Computers & Operations Research 78 (2017) 578–587

Contents lists available at ScienceDirect

Computers & Operations Research

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

Generic Pareto local search metaheuristic for optimization of targeted offers in a bi-objective direct marketing campaign

Vitor N. Coelho[a,b,c,*], Thays A. Oliveira[a,b,g], Igor M. Coelho[b,d], Bruno N. Coelho[b], Peter J. Fleming[e], Frederico G. Guimarães[f], Helena Ramalhinho[h], Marcone J.F. Souza[i], El-Ghazali Talbi[j,k], Thibaut Lust[l]

[a] *Grupo da Causa Humana, Ouro Preto, Brazil*
[b] *Instituto de Pesquisa e Desenvolvimento de Tecnologias, Ouro Preto, MG, Brazil*
[c] *Graduate Program in Electrical Engineering, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil*
[d] *Department of Computer Science, Universidade do Estado do Rio de Janeiro, RJ, Brazil*
[e] *Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, UK*
[f] *Department of Electrical Engineering, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil*
[g] *Department of Information and Communication Technologies, Universitat Pompeu Fabra, Barcelona, Spain*
[h] *Department of Economics and Business, Universitat Pompeu Fabra, Barcelona, Spain*
[i] *Department of Computer Science, Universidade Federal de Ouro Preto, Ouro Preto, MG, Brazil*
[j] *Cité Scientifique, University Lille 1, Lille, France*
[k] *INRIA Lille - Nord Europe Parc Scientifique de la Haute Borne 40, Avenue Halley, Bat A, France*
[l] *Computer Science Laboratory of Paris, Université Pierre et Marie Curie, Paris, France*

A R T I C L E I N F O

Keywords:
Direct marketing campaign
Sharpe ratio
Cross-selling
Metaheuristics
Multi-objective optimization
Pareto Local Search

A B S T R A C T

Cross-selling campaigns seek to offer the right products to the set of customers with the goal of maximizing expected profit, while, at the same time, respecting the purchasing constraints set by investors. In this context, a bi-objective version of this NP-Hard problem is approached in this paper, aiming at maximizing both the promotion campaign total profit and the risk-adjusted return, which is estimated with the reward-to-variability ratio known as Sharpe ratio. Given the combinatorial nature of the problem and the large volume of data, heuristic methods are the most common used techniques. A Greedy Randomized Neighborhood Structure is also designed, including the characteristics of a neighborhood exploration strategy together with a Greedy Randomized Constructive technique, which is embedded in a multi-objective local search metaheuristic. The latter combines the power of neighborhood exploration by using a Pareto Local Search with Variable Neighborhood Search. Sets of non-dominated solutions obtained by the proposed method are described and analyzed for a number of problem instances.

1. Introduction

In this paper, we consider a bi-objective metaheuristic for choosing sets of clients in direct-marketing campaigns. We call this problem the Targeted Offers Problem in Direct Marketing (TOPDM) promotional campaigns. Solving the proposed bi-objective problem involves searching for the sets of customers that maximize both the promotion campaign profit and the risk-adjusted return (reward-to-variability index). Candidate solutions should respect campaign operational requirements related to the investors' minimum desired profit, available budget, viability of the product offers and customer constraints.

Since there is uncertainty concerning whether a client will positively react to a new offer, a low-risk cross-selling campaign is sought. In the examples in this paper, we consider instances in which customers with high expected profits are the ones with higher volatility.

To handle the reward-to-variability concept, a risk-adjusted return measure based on the Sharpe Ratio [1] is proposed, which is a useful index for investment analysis usually adopted by investors facing choices under uncertainties [2].

In the complex scenario of acquiring new customers and improving existing relationships, the field of customer relationship management has been investigated for distinct applications [3,4]. As mentioned by Cohen [5], the right product should be offered to the right customer at the right time. The goal of recognizing and responding to client requirements remains a significant challenge. Among several techniques, Operational Research (OR) methods have been shown to be a

* Corresponding author at: Graduate Program in Electrical Engineering, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil.
E-mail address: vncoelho@gmail.com (V.N. Coelho).

http://dx.doi.org/10.1016/j.cor.2016.09.008
Received 28 February 2015; Received in revised form 4 July 2016; Accepted 12 September 2016
Available online 19 September 2016
0305-0548/

useful and powerful tool that is used by many marketing departments of well-established firms [6–8].

Bhaskar et al. [9] consider uncertainty in connection with the problem of selecting customers for a cross-selling campaign in a retail bank. A linear fuzzy model based on triangular fuzzy rules over the input parameters (client expected profit and positive responses rates) was developed. A real case involving up to 180,000 clients was employed. These customers were aggregated into a small number of groups according to their expected profits. Cohen [5] studied the case of a promotional campaign in an international bank. The term campaign implied one large pro-active customer contact campaign that comprised 11 distinct offers. Approximately 2.5 million potential customers were evaluated for that campaign. From response models, the specific benefit of each offer for each customer was estimated. In Nobibon et al. [10], a TOPDM was solved using heuristic algorithms and mathematical formulations; in particular, different mathematical programming models were developed. A Branch-and-Price [11] method was designed and the achieved upper bounds were used for comparing seven different optimization methods. The Branch-and-Price proved to be unable to obtain good feasible solutions for instances involving a large set of clients.

Among the proposed techniques, a Tabu Search algorithm [12] exhibited the best performance, using three different Neighborhood Structures (NS). Delanote et al. [13] included bundled products and the use of multi-channel structures, which allowed the offers to be made through different offering channels (such as mail, email, telemarketing, etc). In Oliveira et al. [14], the dataset of Nobibon et al. [10] was further explored and a new greedy randomized initial solution builder combined with General Variable Neighborhood Search (GVNS) [15] was proposed.

Here, we present a Multi-Objective approach based on the concepts of maximizing profits and searching, at the same time, for a set of customers with less variability over their expected return (represented by the Sharpe ratio). The use of different NS has been already investigated in the literature and applied for solving several $\mathcal{NP}$ – Hard problems [16–18]. In particular, Nobibon et al. [10] and Oliveira et al. [14] already showed the potential of trajectory search algorithms for the TOPDM. A new concept of Greedy Randomized Neighborhood Structures (GRNS) is also proposed in this paper. The idea of GRNS is to perform neighborhood exploration and reconstruct parts of the neighbor solution by means of a procedure inspired by the metaheuristic Greedy Randomized Adaptive Search Procedure (GRASP) [19]. In order to deal with multiple conflicting objectives, these mechanisms are included in a Two-phase Pareto Local Search with VNS (2PPLS-VNS) [20] to explore the search space in the quest for high quality sets of non-dominated solutions. A Generic 2PPLS-VNS is therefore designed, considering the core of the 2PPLS-VNS and the possibility of using different Neighborhood Exploration (NE) techniques [21]. The problem instances proposed by Nobibon et al. [10] are adapted and used as cases of study.

The main contributions of this current work are:

- use of profit variability measure in connection to the client response;
- consider the Sharpe ratio index for calculating risk-adjusted profit in targeted offers;
- introduction of a bi-objective direct marketing promotional campaign;
- design of a Greedy Randomized Neighborhood Structure;
- generalization of the 2PPLS in order to obtain non-dominated solutions with different neighborhood exploration techniques;

The remainder of this paper is organized as follows. Section 2 describes the TOPDM, as well as an introduction to the uncertainties concerning the client responses and the use of the reward-to-variability concept known as Sharpe ratio. Section 3 describes the proposed framework to engage the multi-objective TOPDM. Solution representation and its evaluation are described in Sections 3.1 and 3.2, respectively. The greedy randomized solution generator is described in Section 3.3. Section 3.4 presents the NS used to guide the search for non-dominated solutions, as well as the new GRNS concepts, which are described in Section 3.5. The combination of three multi-objective metaheuristics is described in Section 3.6. Section 4 presents the computational experiments, and, finally, followed by a summary and conclusions in Section 5.

2. Problem description

The variant of the TOPDM approached here is composed of a set of clients $C = \{c_1, c_2, \ldots, c_m\}$ and a set of possible product offers $O = \{o_1, o_2, \ldots, o_n\}$, quoted for the direct marketing campaign. A cost $c_{ij} > 0$ and profit $p_{ij} \geq 0$ is associated to each customer $i \in C$ if offer $j \in O$ is directed to him/her. For each client $i \in C$, there is a maximum number of offers M_i which would saturate that client. For each product offer $j \in O$, if the product j is selected to be used during the campaign, it should be offered to a minimal number of customers, O_j^{min}. Each product has an initial fixed cost f_j, if it is used in a campaign. A maximum available budget B_j is set by investors of the campaign, which means that the total cost associated with each product offer $j \in O$ should not be bigger than B_j. It also required a minimum rate of return HR of the whole campaign, known as the Hurdle Rate.

Nobibon et al. [10] considered a fixed probability of a client accepting an offer, which they called r_{ij}. This value was multiplied by the return to the firm, defined as DFV_{ij}, when client i responded positively to an offer of product j. Thus, the expected profit p_{ij} was estimated as $DFV_{ij} \cdot r_{ij}$. In contrast to their work, the bi-objective problem introduced in this paper considers a extra objective related to the variability over each client profit p_{ij}.

Client uncertainties, regarding their responding positively to an offer, usually increase when the expected profits from it are also high [1]. The latter is a well-known fact that investors used to face. In this sense, a variability parameter, defined as v_{ij}, is associated with each customer $i \in C$ when it is targeted by the offer $j \in O$. The higher the variability/volatility v_{ij}, the higher is the risk of investing in that client.

Fig. 1 shows a didactic example of a solution with three clients, m=3, and two products, n=2. It should be emphasized that all clients and product constraints are respected in this considered example. It can be seen that product offer o_1 is targeting the set of clients $\{c_1, c_2, c_3\}$, aggregating a total cost and profit of 7 and 10, respectively. Product offer o_2 is being offered to the set of clients $\{c_1, c_3\}$, consuming an amount of resources equal to 7 and with a total expected profit of 12.

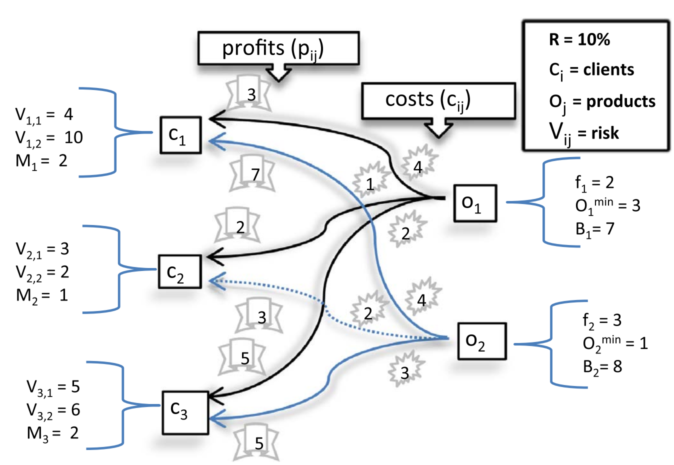

Fig. 1. TOPDM example.

$$s = \begin{array}{c|cc} & o_1 & o_2 \\ \hline c_1 & 1 & 1 \\ c_2 & 1 & 0 \\ c_3 & 1 & 1 \end{array}$$

Fig. 2. Solution representation example.

The total expected return and cost of this whole campaign are equal to 22 and 19, respectively. It means a total profit of 15.78% over the total amount of money invested over the campaign.

The total variability of the campaign is calculated by the weighted average of each individual volatility v_{ij} multiplied by the amount of profit p_{ij} expected from that client, as can be seen in Eq. (1).

$$V(s) = \frac{\sum_{i\in C}\sum_{j\in P} v_{ij}\cdot p_{ij}\cdot s_{ij}}{\sum_{i\in C}\sum_{j\in P} p_{ij}} \tag{1}$$

Obviously, only active offers are considered in the calculation, using the binary variable s_{ij} for this selection, as will be described ahead in Section 3.1.

Thus, the total risk V of the example given in Fig. 1 is:

$$V = \frac{143}{22} = 6.5\%$$

The TOPDM can be reduced to a special case of the 0–1 Multiple Knapsack Problem (MKP). A brief study of this class of problems can be found in [22]. In fact, this analogy can be verified if each possible offer $j \in O$ in the campaign is seen as a knapsack. Thus, in the beginning of the campaign, the knapsack is empty and should be filled with, at least, O_j^{min} offers. The knapsack maximum capacity is the available budget for the campaign of that product. Each client $i \in C$, seen as a single object to be inserted in this knapsack, has an expected profit p_{ij}, an analogy to each object benefit value in the classical Knapsack Problem. The weights of the items in the knapsack are related to the customers in the TOPDM and their respective costs c_{ij}. In this sense, since the MKP belongs to the $\mathcal{NP}$ – Hard class, the TOPDM also does.

In 1966, William Forsyth Sharpe (Economics Nobel prize in 1990) introduced the concept of reward-to-variability ratio as a way to measure portfolio performance. The generic Sharpe Ratio [1], as it was later called by academics and financial operators, can be seen as a deviation risk measure, or a risk-adjusted return, measuring the difference between the expected return of a fund and the benchmark investment, divided by the residual standard deviation, as presented in Eq. (2). It is often cited as a measure to calculate excess return (or risk premium).

$$SR = \frac{E[RI - RI_f]}{\sigma_R - \sigma_{R_f}} \tag{2}$$

In Eq. (2), $E[\cdot]$ represents the mathematical expectation operator, RI is the return of an asset and RI_f is the risk-free return. In the case that RI_f is a constant risk-free return throughout the analyzed period (as in the current case of study, the marketing campaign), $\sigma_{RI} - \sigma_{RI_f} = \sigma_R$. $E[RI - RI_f]$ is the expected value of the excess of the asset return over the benchmark return.

The Sharpe Ratio has been useful when individual investors face a choice under uncertainty, as it happens in the case of the expected profit from huge direct marketing campaigns. It is able to evaluate not only the profitability, but also the risk faced by the investors of the cross-selling campaign.

3. Methodology

3.1. Solution representation

A solution handled by the metaheuristic algorithm is represented by a binary matrix $R_{|C|\times|O|}$, where C indicates the set of available customers and O represents the possible products to be used in the campaign. If a given cell $s_{i,j}$, with $i \in C, j \in O$, is equal to "1" (true), the product j will be offered to the client i; otherwise, the value would be "0" (false).

Fig. 2 shows the solution representation for the example previously given in Fig. 1.

3.2. Objective functions and evaluation

A solution s is evaluated regarding two conflicting objective functions, that should be maximized:

- $f^P(s)$, Eq. (3), which measures the total profit of the campaign;
- $f^{SR}(s)$, Eq. (5), which measures the total reward-to-variability of the whole investment, inspired from the Sharpe Ratio SR index;

$f^P(s)$ is seen as the total return of the marketing campaign and is calculated following Eq. (3). Basically, it is the total expected profits p_{ij}, given the set of active clients i targeted by offers j, minus the total costs TC, described in Eq. (4).

The adaptation of the generic Sharpe Ratio for the TOPDM can be seen in Eq. (5). As mentioned in Section 2, the classic indicator has a variable related to the risk free investment. Here, the Hurdle Rate HR of the campaign, settled by the investors, is used instead of the risk-free R_f. Thus, we define $HR = R_f$, which is constant during the whole campaign, since investors will not change their minimum total profit. Finally, the original σ_R is now seen as the total risk of the campaign, summed into variable $V(s)$, already described in Eq. (1).

$$f^P(s) = \sum_{i\in C}\sum_{j\in P} p_{ij}s_{ij} - TC(s) \tag{3}$$

$$TC(s) = \sum_{i\in C}\sum_{j\in P} c_{ij}s_{ij} - \sum_{j\in P} f_j y_j \tag{4}$$

$$f^{SR}(s) = \frac{\frac{f^P(s)}{TC(s)} - HR}{V(s)} \tag{5}$$

3.3. Building an initial solution

A novel greedy randomized solution generator, adapted from the greedy randomized procedure proposed by Oliveira et al. [14], is described in this section. The procedure is inspired by the pure greedy procedure of Van Praag [23].

The pseudo-code of this procedure is described in Algorithm 1. Its main parameters are described below:

- Set A contains available customers which are not saturated and can receive more offers;
- set S is the set of products already selected to the campaign;
- variable tp measures campaign current total profit;
- variable tc indicates current total cost;
- set LC_j is a list of possible clients to be targeted by the product offers j , greedily ordered;
- set LRC_j is a restricted list of possible clients for a product offer j, ordered according to the greedy randomized parameter $\gamma \in [0, 1]$;
- set CS_j is the current set of selected clients of j;
- variables C_j, P_j and PR_j are the current costs, profits and campaign profit, respectively, of the list of clients in CS_j.

Algorithm 1. BuildGRASPSolution.

Input: $\gamma \in [0, 1]$
Output: initial solution s
1 $tp \leftarrow 0$, $tc \leftarrow 0$, $A \leftarrow C$ and $S \leftarrow \emptyset$.
2 **foreach** $j \in P$ **do**
3   **foreach** $i \in M$ **do**
4     $ANPP_{ij} \leftarrow \frac{(p_{ij}v_{ij} - c_{ij})}{c_{ij}}$
5   **end**
6 **end**
7 **forall** $j \notin S$ **do**
8   $LC_j \leftarrow$ set of clients $i \in A$ sorted in decreasing order by its $ANPP_{ij}$
9   $CS_j \leftarrow \emptyset$, $C_j \leftarrow 0$ and $P_j \leftarrow 0$
10   **for** $|CS_j| \le O_j$ **do**
11     $LRC_j \leftarrow \emptyset$
12   **for** $c = 0 \le \gamma \cdot |LC_j|$ **do**
13       $LRC_j \leftarrow LRC_j \cup \{LC_j(c)\}$
14   **end**
15   Select a client $i \in LRC_j$ at random
16     $CS_j \leftarrow CS_j \cup \{i\}$
17     $LC_j \leftarrow LC_j \setminus \{i\}$
18   **end**
19   $C_j \leftarrow$ sum of all the costs c_{ij} from clients $i \in CS_j$
20   $P_j \leftarrow$ sum of all benefits p_{ij} from clients $i \in CS_j$
21   $PR_j \leftarrow P_j - C_j - f_j$
22 **end**
23 Select j^* with the highest $PR_{j^*} > 0$ such that $[C_{j^*} \le B_{j^*}]$ & $[\ (tp + P_{j^*}) \ge (1 + HR)(tc + C_{j^*} + f_{j^*})]$
24 **if** $\exists j^*$ **then**
25   $S \leftarrow S \cup \{j^*\}$
26   $tp \leftarrow tp + P_{j^*}$ and $tc \leftarrow tc + C_{j^*} + f_{j^*}$
27   Update targeted offers of product j^* in solution s
28   Update A by removing all saturated clients
29   Back to line 7
30 **end**
31 **foreach** *active client* $i \in A$ **do**
32   **foreach** $j \in S$ **do**
33   **if** $p_{ij} > c_{ij}$ *and the offer* j *directed to the client* i *generates a feasible solution* **then**
34       $s_{ij} \leftarrow 1$
35   **end**
36   **end**
37 **end**
38 **return** s

Line 1 of Algorithm 1 initializes auxiliary variables and sets. Variables tp and tc are used in line 23 for verifying if the minimum hurdle rate HR is satisfied. From lines 2 to 6, a metric called ANPP measures clients profitability and variability, which is an adaptation of the one presented in Nobibon et al. [10]. It measures the amount of reward related to the costs, that is being invested, multiplied by the risk of that client.

Between lines 7 and 22, the "best" set of clients with respect to the greedy parameters γ is defined. In line 8 the clients are sorted in decreasing order of their *ANPP* values. This means that clients with higher profits are the most desired ones, but, a balance with its volatility is also done. From lines 10 to 18 the minimum number of clients O_j for the product offer j is selected. The restricted number of candidates $\gamma \cdot |LC_j|$ is added to the set LRC_j.

The best product j^* with the highest PR_j values, satisfying all operational constraints, is selected in the line 23. Clients targeted to receive offer j^* are inserted in solution s and the algorithm returns to line 7. If there are no more available offers nor the minimum number of available clients, the procedure goes to its final steps from lines 31 to 37, when the solution s is refined and improved. At this step, all available clients are sought to receive more offers, if it generates feasible solutions and improves the campaign total profit.

3.4. Neighborhood structures

To explore the search space of the TOPDM, NS were adapted from Nobibon et al. [10] and Oliveira et al. [14]. In their works, three different NS were used, which are briefly described below. It should be noted that the initial solution generator (Section 3.3) and these two NS are both designed to keep feasibility and walk through a feasible search space.

Swap Clients Intra – $NS^{SC^{Intra}}(s)$: This movement consists in swapping two positions, $l, m \in C$ of a given product $j \in O$, such that $s_{l,j} = s_{m,j}$ and $s_{m,j} = s_{l,j}$.

Swap Clients Inter – $NS^{SC^{Inter}}(s)$: Similar to the movement $N^{TC-Intra}(s)$, but in this case, two positions from different products $i, j \in O$ are swapped, e.g., $s_{l,i} = s_{m,j}$ and $s_{m,j} = s_{l,i}$.

$NS^{SC^{Intra}}(s)$ or $NS^{SC^{Inter}}(s)$ represent NS involving small changes in the solution, which is desirable for local search. However, the small steps performed in these NS might lead to slow convergence in large problems. For example, the largest problem dealt by Nobibon et al. [10], composed of 10,000 clients and 15 products, would give a search space with almost 50 millions different moves for $NS^{SC^{Intra}}(s)$.

In the next section, a novel Greedy Randomized NS is described, as well as an example of its use.

3.5. Greedy randomized NS

The concept of a Greedy Randomized NS (GRNS) derives from the same principle of GRASP [19], where a semi-greedy constructive method builds a solution step by step, inserting elements from a list of candidates (line 8 of Algorithm 1), according to a sorting criterion. This criterion is generally related to the best parts of the solution that are able to minimize the desired greedy function. An example was already presented in line 4 of Algorithm 1, where ANPP criterion balances profit and volatility, both related to the objective functions described in Section 3.2.

An example of a GRNS is the neighborhood $GRNS^{SP}(s)$, described in Algorithm 2, adapted from $NS^{SP}(s)$ [10].

Swap Products – $NS^{SP}(s)$: exchanges two bits from two different columns $i, j \in O$ of a given solution s, such that y_i=1 and y_j=0. Thus, a product offer which is not being used in the campaign (y_j=0) can be now part of the active set of products to be used during the campaign. A new product is added if, and only if, there are, at least, O_j available clients for receiving offers (after removing offers from y_i).

The main difference between the strategy introduced here is that a random product from the active set of products is removed (line 2 of Algorithm 2) and a new random one is selected from the set of the product offers which are not being used in the campaign, as can be seen in line 3. As already mentioned, a new product is only inserted in the campaign if there is a minimum O_j available clients in the set V and if operational constraints are still satisfied.

Algorithm 2. $GRNS^{SP}(s)$ move generation.

Input: solution s, $\gamma^{GRNS} \in [0, 1]$, set of available clients A, set of active products S

Output: move $m(s)$
j^{remove} ← product offer from the set of active offers S chosen at random
Update A and S by removing all targeted offers from j^{remove}
j←pick product offer at random from the set of inactive offers $\{O - S\}$
for each $i \in A$ **do**
  $ANPP_{ij} \leftarrow \frac{(p_{ij} v_{ij} - c_{ij})}{c_{ij}}$
end
Call the same procedure from Lines 8 to 18 of Algorithm 1, considering greedy parameter γ^{GRNS}
Update targeted offers of product j in solution s
Update A by removing all saturated clients
for each *active client* $i \in A$ **do**
  if $p_{ij} > c_{ij}$ *and the offer* j *directed to the client* i *generates a feasible solution* **then**
    $s_{ij} \leftarrow 1$
  end
end
return s

3.6. Multi-objective local search algorithm.

Some recent works in the literature have been trying to standardize and disseminate the use of Dominance-based Multi-objective Local Search (DMLS) methods [21]. The core of the generic algorithm used in this paper combines the flexibility provided by the metaheuristics GRASP [19] and Two-phase Pareto Local Search with VNS (2PPLS-VNS) [20]. Pareto Local Search [24–26] is a straightforward extension of the classical Hill–Climbing method. Nevertheless, we design a even more generic method, able to perform local searches considering different neighborhood exploration techniques. For those interested in this topic, different mechanisms for DMLS were pointed out by Liefooghe et al. [21] and the variants based on multi-objective variable neighborhood search methods were discussed by Duarte et al. [27]. The abbreviation G2PPLS-VNS is defined for our Generic 2PPLS-VNS, with its pseudo-code outlined in Algorithm 3.

Algorithm 3. Generic 2PPLS with VNS.

Input: Neighborhoods $N_k(x)$, *graspMaxSol* and $\gamma^{maxRange}$
Output: Approximation of the efficient set Xe
1 P_0←pfBuilder(graspMax,$\gamma^{maxRange}$)
2 Xe and P←P_0
3 $P_a \leftarrow \varnothing$
4 k←1
5 **while** $k \le r$ **do**
6   **forall** $p \in P$ **do**
7   NE$(Xe, p, Added, N_k(x), \ldots)$
8   **if** $Added = true$ **then**
9   update P_a with the new individuals added to Xe;
10   **end**
11   **end**
12   **if** $P_a \neq \varnothing$ **then**
13     $k \leftarrow 1$
14     $P \leftarrow P_a$ and $P_a \leftarrow \varnothing$
15   **else**
16     $k \leftarrow k + 1$
17     $P \leftarrow Xe \setminus \{x \in Xe|$Pareto local optimum for $N_k(x)\}$;
18   **end**
19 **end**
20 **retorna** Xe

In line 1 of Algorithm 3, an initial set of non-dominated solutions is generated by the procedure Pareto Front Builder (*pfBuilder*), described in Algorithm 4. Procedure *pfBuilder* generates *graspMax* solutions and calls addSolution (Algorithm 5, extracted from [28]) procedure for filtering those that are non-dominated.

Algorithm 4. pfBuilder.

Input: *graspMaxSol* and $\gamma^{maxRange}$
Output: Approximation of the efficient set Xe
for $i \leftarrow 1$ ***To*** *graspMaxSol* **do**
  $\gamma \leftarrow$ random value $in\,[0, \gamma^{maxRange}]$
  $s \leftarrow BuildGRASPSolution(\gamma)$
  $Xe' \leftarrow addSolution(Xe, s, f(p))$
end
return Xe

Algorithm 5. addSolution.

Input: Non-dominated population Xe; Solution s and its corresponding evaluations $z(s)$
Output: Xe and Added (optional)

Added←true
forall $x \in Xe$ **do**
  if $(z(x) \preceq z(s))$ **then**
    Added ← false
    Break
  end
  if $z(s) \prec z(x)$ **then**
    $Xe \leftarrow Xe \setminus x$
  end
end
if $Added = true$ **then**
  $Xe \leftarrow Xe \cup s$
end
return Xe

Algorithm 3 performs a Pareto Local Search according to the 2PPLS-VNS designed by Lust & Teghem [20]. This algorithm handles an auxiliary set P_a that contains the solution added in each iteration.

Line 7 explores the neighbors of each solution p from population P using any Neighborhood Exploration (NE) technique. In our designed strategy, the procedure NE, an abbreviation for any Neighborhood Exploration techniques, should update the set of non-dominated solutions X_e. An example of an Exhaustive Neighborhood Exploration (ENE) is given in Algorithm 6, which would result in the classical PLS algorithm. ENE procedure generates all possible neighbors from lines 1 to 3. However, a partial neighborhood exploration can be done with random moves, as exemplified in Algorithm 7.

If, at least one new non-dominated solution was found during the NE procedure (line 12 of Algorithm 3), the local search starts again from the first NS (line 13) and will search over the new obtained ones. Otherwise, if no new solution was found, line 16 makes the algorithm jump to the next available NS. A speed up is made in line 17, in a such way that the method will not repeat the neighborhood search over visited neighbors.

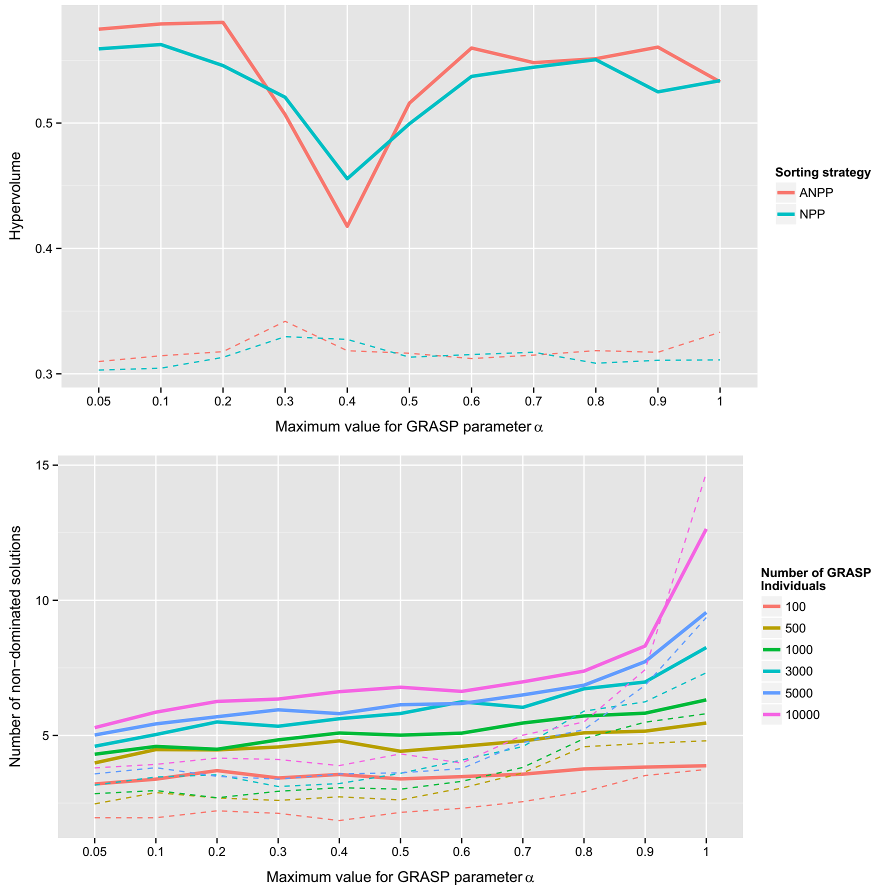

Fig. 3. Interaction plots of number of initial solutions, $\gamma^{maxRange}$ parameters and sorting strategy.

Algorithm 6. Exhaustive Neighborhood Exploration (ENE).

Input: Initial approximation of the efficient set *Xe*, solution *p* and Neighborhood *N*(*p*)
Output: *Xe* and Added (optional)

forall $p' \in N(p)$ **do**
| addSolution$(Xe, p', f(p'), Added)$
end
retorna *Xe*

Algorithm 7. Random Neighborhood Exploration (RNE).

Input: Initial approximation of the efficient set *Xe*, solution *p*, Neighborhood *N*(*p*) and number of random moves *mMax*
Output: *Xe* and Added (optional)

for $m = 1$ ***to*** *mMax* **do**
| $p' \leftarrow$ random move of $N(p)$
| addSolution$(Xe, p', f(p'), Added)$
end
retorna *Xe*

4. Computational experiments

This section is divided into six subsections. Section 4.1 presents the computational resources, some considerations about the code and algorithm parameters. Section 4.2 introduces the cases of study used in this paper. Section 4.3 checks the ability of the constructive in generating diversified solutions for composing initial sets of non-dominated solutions. Section 4.4 analyses the GRNS proposed in this paper. Section 4.5 reports the results considering all features of the proposed algorithm. Finally, Section 4.6 describes two Pareto fronts obtained using the proposed methodology.

4.1. Basic configurations

The metaheuristic algorithm was implemented in C++ in the framework OptFrame 2.2[1] [29]. This framework has been successfully applied to other problems in the literature, as can be seen in [30,31,18].

The tests were carried out on an OPTIPLEX 9010 Intel Core i7-3770, 3.40×8 GHZ with 32GB of RAM, with operating system Ubuntu 14.04 precise, and compiled by g++ 4.8.4, using the Eclipse Kepler Release.

4.2. Datasets

The set of instances was taken from Nobibon et al. [10]. The test problems comprised cases with 300, 2000 and 10,000 clients, respectively small, medium and large size cases. For each different set of clients, three instances with different number of possible offers were used: 5, 10 and 15 available products for the promotional campaign.

The expected return p_{ij} for each offer $j \in O$ directed to customer $i \in C$ is an integer between 0 and 16. Clients were grouped according to their expected profit. Volatility values v_{ij} were generated for each group, as described in Eq. (6). A maximum volatility of 0.6 can be returned by this formula.

[1] Available at http://sourceforge.net/projects/optframe/.

$$v_{ij} = \begin{cases} 0.01, & \text{if } p_{ij} < 3 \\ 0.05, & \text{if } 3 \le p_{ij} < 5 \\ 0.1, & \text{if } 5 \le p_{ij} < 8 \\ 0.25, & \text{if } 8 \le p_{ij} < 11 \\ 0.4, & \text{if } 11 \le p_{ij} < 14 \\ 0.6, & \text{if } p_{ij} \ge 14 \end{cases} \tag{6}$$

4.3. Checking GRASP ability of obtaining non-dominated solutions—pfBuilder procedure

The first batch of experiments sought to analyze the ability of the constructive procedure in finding good initial sets of non-dominated solutions, i.e., initial estimates of the Pareto front. Different sizes of the initial population and GRASP greedy parameters were verified: $graspMaxSol = [1, 10, 100, 500, 1000, 3000, 5000, 10, 000]$ and $\gamma^{maxRange} = [0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]$. The batch was composed of 11,673 executions, considering all possible configurations of both parameters.

Furthermore, the new *ANPP* strategy, which includes the variability v_{ij} in the original NPP calculus was also analyzed. In this sense, both sorting strategies *ANPP* and *NPP* were considered for the constructive procedure.

Obtained sets of non-dominated solutions were evaluated according to:

1. Hypervolume (HV) [32] quality indicator (using the computational tool provided by Beume et al. [33]);
2. Number of non-dominated solutions.

Fig. 3 shows one interaction plot between analyzed parameters and these Quality Indicators (QI). Values were normalized for the HV by subtracting the minimum values and dividing the result by the range of each indicator,

$$\frac{QI - \min(QI)}{\max(QI) - \min(QI)}$$

Dashed lines show the standard deviation while the continuous lines indicate average values. The higher the $\gamma^{maxRange}$, the more non-dominated solutions could be found, since solutions can be generated with more randomness. We decide to set this parameter to $\gamma^{maxRange} = 0.8$, providing an interesting balance between the HV, number of non-dominated solutions and standard deviation.

4.4. Checking PLS with the GRNS

This second batch intended to check if the proposed GRNS was able to improve the quality of the initial estimates of the Pareto Front. Thus, we verify its effectiveness regarding different greedy parameters $\gamma^{GRNS} = \{0.05, 0.1, 0.2, 0.5, 0.7, 0.9, 1\}$, $iterMax^{GRNS} = \{1, 5, 10, 50, 100\}$ and $graspMaxSol = [1, 10, 100, 500]$ In the same way as proceeded in the last batch of experiments, 5000 random runs, with a restricted computational time of 5 min, were performed considering all combinations of these values.

This randomized greedy NS can be exhaustively searched following the procedure defined in Algorithm 6. The number of neighbors is the amount of possible swaps between products (105 for the case involving 15 products, $\frac{15 * 14}{2}$). However, due to the greedy parameter γ^{GRNS}, different clients can be targeted each time a move is generated. Fig. 4 illustrates the obtained results considering both aforementioned *QI* plus a diversity indicator, calculated with the Δ metric [34];

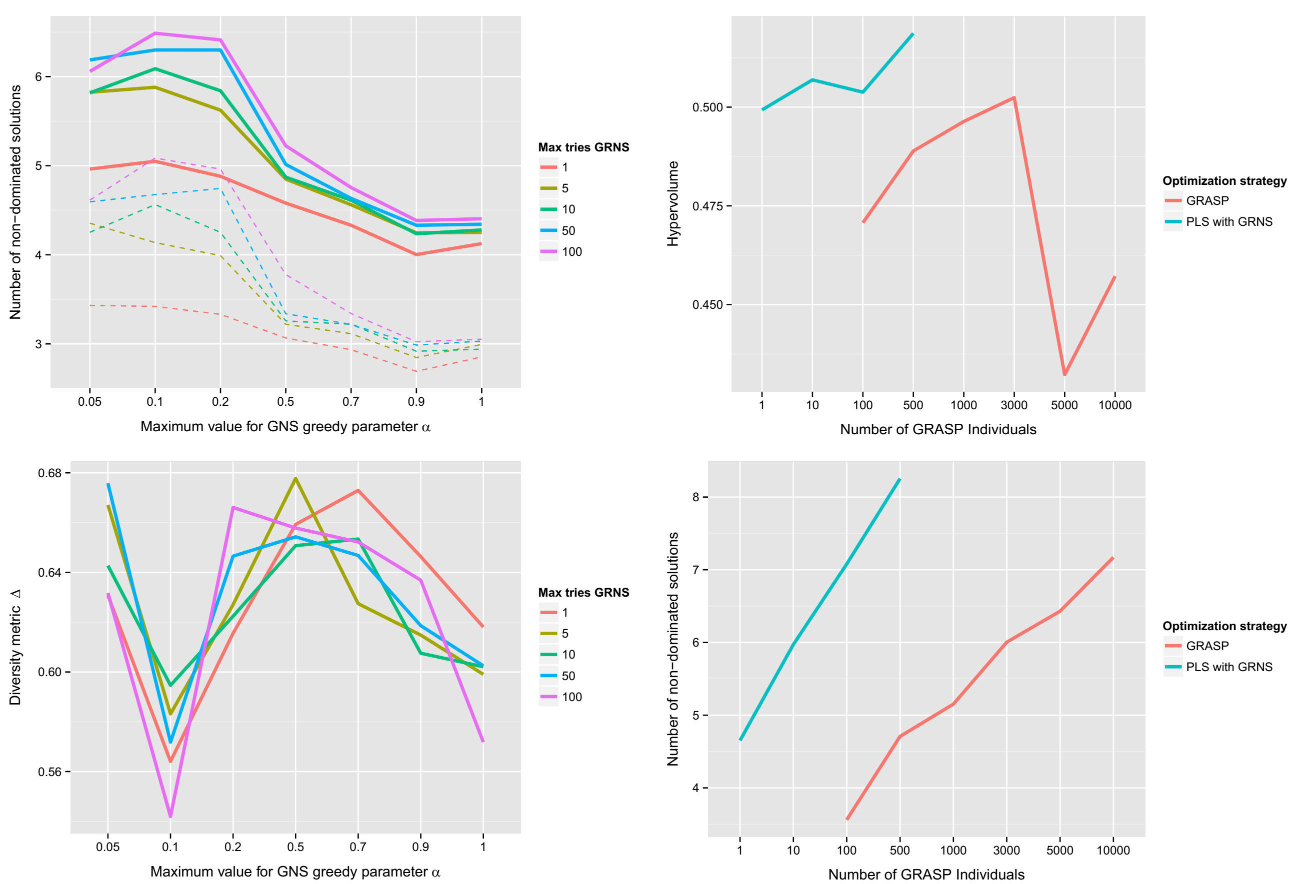

Fig. 4. Interaction plots of different indicators, *GRNS* parameters and optimization strategy.

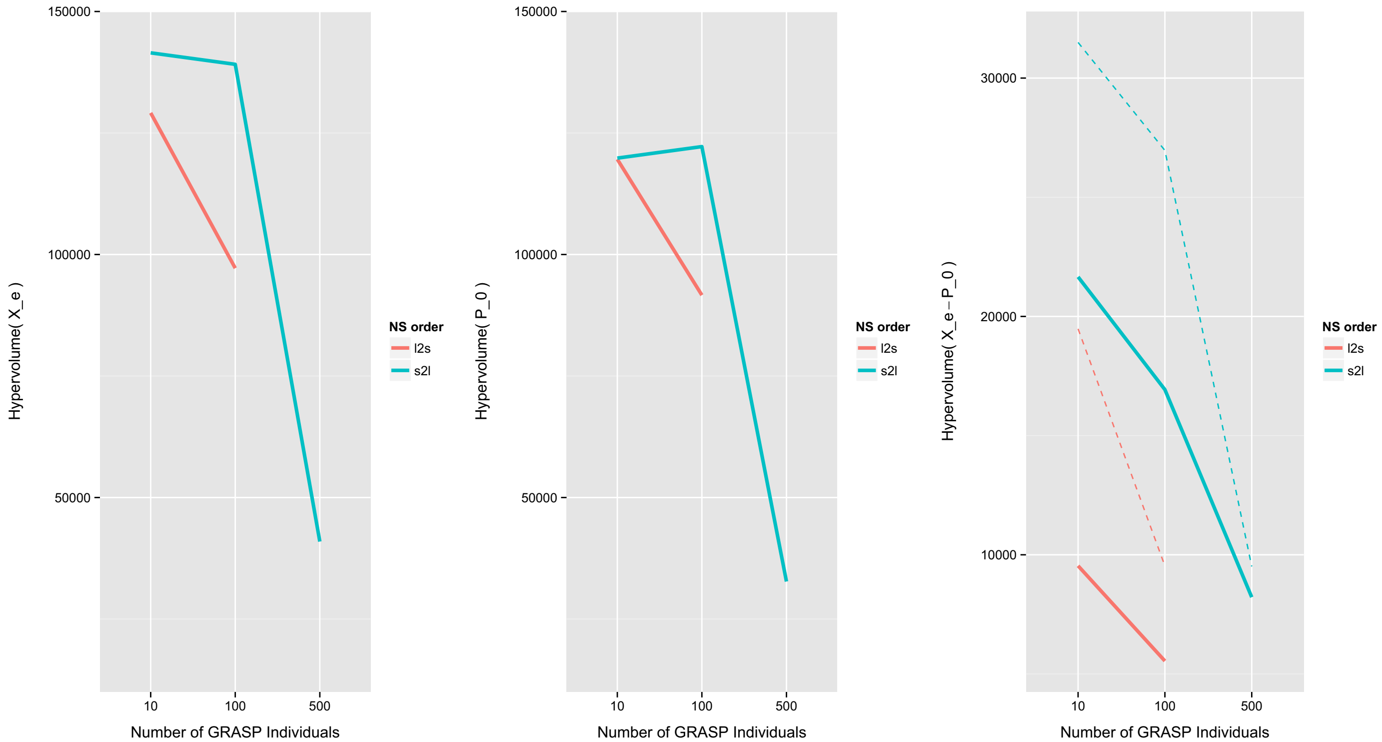

Fig. 5. Interaction plots of different indicators, checking local search improvement.

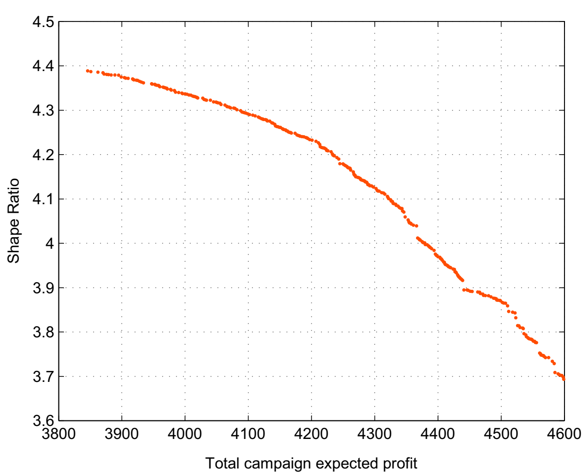

Fig. 6. Set of non-dominated solutions for instance “S3-10-15-1-s”.

As expected, whenever an ENE is performed ($iterMax^{GRNS}$), the quality of the estimate set of non-dominated solutions is improved. Thus, parameter γ^{GRNS} was fixed to 0.1 for the following analysis.

4.5. Checking complete Generic 2PPLS with VNS

This third and last batch of experiments aimed at analyzing the performance of the proposed G2PPLS-VNS. For this purpose, two different configurations were analyzed:

1. searching from the largest to the smallest neighborhoods, *l2s* ($NS^{SC^{Inter}}(s)$, $NS^{SC^{Intra}}(s)$ and $GRNS^{SP}(s)$, respectively);
2. exploring from the smallest to the largest neighborhoods, *s2l*, ($GRNS^{SP}(s)$, $NS^{SC^{Intra}}(s)$ and $NS^{SC^{Inter}}(s)$, respectively).

As pointed out in Section 3.4, it would be a huge computational effort to run ENE for the neighborhood $NS^{SC^{Inter}}(s)$ and $NS^{SC^{Intra}}(s)$. In this sense, two RNE (Algorithm 7) were created for each of these NS with *mMax*=1000. The first two graphics of Fig. 5 show the HV of the final estimate of the Pareto Front (after G2PPLS-VNS refinement) and of its respective initial set of non-dominated solutions, while the third one shows the HV improvement.

Table 1
Targeted offers according to clients risk – Different non-dominated solutions characteristics.

Expected profit	Sharpe ratio	Client risk – v_{ij}					
		0.01	0.05	0.1	0.25	0.4	0.6
Small instance with 100 clients and 15 products							
4599	3.69364	0	2	23	96	192	253
4311	4.11411	0	0	24	99	189	211
4177	4.24447	1	1	40	109	176	196
3990	4.3399	1	9	54	111	172	176
3883	4.37944	1	8	62	121	171	160
3846	4.38868	0	12	66	118	169	158
Large instance “L-10-15-1-l” with 10,000 clients and 15 products							
240,218	1.63729	4758	7631	14250	14851	15075	15045
239,108	1.92499	4617	7018	12989	13586	13816	13811
238,344	1.9392	5250	6993	12929	13613	13791	13736
217,520	2.03421	5020	6206	11503	12176	12330	12231
217,774	2.01374	4464	6210	11552	12145	12347	12294
192,945	2.09869	3422	5161	9804	10431	10725	10661
192,934	2.12584	3936	5161	9759	10465	10713	10601

4.6. Obtained sets of non-dominated solutions

Two different test problems were used for illustrating the Pareto Fronts that can be obtained using the proposed methodology. A single run of 10 min was performed for each case and the obtained sets of non-dominated solutions (or parts of it) are illustrated in Fig. 6 and Table 1.

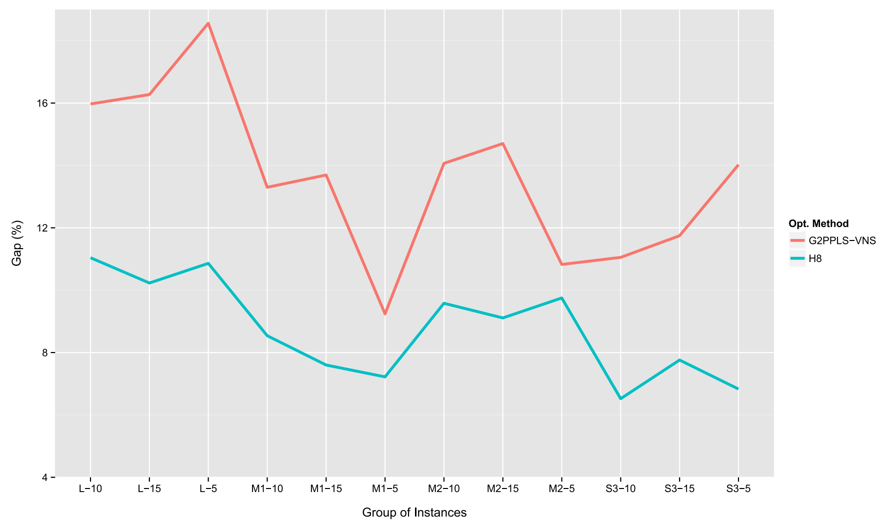

Fig. 7. Univariate objective function comparison.

Table 1 shows characteristics of some of the non-dominated solutions obtained in each case. As can be noticed, those different solutions illustrate several possible scenarios for conducting the promotional campaign. Each of these possible sets of targeted offers indicate a specific expected profit and Sharpe Ratio associated to it.

Finally, Fig. 7 computes the maximum expected profit of each execution from the batch of experiments of this section. In order to compare the performance of the proposed algorithm in terms of minimizing a single objective, the GAP metric ($gap_i^n = \frac{f_i^* - f_i^n}{f_i^*}$) was used, with f_i^* being the best known result for a given test-problem and $f_i{}^n$ the value obtained by each algorithm. Average gaps of the G2PPLS-VNS algorithm are compared with the single values reported from a Tabu Search algorithm (H8) [10].

5. Conclusions and extensions

5.1. Summary and final considerations

In this paper, a bi-objective direct-marketing promotional campaign was discussed by simultaneously optimizing campaign profits and a reward-to-variability index, adapted from the Sharpe Ratio. A multi-objective DMLS metaheuristic was proposed for searching for sets of non-dominated solutions.

Due to the large number of neighborhoods that can be searched, a generic Pareto Local Search was introduced. In order to produce a diversified initial estimate of the Pareto Front, a greedy randomized initial solution builder was proposed for dealing with the concept of volatility. Furthermore, a special case involving a Greedy Randomized Neighborhood Structure, which reconstructs parts of the solution, was described and evaluated.

By adapting test problems from the literature, different characteristics from the obtained Pareto Fronts were described and analyzed. Sharpe ratio index was able to regulate the search for low-risk direct marketing campaigns, providing a trade-off between campaign total profit and the groups of clients which the offers are directed to.

5.2. Extensions

As future extensions for this research, the current approach should be applied in other types of direct marketing campaigns.

The development of new neighborhood structures might improve the ability of the Pareto Local Search in finding non-dominated solutions from the space of solutions. A parallel version of the method could improve the performance of the model over problems with large amount of data. This approach would take advantage of the multi-core technology that is already integrated in the current machines, and with easy abstraction for this metaheuristic algorithm.

The entire code used in this research is, from this moment, available as example on the OptFrame website. Thus, it is expected that future researchers continue contributing to enhancing the proposed model, increasing its efficiency and improving the tools and ideas presented in this paper.

Acknowledgment

The authors would like to thank Brazilian agency CAPES, CNPq (grants 305506/2010-2, 552289/2011-6, 306694/2013-1 and 312276/2013-3), FAPEMIG (grants APQ-04611-10, PPM CEX 497-13) and FP7 CORDIS, “New Horizons for Multi Criteria Decision Making”, for supporting the development of this work. Helena Ramalhinho was also partially supported by the Spanish Ministry of Economy and Competitiveness (TRA2013-48180-C3-P, TRA2015-71883-REDT).

The authors are indebted to the three anonymous reviewers for their constructive suggestions that have helped us improve the original manuscript. Authors would like to thank the organizers of the 3rd International Conference on Variable Neighborhood Search, without their motivation for extending the previous work, this one would not have become reality. Finally, Vitor would like to thanks Roel Leus and Fabrice Talla for their suggestions and information for researching this topic.

References

[1] Sharpe WF. The sharpe ratio. J Portf Manag 1994;21(1): 49–58. http://dx.doi.org/10.3905/jpm.1994.409501.

[2] Chow V, Lai CW. Conditional sharpe ratios. Financ Res Lett 2015;12(0): 117–33. http://dx.doi.org/10.1016/j.frl.2014.11.001.

[3] Murali S, Pugazhendhi S, Muralidharan C. Modelling and investigating the relationship of after sales service quality with customer satisfaction, retention and loyalty—a case study of home appliances business. J Retail Consum Serv 2016;30: 67–83. http://dx.doi.org/10.1016/j.jretconser.2016.01.001.

[4] Ghazavi E, Lotfi M. Formulation of customers' shopping path in shelf space planning: a simulation-optimization approach. Expert Syst Appl 2016;55: 243–54. http://dx.doi.org/10.1016/j.eswa.2016.01.043.

[5] Cohen M-D. Exploiting response models: optimizing cross-sell and up-sell opportunities in banking. Inf Syst 2004;29(4):327–41.

[6] Mahajan V, Muller E, Bass FM. New Product Di usion Models in Marketing: A Review and Directions for Research. In: Nebojša Nakićenović, Arnulf Grübler, editors. Diffusion of technologies and social behavior. Berlin, Heidelberg: Springer Berlin Heidelberg; 1991. p. 125–77. http://dx.doi.org/10.1007/978-3-662-02700-4_6, isbn=978-3-662-

02700-4.
[7] Lönnstedt L. The use of operational research in twelve companies quoted on the stockholm stock exchange. Oper Res Q (1970–1977) 1973;24(4):535–45, URL ⟨http://www.jstor.org/stable/3008331⟩.
[8] Daniel Jr JN, Busyn TF, Batterman BT. Marketing research system and method for obtaining retail data on a real time basis. uS Patent 4,972,504 ; 1990.
[9] Bhaskar T, Sundararajan R, Krishnan PG. A fuzzy mathematical programming approach for cross-sell optimization in retail banking. J Oper Res Soc 2009;60(5):717–27.
[10] Nobibon FT, Leus R, Spieksma FC. Optimization models for targeted offers in direct marketing: exact and heuristic algorithms. Eur J Oper Res 2011;210(3):670–83.
[11] Barnhart C, Johnson EL, Nemhauser GL, Savelsbergh MWP, Vance PH. Branch-and-price: column generation for solving huge integer programs. Oper Res 1998;46(3):316–29.
[12] Glover F, Laguna M. Tabu search. Boston: Kluwer Academic Publishers; 1997.
[13] Delanote S, Leus R, Nobibon FT. Optimization of the annual planning of targeted offers in direct marketing. J Oper Res Soc 2013;64(12):1770–9. http://dx.doi.org/10.1057/jors.2012.165.
[14] Oliveira TA, Coelho VN, Souza MJF, Boava DLT, Boava F, Coelho IM, et al. A hybrid variable neighborhood search algorithm for targeted offers in direct marketing. Electron Notes Discret Math 2015;47(0):205–12. http://dx.doi.org/10.1016/j.endm.2014.11.027, the 3rd international conference on variable neighborhood search (VNS'14).
[15] Hansen P, Mladenovic N, Pérez JAM. Variable neighborhood search. Eur J Oper Res 2008;191:593–5.
[16] Kirkpatrick S. Optimization by simulated annealing: quantitative studies. J Stat Phys 1984;34(5–6):975–86.
[17] Mladenovic N, Hansen P. A variable neighborhood search. Comput Oper Res 1997;24:1097–100.
[18] Coelho V, Grasas A, Ramalhinho H, Coelho I, Souza M, Cruz R. An ils-based algorithm to solve a large-scale real heterogeneous fleet {VRP} with multi-trips and docking constraints. Eur J Oper Res 2016;250(2):367–76. http://dx.doi.org/10.1016/j.ejor.2015.09.047.
[19] Feo TA, Resende MGC. Greedy randomized adaptive search procedures. J Global Optim 1995;6:109–33.
[20] Lust T, Teghem J, Tuyttens D. Very large-scale neighborhood search for solving multiobjective combinatorial optimization problems. In: Takahashi R, Deb K, Wanner E, Greco S, editors, Evolutionary multi-criterion optimization, Lecture notes in computer science, vol. 6576. Springer Berlin/Heidelberg; 2011. p. 254–68.
[21] Liefooghe A, Humeau J, Mesmoudi S, Jourdan L, Talbi E-G. On dominance-based multiobjective local search: design, implementation and experimental analysis on scheduling and traveling salesman problems. J Heuristics 2012;18(2):317–52. http://dx.doi.org/10.1007/s10732-011-9181-3.
[22] Kellerer H, Pferschy U, Pisinger D. Knapsack problems. Springer; 2004.
[23] Praag NV. Optimization of promotion campaigns using tabu search, Master's thesis, Faculty of Business and Economics, KULeuven (2010).
[24] Paquete L, Chiarandini M, Stützle T. Metaheuristics for multiobjective optimisation. Pareto local optimum sets in the biobjective traveling salesman problem: an experimental study. Springer Berlin Heidelberg, Berlin, Heidelberg; 2004. p. 177–99. http://dx.doi.org/10.1007/978-3-642-17144-4_7.
[25] Angel E, Bampis E, Gourvès L. Metaheuristics for multiobjective optimisation. A dynasearch neighborhood for the bicriteria traveling salesman problem; 2004. p. 153–76. http://dx.doi.org/10.1007/978-3-642-17144-4_6.
[26] Talbi E-G, Rahoual M, Mabed MH, Dhaenens C. Evolutionary Multi-Criterion Optimization: First International Conference, EMO 2001 Zurich, Switzerland, March 7–9, 2001 Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, Ch. A Hybrid Evolutionary Approach for Multicriteria Optimization Problems: Application to the Flow Shop, p. 416–28. http://dx.doi.org/10.1007/3-540-44719-9_29.
[27] Duarte A, Pantrigo JJ, Pardo EG, Mladenovic N. Multi-objective variable neighborhood search: an application to combinatorial optimization problems. J Global Optim 2015;63(3):515–36. http://dx.doi.org/10.1007/s10898-014-0213-z.
[28] Lust T, Teghem J. Two-phase Pareto local search for the biobjective traveling salesman problem. J Heuristics 2010;16:475–510.
[29] Coelho IM, Munhoz PLA, Haddad MN, Coelho VN, Silva MM, Souza MJF. A computational framework for combinatorial optimization problems. In: VII ALIO/EURO workshop on applied combinatorial optimization, Porto, 2011. p. 51–4.
[30] Souza MJF, Coelho IM, Ribas S, Santos HG, Merschmann LHC. A hybrid heuristic algorithm for the open-pit-mining operational planning problem. Eur J Oper Res 2010;207(2):1041–51.
[31] Coelho VN, Coelho IM, Coelho BN, Reis AJ, Enayatifar R, Souza MJ, et al. A self-adaptive evolutionary fuzzy model for load forecasting problems on smart grid environment. Appl Energy 2016;169:567–84. http://dx.doi.org/10.1016/j.apenergy.2016.02.045.
[32] Zitzler E, Thiele L. Multiobjective optimization using evolutionary algorithms - a comparative case study. In: Eiben A, Back T, Schoenauer M, Schwefel H-P, editors, Parallel problem solving from nature - PPSN V, Lecture notes in computer science, vol. 1498, Springer Berlin/Heidelberg, 1998. p. 292–301. http://dx.doi.org/10.1007/BFb0056872.
[33] Beume N, Fonseca C, Lopez-Ibanez M, Paquete L, Vahrenhold J. On the complexity of computing the hypervolume indicator. IEEE Trans Evol Comput 2009;13(5):1075–82. http://dx.doi.org/10.1109/TEVC.2009.2015575.
[34] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 2002;6(2):182–97. http://dx.doi.org/10.1109/4235.996017.