Improving traditional subgradient scheme for Lagrangean relaxation: an application to location problems

Luiz A.N. Lorena
LAC/INPE - Instituto Nacional de Pesquisas Espaciais
Av. dos Astronautas, 1758 - Caixa Postal 515
12227-010 São José dos Campos, SP - Brazil
e-mail: lorena@lac.inpe.br

Edson L.F. Senne
FEG/UNESP - Universidade Estadual Paulista
Faculdade de Engenharia - Departamento de Matemática
12500-000 Guaratinguetá, SP - Brazil
e-mail: elfsenne@feg.unesp.br

Abstract

Lagrangean relaxation is largely used to solve combinatorial optimization problems. A known problem for Lagrangean relaxation application is the definition of convenient step size control in subgradient like methods. Even preserving theoretical convergence properties, a wrong defined control can reflect in performance and increase computational times, a critical point in large scale instances. We show in this work how to accelerate a classical subgradient method, using the local information of the surrogate constraints relaxed in the Lagrangean relaxation. It results in a one-dimensional search that corrects the step size and is independent of the step size control used. The application to Capacitated and Uncapacitated Facility Location problems is shown. Several computational tests confirm the superiority of this scheme.

Key words: Location problems, Lagrangean relaxation, Subgradient method.

1. Introduction

Facility location is the problem of locating a number of facilities from a subset of m potential facility locations to completely satisfy at minimum cost all the demands of a set of n clients. Each client has an associated demand and the costs considered include transportation and fixed costs for opening facilities. For the *Uncapacitated Facility Location Problem* (*UFLP*) the facilities are assumed to have unlimited capacity, and for the *Capacitated Facility Location Problem* considered in this paper (*CFLP*) there are constraints on the total demand that can be met from a facility.

Both the problems are well known to be NP-hard [21]. Successful algorithms have been developed for the *UFLP* and have appeared in Beasley [4], Cornuéjols, Fisher and Nemhauser [11], Erlenkotter [16], Galvão and Raggi [20], Guinard [27], and Korkel [37]. The capacitated problem and its variations have been intensively studied in the literature. Cornuéjols, Sridharan and Thizy [12] presented a computational and theoretical comparison of various relaxations proposed in the literature, covering the main mathematical formulations for the problem. Surveys papers can be found in Domschke and Drexl [14] and Krarup and Pruzan [38]. Optimal algorithms and heuristics appeared in Baker [2], Barceló, Fernandez and Jörnsten [3], Beasley [5,6], Chistofides and Beasley [10], Jacobsen [33], and Nauss [45]. Beasley [6] describes Lagrangean heuristics for a class of location problems, including *UFLP* and the *CFLP* considered here.

Lagrangean relaxation is a well known relaxation technique frequently used to give bound information to combinatorial optimization problems [see for example the survey papers [18, 22] and the book [46]). A widely used method to optimize the Lagrangean dual is the well-known subgradient method of Polyak [47]. Although of simple convergence conditions [17,47], the convergence of subgradient methods can consume a long time for some instances. The subgradient optimization is very sensitive to the initial values for the multipliers and the rules applied for step size controlling. Efforts were made to have theoretical foundations for these choices [4, 25], but until today the most popular approaches are based on previous empirical experience [31].

Reducing the initial erratic behavior of the subgradient method can result in fast convergence. For large scale problems that can be interesting, even with the use of fast computers. The Lagrangean relaxation can be adapted to use local information (optimization) provided by the surrogate constraints to accelerate the subgradient method. The idea is to introduce a local optimization step at the initial iterations of a subgradient method. The first relaxation is a surrogate followed by a Lagrangean relaxation of the surrogate constraint. A local Lagrangean dual optimization is approximately solved. The process is repeated for a controlled number of iterations of the subgradient method. The implementation of this scheme produced less computational times for the *UFLP* and *CFLP* instances considered.

The use of surrogate information in Lagrangean relaxation context was tested before with success on Set Covering problems [1,19,40], Generalized Assignment problems [41] and p-median problems [48]. Notably is the gain in computer times for large scale instances.

We present in next section how to use local surrogate information on Lagrangean relaxation. In section three, Lagrangean/surrogate relaxations are particularized to the *UFLP* and *CFLP* considered in this paper. Section four presents the general relaxation heuristics considered for application of Lagrangean/surrogate and Lagrangean heuristics. We then present computational tests considering instances taken from the literature, mainly for large scale instances.

2. The improved subgradient method

To best describe the modification scheme to the subgradient method, we start describing in general terms how to make profit of the surrogate information considered when a Lagrangean relaxation is performed.

In general terms, suppose the following 0-1 linear programming problem:

$$(P) \qquad \begin{aligned} v(P) = Min\ \ & cx \\ subject\ \ to\ \ & Ax \geq b, \\ & Dx \geq e, \\ & x \in \{0,1\}^n \end{aligned}$$

where $c \in R^n, A \in R^{mxn}, b \in R^m, D \in R^{pxn} and\ e \in R^p$.

At this point is only necessary to think of $Dx \geq e$ as the easily enforced constraints and $Ax \geq b$ as the complicating ones. Defining X = { $x \in \{0,1\}^n \mid Dx \geq e$ }, for a given multiplier $\omega \in R_+^m$ the Lagrangean relaxation of (P) is

$$(LP^\omega) \qquad \begin{aligned} v(LP^\omega) = Min\ \ & \{cx - \omega(Ax - b)\} \\ subject\ \ to\ \ & x \in X. \end{aligned}$$

The Lagrangean dual is then the optimization problem in ω ,

$$(D^\omega) \qquad \begin{aligned} v(D^\omega) = Max\ \ & \{v(LP^\omega)\} \\ subject\ \ to\ \ & \omega \in R_+^m. \end{aligned}$$

The surrogate duality theory is an old matter, that was not so intensively explored as the Lagrangean relaxation (see the papers [15, 23, 24, 26, 34] and the book [46] for a formal view of the subject). We explore here the simple relationship between the two relaxations, recalling that Lagrangean multipliers can also be considered surrogate multipliers, and making profit of the local optimization proportioned by the new local Lagrangean relaxation.

For a given $\lambda \in R_+^m$, the surrogate problem of (P) is

$$(SP^{\lambda}) \qquad \begin{aligned} v(SP^{\lambda}) = Min \quad & cx \\ subject\ \ to \quad & \lambda(Ax-b) \geq 0, \\ & x \in X. \end{aligned}$$

Frequently (SP^{λ}) is a difficult problem (like (P)). A linear programming relaxation (surrogate continuous) $(SP^{\lambda})_{linear}$ of (SP^{λ}) can be considered substituting X by $X_{linear} = \{x \in [0,1]^n \mid Dx \geq e\}$ in (SP^{λ}).

Consider now a relaxation in the Lagrangean way of problem (SP^{λ}). Given $\lambda \in R_+^m$, and a parameter $t \geq 0$, the ***Lagrangean/surrogate*** relaxation of (P) is

$$(L_t SP^{\lambda}) \qquad \begin{aligned} v(L_t SP^{\lambda}) = \ & Min \ \{cx - t.\lambda(Ax-b)\} \\ subject\ \ to \quad & x \in X. \end{aligned}$$

A Lagrangean/surrogate dual can be identified here as

$$(D^{t\lambda}) \qquad \begin{aligned} v(D^{t\lambda}) = Max \ \ & \{v(L_t SP^{\lambda})\} \\ subject\ \ to \quad & t.\lambda \in R_+^m. \end{aligned}$$

It is immediate that setting $\omega = t.\lambda$, problem $(D^{t\lambda})$ is the Lagrangean dual (D^{ω}), and the optimal lower bound limits coincide for both Lagrangean and Lagrangean/surrogate approach.

A local dual can be identified using the Lagrangean/surrogate relaxation. Suppose λ fixed, the one-dimensional dual in t is

$$(D_t^{\lambda}) \qquad \begin{aligned} v(D_t^{\lambda}) = Max \ \ & \{v(L_t SP^{\lambda})\} \\ subject\ \ to \ \ & t \geq 0. \end{aligned}$$

When set X has the integrality property $v(D_t^{\lambda}) = v(SP^{\lambda})_{linear}$ [22]. In general we have $v(L_t SP^{\lambda}) \leq v(D_t^{\lambda}) \leq v(D^{t\lambda}) \leq v(P)$. The attractive characteristic of relaxation $(L_t SP^{\lambda})$, is that for $t = 1$ we have the usual Lagrangean relaxation using the multiplier λ. An exact solution to (D_t^{λ}) may be obtained by a search over different values of t (see Handler and Zang [28] and Minoux [44]). However we have a range of values $t_0 \le t^* \le t_1$ with $t_0 = 1$ or $t_1 = 1$ which also produces improved bounds (for $t_1 = 1$ see *figure* 1).

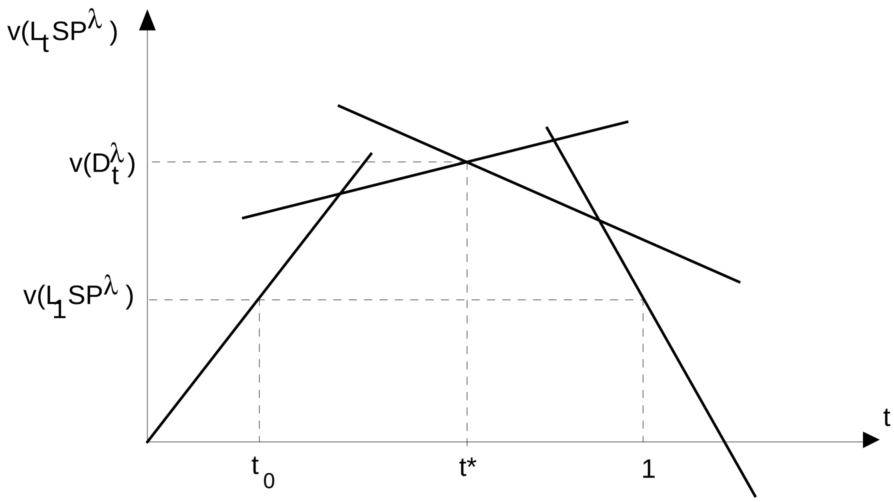

Figure 1: Lagrangean/surrogate bounds.

Alternatively to solve $(SP^\lambda)_{linear}$ at each iteration (justified only when the set X has the integrality property [22], and performed on early works on the subject [1, 40, 41]), a naive line search is used to calculate a t^* belonging to the interval $t_0 \le t^* \le t_1$ ($t_0 = 1$ or $t_1 = 1$). For a fixed λ, the following one-dimensional line search procedure is used:

Search Heuristic (SH)

Let

s	be the initial step size;
k	be the number of iterations;
kmax	be the maximum number of iterations;
t_0	be the initial value of Lagrangean/surrogate multiplier;
t	be the current value of Lagrangean/surrogate multiplier;
t*	be the value of Lagrangean/surrogate multiplier;
z	be the maximum value of $(L_t SP^\lambda)$;

Set

k = 0;
z = 0;
$t = t_0$;

t* = t;
$t^+ = t^- =$ undefined;

Repeat

k = k + 1;

Solve $(L_t SP^\lambda)$ obtaining x^λ

If ($v(L_t SP^\lambda)$ > z) **then**

Set

z = $v(L_t SP^\lambda)$;

t* = t;

Calculate $\mu^\lambda = \lambda(Ax^\lambda - b)$ (μ^λ is the slope of the Lagrangean/surrogate function);

If ($\mu^\lambda < 0$) **then**

$t^- = t$;

If (t^+ is undefined) **then**

t = t + s;

Else

Try to improve the current multiplier solving $(L_{(t^+ + t^-)/2} SP^\lambda)$, updating t* if necessary and **Stop.**

End_If

Else

$t^+ = t^*$;

If (t^- is undefined) **then**

t = t – s;

Else

Try to improve the current multiplier solving $(L_{(t^+ + t^-)/2} SP^\lambda)$, updating t* if necessary and **Stop.**

End_If

End_If

Else

Try to improve the current multiplier solving $(L_{t-s/2} SP^\lambda)$, updating t*= t if necessary and **Stop.**

End_If

Until (k < kmax).

Locally the Lagrangean/surrogate can provide better bounds than their Lagrangean (alone) counterpart. For a fixed λ, the following inequalities are valid

$$v(L_1SP^{\lambda}) \le v(L_{t*}SP^{\lambda}) \le v(D_t^{\lambda}) \le v(SP^{\lambda}) \le v(P).$$

Considering now the application of a traditional subgradient method to solve (D^{ω}), a subgradient is directly identified after solving (LP^{ω}) as $g^{\omega} = Ax^{\omega} - b$, where x^{ω} is an optimal solution of (LP^{ω}). The subgradient method updates the multiplier ω_k as $\omega_{k+1} = \omega_k + \theta_k g^{\omega_k}$, and simple rules for the step sizes establish well known convergence conditions [17,47]. The same convergence rules can be directly identified when the multiplier $t*\lambda$ is used. Using the Lagrangean/surrogate bound $v(L_{t*}SP^{\lambda})$ at the Held and Karp [29, 30, 31] step size rule can be attractive. The subgradient direction g_{t*}^{λ} obtained from problem (L_tSP^{λ}) can give a different direction of the subgradient obtained when problem (L_1SP^{λ}) is solved. Therefore different sequences of relaxation bounds were obtained beginning with the same initial multiplier λ.

Other subgradient methods appeared in literature [8,9,13,35,36,39]. More elaborated, they increase the local computational times computing descent directions [13], or combining subgradients of previous iterations [8,9], or realizing projections onto general convex sets [35,36,39]. Experimental results with some of these methods show an improvement in performance compared to the subgradient method [35,39]. It appears that the local search induced by the surrogate information is independent of the subgradient method employed, then we decided to investigate the use of this scheme at the traditional subgradient method that remains the widely used approach in the Lagrangean relaxation context.

The Lagrangean relaxations for *UFLP* and *CFLP* and corresponding Lagrangean/surrogate are derived in next section. The local optimization procedure SH contributes for the reduction in the oscillating behavior of the sequence $\{v(L_{t^*}SP^{(\cdot)})\}$, but it increases the local computational times resulting in the k^{λ} evaluations of $v(L_t SP^{\lambda})$. In general, the oscillating behavior of subgradient sequences is very accentuated at the first steps, then we propose to use the procedure SH to the point that the same t^* is found for a number (*n_consec*) of consecutive λ. The computational tests of section five confirm the feasibility of this approach for a set of *UFLP and CFLP* instances.

3. Lagrangean/Surrogate Relaxation for *UFLP* and *CFLP*

One objective of this paper is to show the local search benefits on the application of a standard subgradient method in Lagrangean relaxation context. Then the problems chosen are classical facility location problems, first used before in another papers [11,10], and are modeled as an integer programming problem, also denoted as (P).

In the case of *UFLP* [see 11], problem (P) is a binary integer programming problem given by:

$$v(P) = \min \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} F_i y_i$$

subject to:

$$\sum_{i=1}^{m} x_{ij} = 1 \quad ; \quad j \in N = \{1, \ldots, n\} \tag{1}$$

and the additional constraints:

$$x_{ij} \le y_i \quad ; \quad i \in M = \{1, \ldots, m\}\,,\ j \in N$$

$$x_{ij}\,,\ y_i \in \{0, 1\} \quad ; \quad i \in M\,,\ j \in N$$

where:

$[c_{ij}]_{mxn}$ is a symmetric transportation cost matrix;

$[x_{ij}]_{mxn}$ is the allocation matrix, where x_{ij} = 1 if customer j's demand is satisfied by facility i; x_{ij} = 0, otherwise;

$[y_i]_m$ is the vector indicating opened or closed facilities, with $y_i = 1$ if the facility is open; $y_i = 0$, otherwise;

$[F_i]_m$ is the vector of fixed costs for opening facilities;

m is the number of potential facility locations and *n* is the number of clients.

On the other hand, problem (P) for *CFLP* [see 10] is a mixed binary integer programming problem, which is defined as:

$$v(P) = \text{Min} \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} F_i y_i$$

subject to (1) and the additional constraints:

$$\sum_{j=1}^{n} d_j x_{ij} \le s_i y_i;\ \ i \in M = \{1,\ldots,m\}$$

$$x_{ij} \le y_i;\ i \in M,\ j \in N$$

$$\sum_{i=1}^{m} s_i y_i \ge \sum_{j=1}^{n} d_j$$

$$0 \le x_{ij} \le 1,\ y_i \in \{0,1\};\ i \in M,\ j \in N.$$

where:

m, n, $[c_{ij}]_{mxn}$, $[y_i]_m$ and $[F_i]_m$ are as above for *UFLP*;

$[x_{ij}]_{mxn}$ is the allocation matrix, where x_{ij} is the fraction of demand of client j, that is delivered from facility i;

$[s_i]_m$ is the vector of capacities of the facilities (i.e. the upper limit on the total demand that can be supplied from facility i);

$[d_j]_n$ is the vector of demands of the clients.

Contraint $\sum_{i=1}^{m} s_i y_i \ge \sum_{j=1}^{n} d_j$, also used by Beasley [6], is a surrogate constrait introduced to tighten the bounds when contraints (1) are relaxed. For a given $t \ge 0$, constraints (1) are then relaxed, and the lagrangean/surrogate relaxation, denoted as ($L_t SP^{\lambda}$), is given by:

$$v(L_tSP^{\lambda}) = \text{Min} \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} F_i y_i + t\sum_{j=1}^{n}\lambda_j(\sum_{i=1}^{m} x_{ij} - 1)$$

subject to the additional constraints of (P).

For the *UFLP*, (L_tSP^{λ}) is easily solved observing that setting $y_i = 1$ (i.e., facility i is open) the contribution to the objective function is:

$$\gamma_i = F_i + \sum_{j=1}^{n} \min[0, \ (c_{ij} + t\,\lambda_j)] \ , \ i \in M$$

The set I of open facilities is then defined by I = { $i \in M \mid \gamma_i \leq 0$ }. The allocation variables are then directly derived setting $x_{ij} = 1$ if $(c_{ij} + t\,\lambda_j) \leq 0$ and $y_i = 1$ (otherwise $x_{ij} = 0$).

In the case of *CFLP*, (L_tSP^{λ}) is solved separating into m subproblems. For a given $i \in M$ define the following m problems $(L_tSP^{\lambda})_i$:

$$v(L_tSP^{\lambda})_i = \text{Min} \sum_{j=1}^{n}(c_{ij} + t\lambda_j)x_{ij} + F_i y_i$$

subject to:

$$\sum_{j=1}^{n} d_j x_{ij} \leq s_i y_i$$

$$x_{ij} \leq y_i;\ j \in N$$

$$0 \leq x_{ij} \leq 1,\ y_i \in \{0,1\};\ j \in N.$$

Setting $y_i = 1$, each problem $(L_tSP^{\lambda})_i$ is a continue single constraint 0-1 knapsack problem which evaluate the benefit of opening facilities, and can be solved in linear time [43]. After solving the m subproblems, let $z_i = v(L_t SP^{\lambda})_i$, then the set of facilities to be open can be calculated by solving the following 0-1 knapsack problem [32]:

$$v(L_t SP^{\lambda}) = \text{Min} \sum_{i=1}^{m} z_i y_i - t \sum_{j=1}^{n} \lambda_j$$

subject to:

$$\sum_{i=1}^{m} s_i y_i \geq \sum_{j=1}^{n} d_j$$

$$y_i \in \{0,1\}; i \in M.$$

Let I be the set of indexes for the opened facilities ($y_i = 1$). The allocation variables are then directly derived setting $x_{ij} = 1$ if $(c_{ij} + t\lambda_j) \leq 0$ and $i \in I$ (otherwise $x_{ij} = 0$).

4. Relaxation Heuristics

When the solution obtained by relaxations are transformed on primal feasible solutions, the process involving Lagrangean relaxations are called Lagrangean heuristics. The following relaxation heuristic is used as a base to the algorithms proposed in this work. Both relaxations are used and distinguished by the value of t used in $(L_t SP^{\lambda})$. The usual Lagrangean relaxation uses t = 1 at each iteration of the algorithm and for the Lagrangean/surrogate relaxation, the parameter t = t* results from the search procedure SH at each algorithm iteration.

General Relaxation Heuristic (GRH)

Given $\lambda \geq 0$, $\lambda \neq 0$;
Set lb = $-\infty$;
ub = $+\infty$;

$O = \phi$;

$C = \phi$;

While (not stopping tests) **do**

Solve relaxation $(L_t SP^{\lambda})$ obtaining x^{λ} and $v(L_t SP^{\lambda})$;

Obtain a feasible solution x^f and their value v_f using x^{λ};

lb = max [lb , $v(L_t SP^{\lambda})$];

ub = min [ub, v_f];

Set $g_j^{\lambda} = 1 - \sum_i x_{ij}^{\lambda}$, $j \in N$;

Update the step size θ;

Set $\lambda_j := \lambda_j + \theta\, g_j^{\lambda}$, $j \in N$;

Apply tests in order to possibly fix variables. Update the sets O and C accordingly;

Make stopping tests;

end_while.

where:

a) The initial λ used is such as $\lambda_j := \min_{j \in N} \{ c_{ij} \}$, $i \in M$.

b) The step sizes used are $\theta := \pi\,(ub - lb)/\left\| g^{\lambda} \right\|^2$

c) The sets O and C are defined as $O = \{ i \in M \mid y_i = 1 \}$ and $C = \{ i \in M \mid y_i = 0 \}$. For each λ, a variable y_i $(i \in M)$ is fixed at the value α ($\alpha \in \{0,1\}$) if: $v(L_t SP^{\lambda} \mid y_i = 1 - \alpha) \geq ub$.

d) The stopping tests used are:

- $\pi \leq 0.005$;
- $ub - lb < 0.01$;
- $\left\| g^{\lambda} \right\|^2 = 0$; and
- Every facility was fixed.

For *UFLP* the control of parameter π is the classical control [31]. It makes $0 \leq \pi \leq 2$, beginning with $\pi = 2$ and updating π to $\pi/2$, if after 30 successive iterations the value of ub not increases. For *CFLP* the control of parameter π starts with $2 / t^{*}_{initial}$, where $t^{*}_{initial}$ is the value resulted in the search for approximately solving (D^{λ}_{t}) at the first iteration of GRH. If after 30 successive iterations ub not increases, π is updated to $\pi := \pi/\kappa$, $\kappa \neq 0$. For the usual Lagrangean heuristic $t = t^{*}_{initial} = 1$ and $\kappa = 2$, and for the lagrangean/surrogate heuristic κ is calculated in order to make π less than 0.005 after 6 divisions of π.

To calculate v_f, for each iteration the solution x^{λ} is made feasible conserving the opened facilities and changing the allocations $x^{f}_{ij*} = 1$ if $i \in I$, $j* \in N$ and $c_{ij*} = \min \{ cij \}$. However, when the parameter π is updated, an interchange heuristic is used in order to search for feasible solutions of better quality. In the case of *CFLP*, an additional step is performed, in which a transportation problem that results from problem (P) when the facilities are fixed as open $(y_i = 1)$ or closed $(y_i = 0)$ is solved. The heuristics are best described on Beasley [Beasley, 1993]. For *CFLP*, in order to solve the transportation problems associated with deciding the allocation cost of a set of open facilities we used a specialized network flow algorithm discussed in Marins *et al*. [42].

The parameters used by the search procedure SH are: [s = 0.50; t_0 = 0.0; kmax = 5; n_consec = 3] for *UFLP* instances and [s = 0.35; t_0 = 1.5; kmax = 10; n_consec = 2] for *CFLP* instances.

5. Computational Tests

Algorithm GRH uses the relaxation $(L_t SP^{\lambda})$, i.e., the Lagrangean relaxation (when t = 1, fixed at each iteration) and the Lagrangean/surrogate relaxation (when t = t*, obtained directly by SH or fixed after a number of consecutive applications of SH). Both relaxations are considered in the computer implementation and compared with the same set of instances.

The algorithms were coded in C and run on an IBM Risc6000 3AT workstation (compiled using xlc compiler with -O2 optimization option) for test problems drawn from OR-Library [7] for which the optimal solutions are known. The results are reported in the tables below, where the results for Lagrangean/surrogate heuristic are shown enclosed in brackets. Tables 1 to 3 refer to *UFLP* and tables 4 to 6 refer to *CFLP*. In these tables, all the computer times shown exclude the time needed to setup the problem and the instances are of size n = 1000 and m = 100.

Table 1, for *UFLP*, and table 4, for *CFLP*, show that Lagrangean/surrogate heuristic reaches the same good results of Lagrangean (alone) heuristic. In each table are given:

- The percentage deviation from optimal of upper bound (100*[ub - optimal] / optimal) found by the corresponding heuristic procedure.
- The percentage deviation from optimal of lower bound (100*[optimal - lb] / optimal) found by the corresponding heuristic procedure.

- The number of Lagrangean relaxations solved. It is important to observe that for Lagrangean heuristic (Table 1) the number of Lagrangean relaxations is also the number of subgradient iterations. For Lagrangean/surrogate heuristic (Table 4) however, the number of Lagrangean relaxations solved includes the relaxations solved by the procedure SH discussed in Section 2, and therefore it is greater than the corresponding number of subgradient iterations.
- The total computational time (in seconds).

In order to avoid the effect of spurious stop tests and to see that the Lagrangean/surrogate sequences are more stable and faster than their Lagrangean counterpart, table 2, for *UFLP*, and table 5, for *CFLP*, show the computational time necessary to reach some percentage deviations from optimal of lower bound as found by Lagrangean heuristic and by Lagrangean/surrogate heuristic, for each instance.

Examining the tables it seems reasonable to conclude that Lagrangean/surrogate heuristic is able to approximately solve *UFLP* problems faster than Lagrangean alone heuristic. In order to facilitate this analysis, table 3, for *UFLP*, and 6, for *CFLP*, show the ratio (time for LSH - Lagrangean/surrogate heuristic) / (time for LH - Lagrangean heuristic) and the average ratio for each instance.

Table 1: Computational results for UFLP

Problem number	% deviation from optimal of ub	% deviation from optimal of lb	Lagrangean relaxations solved	Total time
A	- (-)	- (-)	102 (100)	8.24 (6.80)
B	- (-)	- (-)	315 (187)	25.43 (16.63)
C	0.033 (0.033)	0.057 (0.051)	603 (578)	64.17 (64.71)

Table 2: CPU time to reach at least d %
as the deviation from optimal of ub for UFLP

Problem number	d = 95	d = 99
A	5.32 (1.00)	6.17 (2.32)
B	7.08 (1.83)	11.41 (8.70)
C	7.13 (2.52)	19.16 (7.24)

Table 3: Ratios (CPU time for LSH)/(CPU time for LH) for UFLP

	Ratios for percentage deviation times			Ratios for total time	
Problem number	95%	99%	Average ratio	Ratio	Average ratio
A	0.19	0.38		0.83	
B	0.26	0.76		0.65	
C	0.35	0.38	0.39	1.01	0.83

Table 4: Computational results for CFLP

Problem number	% deviation from optimal of ub	% deviation from optimal of lb	Lagrangean relaxations solved	Total time
A-1	2.761 (0.263)	0.065 (0.070)	597 (505)	1484.94 (1023.98)
A-2	3.759 (4.123)	0.052 (0.058)	683 (454)	1463.76 (884.00)
A-3	- (-)	0.208 (0.202)	506 (475)	1135.72 (813.53)
A-4	0.008 (0.008)	- (-)	706 (360)	1042.80 (548.38)
B-1	0.351 (0.351)	- (-)	490 (569)	1752.42 (1300.43)
B-2	4.600 (4.892)	0.300 (0.302)	673 (735)	1753.72 (1396.15)
B-3	3.950 (3.061)	0.340 (0.366)	590 (469)	1600.45 (1035.73)
B-4	2.265 (2.265)	0.109 (0.114)	803 (780)	1750.28 (1311.43)
C-1	0.105 (0.105)	0.161 (0.164)	1136 (993)	2387.25 (1706.54)
C-2	3.566 (0.388)	0.438 (0.451)	670 (655)	1763.89 (1339.78)
C-3	1.340 (2.092)	0.095 (0.338)	709 (614)	1744.86 (1193.15)
C-4	0.027 (0.027)	0.050 (0.050)	508 (601)	1493.86 (1186.83)

Table 5: CPU time to reach at least d % as the
deviation from optimal of ub for CFLP

Problem number	d = 95	d = 99
A-1	302.29 (172.72)	475.51 (361.99)
A-2	304.98 (149.30)	449.11 (202.63)
A-3	309.99 (135.73)	495.94 (266.19)
A-4	195.29 (112.30)	391.21 (133.57)
B-1	406.73 (206.36)	642.16 (285.09)
B-2	344.45 (188.54)	605.51 (473.50)
B-3	326.43 (196.45)	716.63 (403.66)
B-4	334.62 (174.82)	713.83 (324.45)
C-1	371.09 (233.80)	726.70 (352.42)

C-2	367.31 (215.28)	808.29 (464.38)
C-3	327.23 (187.43)	556.78 (399.14)
C-4	328.52 (176.36)	481.65 (228.00)

Table 6: Ratios for CFLP
(CPU time for LSH)/(CPU time for LH)

	Ratios for percentage deviation times			Ratios for total time	
Problem number	95%	99%	Average ratio	Ratio	Average ratio
A-1	0.57	0.76		0.69	
A-2	0.49	0.45		0.60	
A-3	0.44	0.54		0.72	
A-4	0.58	0.34		0.53	
B-1	0.51	0.44		0.74	
B-2	0.55	0.78		0.80	
B-3	0.60	0.56		0.65	
B-4	0.52	0.45		0.75	
C-1	0.63	0.48		0.71	
C-2	0.59	0.57		0.76	
C-3	0.57	0.72		0.68	
C-4	0.54	0.47	0.55	0.79	0.70

Table 3 shows that, for *UFLP*, Lagrangean/surrogate heuristic is able to generate approximate solutions in a computational time that is about 39% of computational time needed to Lagrangean alone heuristic. The Lagrangean/surrogate heuristic seems to be better than the ordinary Lagrangean relaxation for the instances tested.

From table 6 the following conclusion can be drawn, for *CFLP*: the average ratio shows that the gain in computational times for the Lagrangean/surrogate was almost twice for the times associated with the percentage deviations and of 30% for the total times.

Based on these observations it is possible to concluded that the Lagrangean/surrogate version of GRH can generate approximate solutions of *UFLP* and *CFLP* faster than the Lagrangean one when large scale instances is considered. The economy in times for large scale problems can be very important for most applications.

We have made some additional tests looking only for relaxation bounds, fixing the upper bound (*ub* in GRH) to the known optimal solutions. The objective is to compare the effect of primal heuristics on the subradient scheme proposed. The table below shows the computational times (in seconds of a Sun Ultra30 workstation) for CFLP instances when all primal evaluations are rid by inputting the upper bound and the stopping test $\pi < 0.005$ is replaced by a stopping test of the form $\left|v(L_t SP^{\lambda})_k - v(L_t SP^{\lambda})_{k-1}\right| / v(L_t SP^{\lambda})_{k-1} < \varepsilon$ for n consecutive iterations, where $v(L_t SP^{\lambda})_k$ is the value of the dual function at iteration k of the main loop of GRH. Table 7 shows the results for $\varepsilon = 10^{-5}$ and n = 10.

Table 7: Computational results and ratios for CFLP
(Total time for LSH)/(Total time for LH)

Problem number	% deviation from optimal of lb	Total time	Ratio	Average ratio
A-1	0.065 (0.070)	197.10 (135.76)	0.69	
A-2	0.046 (0.050)	190.61 (177.17)	0.93	
A-3	0.207 (0.205)	86.07 (82.10)	0.95	
A-4	- (-)	19.38 (9.65)	0.50	
B-1	0.001 (0.001)	49.89 (55.38)	1.11	
B-2	0.301 (0.327)	271.52 (159.40)	0.59	
B-3	0.336 (0.361)	230.66 (146.10)	0.63	
B-4	0.096 (0.103)	370.89 (328.04)	0.88	
C-1	0.165 (0.170)	198.77 (137.34)	0.69	
C-2	0.425 (0.432)	274.38 (202.45)	0.74	
C-3	0.096 (0.102)	157.55 (148.91)	0.95	
C-4	0.050 (0.050)	179.95 (142.73)	0.79	0.79

Results on table 7 (0.79) are little worst than the ones of table 6 (0.70). The imposed parameter *ub = optimal solution*, generates an external information to the proposed schemes. It can be conjectured here that the Lagrangean/surrogate performs better for small level of information, where the local search SH produces higher corrections.

6. Conclusion

The tables above show that results of high quality are obtained for both heuristics, the Lagrangean (alone) heuristic and the Lagrangean/surrogate heuristic. However, the combination of relaxations in Lagrangean/surrogate heuristic seems to be interesting to improve computational times for a class of instances of the facility location problem.

The local searches in the Lagrangean/surrogate optimization appear to be beneficial to the whole subgradient search. For the same multiplier, different subgradients are used on the Lagrangean and Lagrangean/surrogate. As the Lagrangean/surrogate is also a Lagrangean relaxation, it can be described as a step size corrector of the classical subgradient step size.

The local search also depends on a number of parameters that can influence the performance of the Lagrangean/surrogate heuristic. The adopted naive line search prevents the use of a large number of parameters, and proved to be useful for the instances tested. The local optimization also produces a (different) subgradient that can be used with other subgradient methods, a possible extension of the current approach.

Acknowledgments:

The first author acknowledges Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq (proc. 350034/91-5, 520844/96-3, 680082/95-6) and Fundação para o Amparo a Pesquisa do Estado de S. Paulo - FAPESP (proc. 95/9522-0 e 96/04585-6) for partial financial support.

The second author acknowledges Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq (proc. 350410/94-1) and Fundação para o Amparo a Pesquisa do Estado de S. Paulo - FAPESP (proc. 96/04585-6) for partial financial support.

Comments and suggestions of two anonymous referees were very appreciated.

References

[1] Almiñana, M. and Pastor, T. J. "An adaptation of SH heuristic to the location set covering problem". *European Journal of Operational Research* 100 (1997) 586-593.

[2] Baker, B.M., “Linear relaxations of the capacitated warehouse location problem”, *Journal of the Operational Research Society* 33(1982) 475-479.

[3] Barceló, J.; Fernandez, E. and Jörnsten K.O., “Computational results from a new lagrangean relaxation algorithm for the capacitated plant location problem”, *European Journal of Operational Research* 53 (1991) 38-45.

[4] Bazaraa, M. S., Sherali, H. D. "On the choice of step size in subgradient optimization". *European Journal of Operational Research* 7 (1981) 380-388.

[5] Beasley, J.E., “An algorithm for solving large capacitated warehouse location problems”, *European Journal of Operational Research* 33 (1988) 314-325.

[6] Beasley, J.E., “Lagrangean heuristics for location problems”, *European Journal of Operational Research* 65 (1993) 383-399.

[7] Beasley, J.E., “OR-Library: Distributing test problems by electronic mail”, *Journal of the Operational Research Society* 41 (1990) 1069-1072.

[8] Brännlund, U., “A Generalized subgradient method with relaxation step”, *Mathematical Programming* 71 (1995) 207-219.

[9] Camerini, P. ; Fratta, L. and Maffioli F., On improving relaxation methods by modified gradient techniques” *Mathematical Programming Study* 3 (1975) 26-34.

[10] Christofides, N. and Beasley J.E., “Extensions to a lagrangean relaxation approach for the capacitated warehouse location problem”, *European Journal of Operational Research* 12 (1983) 19-28.

[11] Cornuéjols, G.; Fisher, M. and Nemhauser, G. L., “Location of banks account to optimise float: an analytic study of exact and approximate algorithm”, *Management Science* 23 (1977) 789-810.

[12] Cornuéjols, G.; Sridharan, R. and Thizy, J.M., “A comparison of heuristics and relaxations for the capacitated plant location problem”, *European Journal of Operational Research* 50 (1991) 280-297.

[13] Correa, R. and Lemaréchal, C, Convergence of some algorithms for convex minimization”, *Mathematical Programming* 62 (1993) 261-275.

[14] Domschke, W. and Drexl, A., “Location and layout planning: An international bibliography”, Volume 238 in: *e Notes in Economics and Mathematical Systems* , Springer-Verlag, Berlin, 1985.

[15] Dyer, M. E. "Calculating surrogate constraints". *Mathematical Programming* 19 (1980) 255-278.

[16] Erlenkotter, D., “A dual-based procedure for uncapacitated facility location”, *Operations Research* 26 (1978) 992-1009.

[17] Ermol'ev, Y. M. , "Methods for solving nonlinear extremal problems", *Cybernetics* 16/1 (1966) 1-14.

[18] Fisher, M. L. "The lagrangian relaxation method of solving integer programming problems". *Management Science* 27(1981) 1-18.

[19] Freville, A.; Lorena, L.A.N. and Plateau, G., "New subgradient algorithms for the 0-1 multiknapsack Lagrangian and surrogate duals", *Pre-publication # 90. LIPN-Universite Paris Nord* (1990).

[20] Galvão, R.D. and Raggi, L.A., "A method for solving to optimality uncapacitated location problems" *Annals of Operations Research* 18 (1989) 225-244.

[21] Garey, M.R. and Johnson, D.S., "Computers and intractability: a guide to the theory of NP-completeness", W.H. Freeman and Co., San Francisco, (1979).

[22] Geoffrion, A., "Lagrangean relaxation and its uses in integer programming", *Mathematical Programming Study* 2 (1974) 82-114.

[23] Glover, F., "Surrogate constraints", *Operations Research* 16/4 (1968) 741-749.

[24] Glover, F. "Surrogate Constraints Duality in Mathematical Programming". *Operations Research* 23 (1975) 434-451.

[25] Goffin, J. L. "On convergence rates of subgradient optimization methods", *Mathematical Programming* 13 (1977) 329-347.

[26] Greenberg, H. J., Pierskalla, W. P. "Surrogate Mathematical Programming". *Operations Research* 18 (1970) 924-939.

[27] Guinard, M., "A Lagrangean dual ascent algorithm for the simple plant location problem", *European Journal of Operational Research* 35 (1988) 193-200.

[28] Handler, G. and Zang, I., "A dual algorithm for the constrained shortest path problem", *Networks* 10 (1980), 293-310.

[29] Held, M., Karp, R. M. " The Traveling salesman problem and minimum spanning trees". *Operations Research* 18 (1970) 1138-1162.

[30] Held, M., Karp, R. M. " The Traveling salesman problem and minimum spanning trees: Part II". *Mathematical Programming* 1(1971) 6-25.

[31] Held. M., Wolfe, P., Crowder, H. P. "Validation of subgradient optimization", *Mathematical Programming* 6 (1974) 62-88.

[32] Horowitz, E. and Sahni, S. "Computing partitions with applications to the knapsack problem", *Journal of the ACM* 21 (1974), 277-292.

[33] Jacobsen, S.K., "Heuristics for the capacitated plant location model", *European Journal of Operational Research* 12 (1983) 253-261.

[34] Karwan, M. H., Rardin, R. L. "Some relationships between lagrangian and surrogate duality in integer programming". *Mathematical Programming* 17 (1979) 320-334.

[35] Kim, S. and Um, B. S., "Polyak's subgradient method with simplified projection for nondifferentiable optimization with linear constraints", *Optimization* 20 (1989) 451-456.

[36] Kim, S. and Um, B. S., "An improved subgradient method for constrained nondifferentiable optimization", *Operations Research Letters* 14 (1993) 61-64.

[37] Korkel, M., "On the exact solution of large-scale simple plant location problems", *European Journal of Operational Research* 39 (1989) 157-173.

[38] Krarup, J. and Pruzan, P.M., "The simple plant location problem: Survey and synthesis", *European Journal of Operational Research* 12 (1983) 36-81.

[39] Larsson, T. ; Patriksson, M. and Strömberg, A-B., "Conditional subgradient optimization - theory and applications", *European Journal of Operational Research* 88 (1996) 382-403.

[40] Lorena, L.A.N. and Lopes, F.B., "A surrogate heuristic for set covering problems", *European Journal of Operational Research* 79/1(1994) 138-150.

[41] Lorena, L.A.N. and Narciso, M.G. "Relaxation heuristics for a generalized assignment problem" *European Journal of Operational Research* 91 (1996) 600-610.

[42] Marins, F.A.S.; Senne, E.L.F.; Darby-Dowman, K.; Machado, A.F.; Perin, C. "Algorithms for network piecewise-linear programs: A comparative study", *European Journal of Operational Research* 97 (1997) 183-199.

[43] Martello S. and Toth, P., "Knapsack problems: Algorithms and computer implementations" , John Wiley & Sons, New York, 1990.

[44] Minoux, M., "Plus courts chemins avec constraints: Algorithmes et applications", *Annals of Telecommunications* 30 (1975), 383-394.

[45] Nauss, R.M., "An improved algorithm for the capacitated facility location problem", *Journal of the Operational Research Society* 29 (1978) 1195-1201.

[46] Parker, R. G., Rardin, L. R. "Discrete Optimization". Academic Press, Inc., London, 1988.

[47] Polyak, B. T, "Minimization of unsmooth functionals", *USSR Computational Mathematics and Mathematical Physics* 9 (1969) 14-29.

[48] Senne, E.L.F. and Lorena, L.A.N **"**A lagrangean/surrogate approach to facility location problems", EURO-TIMS Congress - Barcelona (1997).