balance and connectivity constraints in an efficient way.

The GRASP was compared with NSGA-II, a state-of-the-art evolutionary method for multiobjective combinatorial optimization problems. NSGA-II was tailored to this specific application. Empirical work show the proposed GRASP, under any strategy, significantly outperformed NSGA-II. In many cases, NSGA-II was unable to find feasible solutions to the problem, struggling particularly on satisfying the connectivity constraints. In contrast, the GRASP strategies proposed in this work reported feasible solutions for all instances tested.

In addition, a real-world instance provided by the industrial partner was solved successfully by using the TGRASP-I scheme. The company has not generated feasible solutions for this instance, even for the single-objective version of the problem. Our proposed strategy reported efficient (feasible) solutions for this case.

We have developed a very efficient method for addressing biobjective commercial territory design. A natural extension to this work is to incorporate in the model other criteria such as routing costs for instance. This would imply of course taking both design and routing decisions at the same level, and a greater challenge from the computational efficiency perspective, but several of the methods and concepts developed in this work can be used to this end.

Acknowledgements This work was supported by the Mexican National Council for Science and Technology under grants SEP-CONACYT 48499-Y and SEP-CONACYT 61903, Universidad Autónoma de Nuevo León under its Scientific and Technological Research Support Program grants UANL-PAICYT (CA1478-07 and CE012-09), and Tecnológico de Monterrey under research grant CAT128. The research of the first author has been funded by a doctoral fellowship from Universidad Autónoma de Nuevo León, grant NL-2006-C09-32652.

The authors wish to thank Professor Julián Molina from Universidad de Málaga, Spain, for his help on computing some performance measures. We are grateful to two anonymous reviewers whose criticism helped improve the quality of this work.

References

Bozkaya, B., Erkut, E., Laporte, G.: A tabu search heuristic and adaptive memory procedure for political districting. Eur. J. Oper. Res. **144**(1), 12–26 (2003)

Caballero, R., Gandibleux, X., Molina, J.: MOAMP: A generic multiobjective metaheuristic using an adaptive memory. Technical report, University of Valenciennes, Valenciennes, France (2004)

Caballero-Hernández, S.I., Ríos-Mercado, R.Z., López, F., Schaeffer, E.: Empirical evaluation of a metaheuristic for commercial territory design with joint assignment constraints. In: Fernandez, J.E., Noriega, S., Mital, A., Butt, S.E., Fredericks, T.K. (eds.) Proceedings of the 12th Annual International Conference on Industrial Engineering Theory, Applications, and Practice (IJIE), Cancun, Mexico, pp. 422–427 (2007). ISBN: 978-0-9654506-3-8

Caro, F., Shirabe, T., Guignard, M., Weintraub, A.: School redistricting: Embedding GIS tools with integer programming. J. Oper. Res. Soc. **55**(8), 836–849 (2004)

Davoudpour, H., Ashraf, M.: Solving multi-objective SDST flexible flow shop using GRASP algorithm. The Int. J. Adv. Manuf. Technol. **44**(7–8), 737–747 (2009)

Deb, K., Pratap, A., Agarwal, S., Meyerivan, T.: A fast elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)

Erkut, E.: The discrete p-dispersion problem. Eur. J. Oper. Res. **46**(1), 48–60 (1990)

Ferlang, J.A., Guénette, G.: Decision support system for the school districting problem. Oper. Res. **38**(1), 15–21 (1990)

Guo, J., Trinidad, G., Smith, N.: MOZART: A multi-objective zoning and aggregation tool. In: Proceedings of the Philippine Computing Science Congress (PCSC), pp. 197–201 (2000)

Higgins, A.J., Hajkowicz, S., Bui, E.: A multi-objective model for environmental investment decision making. Comput. Oper. Res. **35**(1), 253–266 (2008)

Kalcsics, J., Nickel, S., Schröder, M.: Towards a unified territorial design approach: Applications, algorithms, and GIS integration. Top **13**(1), 1–56 (2005)

Lourenço, H.R., Paixpo, J.P., Portuga, R.: Multiobjective metaheuristics for the bus-driver scheduling problem. Transp. Sci. **35**(3), 331–343 (2001)

Molina, J., Martí, R., Caballero, R.: SSPMO: A scatter tabu search procedure for non-linear multiobjective optimization. INFORMS J. Comput. **19**(1), 91–100 (2007)

Reynolds, A.P., de la Iglesia, B.: A multi-objective GRASP for partial classification. Soft Comput. **13**(3), 227–243 (2009)

Ricca, F., Simeone, B.: Local search algorithms for political districting. Eur. J. Oper. Res. **189**(3), 1409–1426 (2008)

Ríos-Mercado, R.Z., Fernández, E.A.: A reactive GRASP for a commercial territory design problem with multiple balancing requirements. Comput. Oper. Res. **36**(3), 755–776 (2009)

Salazar-Aguilar, MA, Ríos-Mercado, R.Z., Cabrera-Ríos, M.: New models for commercial territory design. Networks Spatial Econ. (2011) doi:10.1007/s11067-010-9151-6

Salazar-Aguilar, M.A., Ríos-Mercado, R.Z., González-Velarde, J.L.: A bi-objective programming model for designing compact and balanced territories in commercial districting. Transp. Res., Part C (2010). doi:10.1016/j.trc.2010.09.011

Segura-Ramiro, J.A., Ríos-Mercado, R.Z., Álvarez-Socarrás, A.M., de Alba Romenus, K.: A location-allocation heuristic for a territory design problem in a beverage distribution firm. In: Fernandez, J.E., Noriega, S., Mital, A., Butt, S.E., Fredericks, T.K. (eds.) Proceedings of the 12th Annual International Conference on Industrial Engineering Theory, Applications, and Practice (IJIE), Cancun, Mexico, pp. 428–434 (2007). ISBN: 978-0-9654506-3-8

Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Chapman and Hall, London (1986)
Tavares-Pereira, F., Figueira, J.R., Mousseau, V., Bernard, R.: Multiple criteria districting problems. The public transportation network pricing system of the Paris region. Ann. Oper. Res. **154**(1), 69–92 (2007)
Vianna, D.S., Arroyo, J.E.C.: A GRASP algorithm for the multiobjective knapsack problem. In: Proceedings of the XXIV International Conference of the Chilean Computer Science Society (SCCC'04), pp. 69–75. IEEE Computer Society, Arica (2004). ISBN: 0-7695-2200-9
Wei, B.C., Chai, W.Y.: A multiobjective hybrid metaheuristic approach for GIS-based spatial zone model. J. Math. Model. Algorithms **3**(3), 245–261 (2004)
Ziztler, E., Thiele, L.: Multiobjective evolutionary algorithms: A comparative case study and strenght Pareto approach. IEEE Trans. Evol. Comput. **3**(4), 257–271 (1999)
Ziztler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strenght Pareto evolutionary algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland (2001)
Zoltners, A.A., Sinha, P.: Sales territory design: Thirty years of modeling and implementation. Mark. Sci. **24**(3), 313–331 (2005)