CEEApIA
Centro de Estudos de Economia Aplicada do Atlântico

WORKING PAPER SERIES

CEEApIA WP No. 09/2013

A regret model applied to the maximum coverage location problem with queue discipline

Pedro Nunes
Francisco Silva
Helena Ramalhinho Lourenço

December 2013

FCT Fundação para a Ciência e a Tecnologia
MINISTÉRIO DA EDUCAÇÃO E CIÊNCIA

Universidade dos Açores
Universidade da Madeira

A regret model applied to the maximum coverage location problem with queue discipline

Pedro Nunes
Universidade dos Açores (DEG e CEEAplA)

Francisco Silva
Universidade dos Açores (DEG e CEEAplA)

Helena Ramalhinho Lourenço
UPF- Barcelona GSE

Working Paper n.º 09/2013
dezembro de 2013

CEEAplA Working Paper n.º 09/2013
dezembro de 2013

RESUMO/ABSTRACT

A regret model applied to the maximum coverage location problem with queue discipline

This article discusses issues related to the location and allocation problems where is intended to demonstrate, through the random number generation, the influence of congestion of such systems in the final solutions. It is presented an algorithm that, in addition to the GRASP, incorporates the Regret with the p-minmax method to evaluate the heuristic solution obtained in regard to its robustness for different scenarios. To the well know Maximum Coverage Location Problem from Church and Revelle [1] an alternative perspective is added in which the choice behavior of the server does not only depend on the elapsed time from the demand point looking to the center, but also includes the waiting time for service conditioned by a waiting queue.

Key-words: location, allocation, coverage, heuristic, regret, queue, scenarios.

Pedro Nunes
Universidade dos Açores
Departamento de Economia e Gestão
Rua da Mãe de Deus, 58
9501-801 Ponta Delgada

Francisco Silva
Universidade dos Açores
Departamento de Economia e Gestão
Rua da Mãe de Deus, 58
9501-801 Ponta Delgada

Helena Ramalhinho Lourenço
Universitat Pompeu Fabra
Barcelona Graduate School of Economics
Ramon Trias Fargas, 25-27
08005 Barcelona
Catalonia-Spain

A regret model applied to the maximum coverage location problem with queue discipline

Pedro Nunes

Economics, University of the Azores

Francisco Silva

Economics and Business, University of the Azores

Helena Ramalhinho Lourenco

UPF- Barcelona GSE

Abstract

This article discusses issues related to the location and allocation problems where is intended to demonstrate, through the random number generation, the influence of congestion of such systems in the final solutions. It is presented an algorithm that, in addition to the GRASP, incorporates the Regret with the *p-minmax* method to evaluate the heuristic solution obtained in regard to its robustness for different scenarios. To the well know Maximum Coverage Location Problem from Church and Revelle [1] an alternative perspective is added in which the choice behavior of the server does not only depend on the elapsed time from the demand point looking to the center, but also includes the waiting time for service conditioned by a waiting queue.

Key-words: *location, allocation, coverage, heuristic, regret, queue, scenarios*.

1. Introduction

Given the increasing relevance concerning the provision of services compared to existing demand and the costs of setting up such a system, the location problems are of utmost importance both in our daily lives as well as in scientific circles. Typically, the performance of such services is evaluated by the number of customers in the queue and the waiting time expect since the arrival at the center. Overall, what can be concluded is that these indicators are highly correlated with the number of centers providing services and their specific location. Examples of such services are medical systems, police operations, firefighters, roadside assistance services, among others.

The problem that researchers intend to resolve is related to the location of service centers and the respective allocation of demand to these centers. In an attempt to capture the characteristics of these systems and make an approximation to reality, the location models have become so complex that obtaining results by complete enumeration became more difficult, essentially as a result of the exponential growth of computing time.

The various models developed have in common that the inherent complexity hampers the process of finding a solution. Thus, the formulations were constrained by simplifying assumptions, sometimes occurring deviations from the reality faced by planners, whether in the public or private sector. Technological advances have allowed a gradual development of more realistic formulations, given the possibility of finding a solution to complex models with acceptable computing times.

Despite the different proposed formulations over time for the various problems related to location and allocation, it is common that the regions are represented by networks, a continuous space or a set of discrete points.

This paper presents an algorithm that, besides the known GRASP - Greedy Randomized Adaptive Search Procedure incorporates the p- *minmax* Regret method to evaluate the heuristic solution obtained in regard to its robustness for different scenarios. The use of these processes is in line with previous researches, like the work of Daskin et al [2], and aims at their integration in order to explore new methodologies that improve or better adapt to the circumstances of the cases studied.

By varying the limits imposed in terms of waiting times, maximum distance, demand processing capacity and size of the network, it is possible to notice significant changes in the final solutions. The problem under study is the well-known Maximum Cover Location Problem developed by Church and Revelle [1] incorporating an alternative model in which the server choice behavior does not only depend on the elapsed time from the node to the center, but also includes the waiting time for the service.

The model and its various examples were tested using the random number generation. In most cases different results show up making it possible to confirm that the proposed formulation produces significant differences in the results. In general, the more "tighter" the systems are, ie, when the related distance limit between demand and center values are lower, the number of service centers to locate or the waiting time limit, location decisions are more sensitive to pre-defined parameters for the model.

As regards the classification of the location problem in focus, this may be done according to the grouping suggested by Current et al. [3]: maximum distance models, "p-dispersion" problems and average or overall distance models. Thus is characterized by being a maximum distance model when is explicitly considered a maximum distance within which a facility must be located to provide service as well as a time limit within which the service should be provided. This is usually the case of schools, hospitals or police stations location where people expect to have an available facility within acceptable limits away from the original demand area.

A consumer is considered to be covered by a server – which is considered fixed and presents unlimited capacity concerning the possibility of processing the service needs - if you have an installation within the pre-established distance limit. Where a service is provided by a facility located below this maximum, then the service is considered appropriate or acceptable.

The formulation used in this work is inherited from previous studies, such as the works of Marianov and Serra [4] and the one of Silva and Serra [5], mainly in relation to the "Maximum Coverage Models" and the additional incorporation of results from the Queuing Theory.

The complexity associated with the model, assumed sophistication in an attempt to capture more elements of reality under study, requires the use of heuristic procedures in the search for solutions. Thus, in addition to GRASP, the algorithm contains a Regret component, based on the work of Daskin [2], which demonstrates to produce acceptable results both in terms of computing speed as, and perhaps more importantly, in terms of approximation to the optimal solution.

This work aims to highlight the importance, as verified in real life decisions, of considering the congestion of the systems in its various forms as a determining factor of location and allocation decisions.

2. Related Literature

Location Models have been studied for some decades now, with submitted proposals that fit both the public and private sectors. Classic examples of such problems are the ones that were carried out and pioneered by Hakimi [6, 7]. This author considered a network without imposed direction on arcs, where consumers are located only at the nodes (or demand point). Each demand point presents a certain percentage of demand or need for care. The 1-median problem explored is based on the location of a facility on a network with the goal of serving the consumer minimizing the average travel distance between the facility and demand node. Hakimi showed that, among all the nodes, there was at least an optimal location for the server, this way reducing a continuous search by a finite one. A similar result is also applied to (p-median) multi-median problem, in which various facilities should be located so as to minimize the average distance from the nearest facility to the consumers.

These types of problems addressed, Discrete Location Models, are generally formulated with integer linear programming and can be solved using the algorithm known as "*branch and bound*". However, somehow justifying the use of heuristic processes, even the most basic location problem is classified as "NP-Hard" and requires an unacceptable computation time to find patterns of employment associated with more realistic situations.

In this category, the maximum coverage model of Church and Revelle [1], pretends to limit the number of facilities to be located. The goal becomes to locate a predetermined or budgeted number of facilities so that the requests affected to a certain server installation are maximized. This model does not require that all demand is covered.

However, it has already been studied for some time within the problems of location, the uncertainty that such systems may exhibit leads researchers to assume alternative positions in regard to demand modeling, travel time or even costs associated with the chosen locations. Four basic approaches were formulated: approximations through a deterministic replacement, finding deterministic equivalent, probabilistic models with restraints, queuing systems spatially distributed and scenario planning.

Regarding this approach with the inclusion of scenarios, such as developed in the present work, we highlight the introduction to the topic held by Sheppard [8] in 1974. This shows the importance of making location and allocation decisions taking account environments which are uncertain as it is these that dominate our object of study. This way we take into account the randomness in the frequency of demand for certain service as a perspective on uncertainty, the source of congestion of the systems in question.

Still under a deterministic point of view, others are addressing congestion with alternative formulations as models for the location and coverage with redundant coverage, like Hogan and Revelle [9, 10] suggest. Daskin [11, 12] suggests location models with probability in one of its known extensions to the problem "Maximal Covering" assuming that servers are busy according to a given probability. In this case the goal is to maximize the demand covered by the other servers that are not occupied.

The study area that intends to incorporate interactions occurring in queues with location models shows some interesting advances that have been made in recent decades in particular with respect to emergency services. The pioneering work regarding this topic was Larson´s [13] – "*hypercube queuing*" model, which already considers a system of stochastic service spatially distributed, but also with mobile servers scattered. Batta et al [14] used this model to demonstrate that the implicit assumption of independence of the server, as assumed by Daskin [11],[12], is often violated.

To consider, as in the present work, fixed servers it is also possible to verify that they can "face" congestion. This is the case of health services and, in general, public services of any kind availing servers fixed. See it as an example Marianov and Serra [15].

Berman, Larson and Chiu [16] effected the work considered the beginning of the "marriage" between the location theories and queuing theories. These expanded the Hakimi´s [6] 1-median problem incorporating it in the context of queues. In deciding the location of service centers is explicit their dependence in relation to service times, travel times and delays arising from queues themselves. As a basis for this work is also the Larson´s [13] Hypercube Queuing Model, opportunely mentioned.

In this context, many are the objectives proposed. Batta [17] considers the problem of locating a single service center in a network that operates as an M/G/1 queue where waiting calls are answered according to a class of queuing disciplines that rely solely on the information about the expected service time.

Brandeau and Chiu [18], also dedicated to congested systems, develop the model "*Stochastic Queue Center Location*" that aim to minimize the maximum response time to any consumer. For these authors the expected response time takes into account not only the waiting time until the server is on but also the travel time to the contact center.

In turn, the Revelle and Hogan [9] model opportunely mentioned, deal with congestion presenting a probabilistic version of the location problem. It is also in this line of development, that Marianov and ReVelle [19] present the “Probabilistic Covering Problem with Queues”. With an identical formulation as the one followed by in the “Probabilistic Covering Problem”, a simple modification of Maximum Coverage Location Problem of Church and ReVelle [1], Marianov and Serra [4] introduce the "Maximum Coverage Location-Allocation Model with Queues" where the goal is to locate *p* service centers and affect these users so that the maximum population is covered.

3. The Maximum Coverage Problem with Queues

This section seeks to present, as regards employed notation (3.1) and proposed formulation (3.2), the Maximum Coverage Problem studied to which a heuristic is applied and measured to assure the respective quality.

3.1. Notation

The following notation is defined as:

Sets:

- I is the set of all demand nodes, indexed with i;
- J is the set of all potential location nodes, indexed with j;
- N_i is the set of centers located at a distance lower to or equal to a distance lower to or equal the distance limit (*ldist*) from the demand node *i*.

Parameters:

- W_j is the average waiting time at center *j* (see equation 3.1.1), where $\rho_j = \frac{\lambda_j}{\mu}$ and $\lambda_j = \sum_i a_i X_{ij}$;

- τ is the parameter that represents the limits imposed for the waiting times;
- p is the number of centers to be located;
- a_i is the population in the demand point i;
- d_{ij} is the distance from the center i looking to the service center located in j;
- λ_i is the rate that represents the process of service calls for each node i according to a Poisson process;
- λ_j is the rate defined as the sum of the rates of calls from all demand points affected to a service center in j;
- S is the average service time according to the distribution function;
- μ is the average rate service per time unit;
- f_i is the demand call rate for service on demand point i;
- $\rho_j = \overline{S_j}\lambda_j = \overline{S_j}\sum_i f_i X_{ij}$ is the utilization factor defined as the product of the average service time and arrival rate;
- ns represents the number of scenarios for the problem development.

<u>Decision Variables</u>:

- X_{ij} is 1 if demand node i is affected to a center and 0 otherwise;
- Y_j is 1 if a center is located at j and 0 otherwise.

According to the Queues Theory, taking into account the work of Kleinrock [20], the waiting time and the occupation factor are obtained, respectively, using the expressions

$$W = \frac{\rho}{\mu(1-\rho)} \qquad 3.1.1$$

$$\rho = \frac{\lambda}{\mu} = S\lambda \qquad 3.1.2$$

where μ represents the service rate.

3.2. Formulation

The Location Coverage Problem with Queues assumes static allocation of customers to service centers. This is a typical assumption in the case of servers with fixed locations where consumers move to the center for treatment [21].

We will assume a Direct Choice Environment in which a central decision maker sets the allocation of a client to a center:

$$^{Max}Z = \sum_i \sum_j a_i X_{ij}$$

s. t.

$$X_{ij} \leq Y_j \qquad \forall \quad i \in I, \forall j \in N_i \qquad (3.2.1)$$

$$\sum_{j \in N_i} X_{ij} \leq 1 \qquad \forall \quad i \in I \qquad (3.2.2)$$

$$\sum_j Y_j = p \qquad (3.2.3)$$

$$W_j \leq \tau \qquad \forall \quad j \qquad (3.2.4)$$

$$X_{ij};\ Y_j \in \{0,1\} \qquad \forall \quad i \in I, \forall j \in N_i$$

$$N_i = \{j | d_{ij} \leq ldist\} \qquad (3.2.5)$$

$$W_j = \frac{\rho_j}{\mu(1-\rho_j)} \qquad (3.2.6)$$

The objective function maximizes the population coverage. Restriction (3.2.1) states that if the population *i* is allocated to a center *j*, then there must be a center in *j*; (3.2.2) requires that each search point is not to affect more than one center; (3.2.3) defines the number of centers to be located; (3.2.4) forces that the average waiting time is less than a pre defined limit τ. Additionally, *j* necessarily has to be in the range N_i. The same is to say that if demand point is covered, then there should be a center located within a distance limit *ldist*.

4. Heuristic Procedure

It will be shown now, as defined by Reeves [22], a heuristic method that seeks to find good solutions (ie near-optimal) in reasonable computing time. Therefore it is assumed that the solutions found by these heuristics methods are not always able to ensure an optimal and possibly may not present possible solutions.

The implemented algorithm uses, at some point, the "Greedy Randomized Adaptive Search Procedure" (GRASP) developed by Feo and Resende [23] which comes up as an unfolding of one of the heuristics used to solve the 1st location models designed by Teitz and Bart [24]. This is seen as an approach through "exchange" or "replacement" moving the servers from their current positions to other non used positions and keeping this new position whenever the objective value is improved. For additional information on the subject see Festa andResende [25].

It is also used in the proposed algorithm the heuristic p-*minmax* Regret developed by Daskin et al [2]. To understand the importance of this method, before we take into account the issue addressed, in an attempt to deal with the unpredictability of demand, the problem is solved for different scenarios by random generation (where these represent different population levels) and different frequencies of demand for service.

The term Regret is thus associated with the notion of deviation or difference thus giving us an "opportunity cost" when implementing the chosen location. Given the design of the service delivery system that we considered optimal for a given scenario, the Regret, based on population characteristics of the remaining scenarios and respective objective function value (covered population), returns the difference in case the locations were kept.

According to the author, many may be the objectives for this indicator. This work intends to, from the largest differences select the lowest and, by successive adaptations, minimize it. We can interpret this process as finding a location solution that, in the worst possible scenario (the one that presents the greatest Regret given the optimal solution) presents the lower deviation or "opportunity cost".

We now presented the algorithm implemented in C + + in which is introduce the two heuristics discussed above - GRASP and *p-minmax* Regret. Before that, we indicate the notation used:

- j index of possible locations;
- i index of demand nodes;
- D_j list of potential location points for services ordered according to the total population;
- S solution
- $\overline{S}$ complementary solution
- C candidate set of points
- p number of services to locate
- n number of demand nodes
- *inc_j* overall rate of calls to potential service location j
- D_{ij} list of demand nodes within the distance limit counting from the potential location for service j

Thus, the system starts by reading the distance file recognizing at this stage the network of demand points - all of them potential locations – and the distances between them - our associated cost measured in terms of time units.

For each node representative of a population center, the respective population is generated according to a Uniform distribution. Since this is a user-defined parameter, in order to subject the model to different conditions, the demand is then estimated based on a percentage of the population previously obtained. These values – population and demand- are generated for each one of the (ns) scenarios and will be as many as the number of scenarios with which we intend to work.

In possession of the characterization of the network, as regards the number of nodes and distance between these and respective populations and demand, using the CPLEX optimization software, the model begins by solving the maximum coverage location problem for each scenario.

The optimal value of our objective function for each scenario is thus obtained, acting as a future reference when compared to the results obtained for other levels of population and when maintaining the optimum locations obtained. This solution includes: in which nodes are located service centers, the allocation of demand points to the respective centers and the value for the objective function concerning the covered population.

Having simulated the scenarios, a matrix titled Regret 1 (ns, ns) is built. The diagonal of the matrices Regret 1 indicates the optimal values obtained in each of the simulated scenarios. Values outside this diagonal, ie, adjacent to the optimal solution values, are obtained by calculating the objective function considering the characteristics of the other scenarios (populations and demands) but maintaining the location-allocation patterns given to us by the CPLEX optimization software.

197182	201432	198357	196556	195265	194325	198562	196663	197701	194003
161050	190007	181250	182356	192123	179125	183459	168971	179157	189457
170892	167845	157853	164887	169741	190187	178112	156746	192454	189451
171313	175949	194848	164073	169787	199454	185464	177989	184188	188774
169787	191717	188772	169745	180869	183556	197010	191141	186311	188745
196787	192776	193741	183797	181579	171896	181235	182454	192478	179878
197121	195798	177131	193457	189743	172656	164681	165888	178432	182747
177336	181656	189743	187141	191778	194331	184556	175624	185655	189774
179487	173998	167131	189477	183454	199466	193473	195471	164635	167979
189741	185731	159887	169741	200874	190157	182486	189478	168635	182916

Figure 1. 10 scenarios Regret 1 matrix.

A caution since the calculation of the values adjacent to the "optimality diagonal" requires a possibility test so that, in the continuation of the algorithm, a necessarily workable initial solution is obtained. The possibility test of a value adjacent to the optimal solution should take into account:

i. The distance limit (ldist) from demand node *i* allocated to service center in *j* é is taken into account;
ii. The limit for the waiting time (wlim) in *j* is respected; and
iii. Whenever negative valued waiting times are associated to a solution, in the matrix Regret 1 a zero will be shown.

When the objective value calculated with the standard locations and allocations for the other scenarios is not possible, under the conditions of possibility described above, the value in the Regret 1 matrix will be zero. Thus the associated solution is omitted of the rest of the process as an initial starting solution for GRASP since this zero value will not be considered in the choice of maximum deviation that is performed in the *p-minmax* process.

After the possibility test, based on Regret 1 matrix, the Regret 2 and Regret 3 matrixes are build up, where each one being a transformation of the one preceding, as described below:

Regret 2: each value of this matrix will be obtained by the difference between the goal of a given scenario and its respective optimal (reciprocated by CPLEX and contained in the diagonal of the Regret 1 matrix). This way, imperatively the diagonal of matrix of Regret 2 will contain only zeros. This procedure allows us thus obtain the values of an "Absolute Regret".

0	4250	1175	-626	-1917	-2857	1380	-519	519	-3179
-28957	0	-8757	-7651	2116	-10882	-6548	-21036	-10850	-550
13039	9992	0	7034	11888	32334	20259	-1107	34601	31598
7240	11876	30775	0	5714	35381	21391	13916	20115	24701
-11082	10848	7903	-11124	0	2687	16141	10272	5442	7876
24891	20880	21845	11901	9683	0	9339	10558	20582	7982
32440	31117	12450	28776	25062	7975	0	1207	13751	18066
1712	6032	14119	11517	16154	18707	8932	0	10031	14150
14852	9363	2496	24842	18819	34831	28838	30836	0	3344
6825	2815	-23029	-13175	17958	7241	-430	6562	-14281	0

Figure 2. 10 scenarios Regret 2 matrix

Regret 3: the difference obtained according to calculations made in Regret 2 matrix is now divided by the optimal value of reference for the scenario in question. So we get to know the value of "Relative Regret" associated to each scenario.

0,00000	0,02155	0,00596	0,00317	0,00972	0,01449	0,00700	0,00263	0,00263	0,01612
0,15240	0,00000	0,04609	0,04027	0,01114	0,05727	0,03446	0,11071	0,05710	0,00289
0,08260	0,06330	0,00000	0,04456	0,07531	0,20484	0,12834	0,00701	0,21920	0,20017
0,04413	0,07238	0,18757	0,00000	0,03483	0,21564	0,13037	0,08482	0,12260	0,15055
0,06127	0,05998	0,04369	0,06150	0,00000	0,01486	0,08924	0,05679	0,03009	0,04355
0,14480	0,12147	0,12708	0,06923	0,05633	0,00000	0,05433	0,06142	0,11974	0,04644
0,19699	0,18895	0,07560	0,17474	0,15219	0,04843	0,00000	0,00733	0,08350	0,10970
0,00975	0,03435	0,08039	0,06558	0,09198	0,10652	0,05086	0,00000	0,05712	0,08057
0,09021	0,05687	0,01516	0,15089	0,11431	0,21156	0,17516	0,18730	0,00000	0,02031
0,03731	0,01539	0,12590	0,07203	0,09818	0,03959	0,00235	0,03587	0,07807	0,00000

Figure 3. 10 scenarios Regret 3 matrix.

It is precisely based on Regret 3 matrix that we continue our algorithm applying the *p-minmax* heuristic as suggested by Daskin et al. [2]. Before the latter matrix, the procedures are as follows:

i. From the Regret 3 matrix, observing the values in line, we choose the one that presents the higher relative regret, in other words, the possible solution that departs furthest in percentage terms from the optimum control solution contained in Regret 1 diagonal obtained using the CPLEX;

ii. Subsequently, all these maximum percentage deviations (relative Regrets) the minor is picked up with the intention of using it an initial solution in the local search that follows on the GRASP heuristic.

The GRASP consists of two phases – construction phase and local search phase – and is an iterative process with reliable solution built independently at each iteration. Described below is a pseudo-code for the GRASP.

Procedure GRASP (Max_iterations, Seed)
  For **k = 1** to **Max_iterations** do
    S←Greedy_Randomized_Construction(Seed, γ);
    S←Local_Search(Solution);
    Update_Solution(Solution, Best_Solution)
  enddo
  end GRASP

Pseudo-Code 1: Pseudo-Code GRASP

From a general point of view, the process developed in this heuristic, after the selected initial solution as described above, follows like this:

1. From the locations contained in the initial solution previously obtained, randomly one is chosen to be removed and replaced by another which, necessarily, must be on the RCL – *Restricted Candidate List*;
2. The potential locations belonging to the RCL must meet the requirements of acceptability with regard, not only the distance limits imposed but presenting a priori a demand frequency greater than or equal to *gamma* percent of the search node with the highest demand frequency;
3. If part of the RCL it is temporarily accepted to be considered in the iterative process and, when replacing the previous location, switches its position regarding its allocation to demand points;
4. When the initial solution is improved, this new locations and allocations pattern is accepted;
5. Otherwise, the initial solution remains.

Are then built again the Regret1, Regret 2 and Regret 3 matrixes based on the values obtained in the local search now held. In this process, the first matrix goes again through the possibility test once described. This process is repeated for a predefined number of iterations.

Now it is explained in detail, with the use of pseudo-code for the two phases of that process, the GRASP heuristic. The construction phase, which will return an initial solution at each iteration, is called $Greedy\_Randomized\_Construction(Seed, \gamma)$ and is a function from the root in the random number generator and of the gamma parameter that defines what solutions will be included in RCL - *Restricted Candidate List*, the list containing the best solutions.

The development of the $Greedy\_Randomized\_Construction(Seed)$ is now described:

procedure *Greedy Randomized Construction* (Seed,γ)
{sort candidate sites by decreasing order of population}

```
D_j ← Sort_Candidate_Sites( population );
{initialize solution set}
S := { };
S̄ := C;
{while solution is not a complete solution}
while |S| ≠ p do
        {loop over all candidate sites not in the solution list}
For j=1 to |S̄| do
                {initialize parameters}
                {restrict demand points list to the standard covering distance to site j}
D_ij ← {i ∈ D, d_ij ≤ d}
                {sort demand points by increasing distance to site j}
D_ij ← Sort_Demand_Points(distance);

                {loop over demand points in set Dij}
For i=1 to |D_ij| do
                        {sum frequencies at each demand point if waiting time limit is not reached}
If (W_j < τ and ρ_j<1) do
inc_j := inc_j+ f_i ;
actualize w_j;
actualize ρ_j;
Endif
Enddo

{construct the restricted candidate list}
c^max := max{inc_j};
RCL ← {j ∈ S̄, inc_j ≥ γ c^max};
        {select randomly one site from the RCL}
j* ← Random_Select(RCL);
S := S ∪ {j*};
S̄ := S̄ \ {j*};
{take the demand points allocated to j* out of the demand points list}
For i=1 to |D_ij*| do
                D := D\{i ∈ D_ij*};
Enddo
Enddo
end Greedy Randomized Construction
```

Pseudo-Code 2: Construction Phase Pseudo-Code

The proposed algorithm starts by choosing the candidate nodes according to their respective demands/populations**.** We considered in our example that all demand nodes are also potential service location points. Another possibility would only consider a subset of demand nodes from the $D_{j.}$ list.

This way, starting with the first node from the candidate list, it is affected to it the closest demand nodes until de coverage limit is attained. Here, the coverage limit can be seen whether by the utilization coefficient or the imposed limit for the waiting time.

Total demand affected to each of the potential sites *j* is called *incoming call rate*. The *incoming call rate* works as a "*greedy*" function of the algorithm and can be defined as a weighing of the demand nodes no yet covered but that will do if location *j* was chosen to have a server facility.

It is included on the RCL - Restricted Candidate List (sub-set of best solutions) the candidate nodes with a total *incoming call rate* greater to or equal to *gamma* per cent of the *incoming call rate* indexed to the potential location with higher value.

In the GRASP, the *gamma* parameter is established before. (for instance, if *gamma* equals 0.8, we therefore mean that we include on the list containing the best solutions - *Restricted Candidate List* – all the potential locations with a total incoming call rate greater then 80% of the highest value between all incoming call rates).

Note that in the "*greedy*" heuristic, as suggested by Marianov and Serra [4] , the choice would always be to locate a center at the node with the highest sum of incoming call rates, i.e., γ=1.

At each iteration, we choose randomly from among the candidate locations with the highest incoming call rate (i.e., the ones included on *Restricted Candidate List*) the *p* locations for servers.

procedure *Local_Search* (Solution, Best_Solution)
    obj_best : = *obj(S)*;
        {loop over sites in the solution}

    for all $j_1 \in S$ **do**
    $S := S \setminus \{j_1\};$
        {loop over sites not in the solution}

```
for all  j_2 ∈ S̄ do
        evaluate obj(S ∪ {j_2});
        if obj_best< obj(S ∪ {j_2}) do
                S := S ∪ {j_2}:
                obj_best := obj(S ∪ {j_2});
          else
                S := S ∪ {j_1}:
          endif
          enddo
Enddo
end Local_Search
```

Pseudo-Code 3: Local Search Phase Pseudo-Code

At the local search phase, for each center *per se*, we un-allocate its assigned demand and move it to all the potential locations not yet used, repeating at each time the steps 9 to 20 from the *Greedy Randomized Construction* procedure, aiming to evaluate the objective. If any of the locations reciprocate a better objective value, we maintain the service center at that node; otherwise, we keep it in the original location (see Pseudo-Code 3). We repeat the procedure until it is not possible to improve the initial solution or the limit of iterations is reached.

In a user defined environment, the algorithm would need to be modified, both in construction phase and in the local search, in order to strengthen a closest allocation. The proposed algorithm penalizes the final objective whenever an unreliable solution is obtained. In the case of obtaining a reliable solution, this set of locations is considered as potential site for the placement of service centers. Otherwise, we consider this set of locations only as an initial solution and not as a potential service location penalizing the objective with a large negative value M. This will match the following objective evaluation procedure:

```
procedure evaluate_objective (S)

    Allocate each demand point to its closest center location;
    Evaluate W_j and ρ_j;
    obj(S):=0;
    If (W_j < τ and ρ_j<1) do
```

```
            For j=1 to p do
                For i=1 to n do
                    If (i is allocated to j) do
                        obj(S):=obj(S) + f_i;
                    endif;
                enddo;
            enddo;

Else
        obj(S):=M;
end evaluate_objective;
```

Pseudo-Codoo 4: Objective Evaluation Pseudo-Code

During the Local Search phase, for each center at a time, we un-allocate their assigned demands and mode them to all the un-used potential locations. We always affect a demand node to the nearest potential location e check the possibility observing the waiting time limit. If the solution is not possible, the objective is penalized with a very high negative value M. Whenever new allocations result in a better objective we maintain that center in that location. Otherwise, the starting location is kept. This procedure is repeated until, when comparing with the previous, no better solution is found.

5. Computational Experience

In order to observe the difference between the results of the heuristic solution and the initial results obtained, which will serve as a starting point (and comparison) for the GRASP Local Search, were randomly generated problematic situations in the demand network model proposed. The size of this network will be variable and each center is assigned a particular demand frequency (need for service / care). Furthermore, the characteristics of this will also be amended concerning the number of nodes and demand centers.

Were also set alternative values for both Waiting Time limit as to Distance limit, as can be seen in Table 5.1. As regards the recursive process of the algorithm, programs were tested sometimes for different number of iterations and for different number of scenarios.

In each generated scenario and for each specific network, the distance between nodes is constant since it is only changed the size of the network regarding the number of nodes and the demand recorded at each respective point. The distance between demand points is achieved by using a distances matrix common to all scenarios and studied networks.

A summary of the characteristics and parameters of the worked data are presented below in Table 5.1.

	Number		**Limits**		**Number**	
Cases	**Nodes**	**Centers**	**Distance**	**Waiting Time**	**Scenarios**	**Iterations**
1	55	5, 10 and 20	10	0.02	10, 100 and 1000	100
2	55	5, 10 and 20	10	2	10, 100 and 1000	100
3	55	5, 10 and 20	10	20	10, 100 and 1000	100
4	55	5, 10 and 20	1	20	10, 100 and 1000	100
5	55	5, 10 and 20	10	20	10, 100 and 1000	100
6	55	5, 10 and 20	20	20	10, 100 and 1000	100
7	25	5, 10 and 20	10	0.02	10	500, 1000 and 2000
8	40	5, 10 and 20	10	0.02	10	500, 1000 and 2000
9	50	5, 10 and 20	10	0.02	10	500, 1000 and 2000

Table 5.1. Characteristics and parameters of the worked data

The algorithm in study was implemented on a computer with 2.50 GHz Pentium Dual-Core processor with 1920 MB of memory and using the compiler C++ *Microsoft Visual Studio 2005* which integrates, for the resolution of the problems proposed the optimization software CPLEX *Optimization Studio 12.2*.

In general, we aim to analyze the results checking if the location-allocation patterns, given to us by the heuristics method, show any differences when comparing with the location-allocation pattern from the initial solution in the GRASP Local Search phase. It is also obtained the average value of the percentage deviation associated with the solution obtained which, also in a greedy fashion, we tried to low it as possible.

5.1. Changing the Waiting Time Limits and Distances

A 55 demand nodes network was used for a total of 100 iterations. Varies, in this case, the number of service centers to locate as well as the number of scenarios studied. Additionally, in order to generate different situations for the system, both the waiting time limit on the server (wlim) as the distance limit between the server and the demand node (ldist) are amended.

The tested values for the waiting time limit on the server were wlim = 0.02, wlim = 2 and wlim = 20. In all these cases the distance limit between the server and the demand node was fixed at ldist = 10.

For wlim = 0.02, regardless of the number of scenarios and number centers lo locate, the heuristic always reciprocate an identical final solution to the starting solution (initial solution) thus matching the initial and final locations.

When wlim = 2, it was obtained the greatest number of cases where the final locations differed from the initial locations. It turns out as described in 3 of 9 cases tested, especially when considering the location of 10 service centers and simulating 100 scenarios like when trying to locate 20 service centers, for any of the 10 and 100 scenarios simulations.

By using the waiting time limit on the server of wlim = 20, only in 2 of the 9 cases studied the final locations differed from the initial locations. These are the cases where it is wished to locate 20 service centers, both for 10 and 100 scenarios.

While continuing to review the presented heuristics, basing on the same 55 demand points network with a total of 100 iterations, fixing the waiting time limit on the server (wlim), now were changed the distance limit values between the server and demand center (ldist).

The rested values for the distance limit between server and demand center were ldist = 1, ldist = 10 and ldist = 20. In all those cases the waiting time limit on the server was fixed at wlim = 20.

It is precisely in the observation of ldist = 1, for the proposed location problem, that the heuristic analysis always shows final locations different from the initial location solution used as a starting point.

By increasing the distance limit between the server and the demand center for ldist = 10, in 2 of the 9 cases worked show differences regarding the initial locations and the final locations obtained through the heuristic procedure. This happens when trying to locate 20 service centers for the cases when 10 and 100 scenarios were simulated.

In turn, when it is intended to use ldist = 20, possible conclusions are the same, including in terms of the number of scenarios, with the difference that the non-coincidence of locations arises when pretending to locate 5 centers.

Taking into account all the waiting time limit values tested (wlim = 0.02; 2; and 20 for ldist = 10) and the distance limit between the server and demand center (ldist = 1; 10; and 20 for wlim = 20), when comparing the Minimum Relative Regret obtained for all of them it´s possible to see its positive tendency towards the increase in the number of scenarios tested.

Regarding this late indicator, the heuristic results obtained, related to the increase in the number of centers to locate, don´t show a trusty behavior pattern that allow a generalization. It is possible to observe this both in the variation of ldist and in the variation of wlim.

5.2. Changing the Size of the Network and Number of Centers to Locate

Subsequently a study was conducted where it was processed data obtained simulating 100 samples and considering for each of these a 10 scenarios generation. This way, the used network size varies in terms of the number of nodes representing all points of demand. So we chose to analyze networks with 25, 40 and 50 nodes and, in each one of these cases, varying the number of iterations. Are considered results obtained for 500, 1000 and 2000 iterations.

As regards the average CPU processing time (measured in seconds), it can be seen that this increases due to three factors: network size, i.e., number of demand nodes utilized; iterations number; and, for last, number of service centers to locate during the problem solving. However, this pattern is not as straightforward when assessing the 25 nodes network. In this case, for any number of iterations considered, when going from 5 to 10 centers to locate, the average processing time follows the pattern described above. However, changing from 10 to 20 service centers to locate, the average processing time decreases.

Analyzing the percentage of initial and final locations matching, these values are increasing over the number of network nodes and compared to the number of centers to find. In turn, considering an increasing number of iterations, it is noted, generally, that the percentage of matching locations decreases. Higher values for this indicator are found in smaller networks and with less number of iterations.

		25 Nodes Network			40 Nodes Network			50 Nodes Network		
	Number of Centers to Locate	**5**	**10**	**20**	**5**	**10**	**20**	**5**	**10**	**20**
500 iterations	*Average Processing Time*	1,207	1,268	1,141	2,598	3,449	4,704	3,967	4,96	8,809
	% Matching Locations	5%	37%	95%	10%	24%	82%	14%	24%	92%
	Average Regret	0,11866	0.15134	0,16351	0,08012	0,10989	0,12178	0,04482	0,08237	0.08504
1000 iterations	*Average Processing Time*	1,661	2,227	1,806	3,645	5,661	8,242	5,742	8,015	13,88
	% Matching Locations	1%	21%	88%	5%	15%	77%	12%	22%	90%
	Average Regret	0,1149	0,14312	0,16052	0,07705	0,10031	0,11625	0,06552	0,07983	0,08349
2000 iterations	*Average Processing Time*	2,886	3,733	3,1	5,446	10,535	14,612	8,356	13,939	23,258
	% Matching Locations	5%	14%	84%	6%	7%	66%	18%	11%	64%
	Average Regret	0.10729	0,13702	0,15885	0,06570	0,10122	0,12052	0,03293	0,06982	0,08623

Table 4.2. Simulations Results for 100 examples and 10 scenarios; average processing time measured in seconds

5.3. Conclusions

Increasing the limit for the waiting time (wlim), well as increasing the distance limit (ldist) between demand node and service center, lead us to believe that this heuristic produces solutions that improve the solutions defined originally.

Under the perspective of system congestion, fewer centers locate and a larger network, lead the heuristic to find different solutions from those obtained initially by the Regret method.

It is important here to analyze not only the percentage of matching locations, but also the values of Minimum Relative Regret. By decreasing these indicate that possible solutions were found, and these, in the case of problems for freer systems, deviate less from the starting solution for the heuristic given by the Regret method for the sets of simulated scenarios.

The same conclusion can be drawn if you equate the cases in which the number of service centers to locate were increased, allowing this aspect to generalize the behavior of the heuristics developed.

6. General Conclusions

By analyzing the literature related with the location and allocation problems, it´s easy to realize the trend of including in this type of models the effects of queues. This might happen because, considering a certain demand for service, in reality, it appears that this is random and one of the sources of systems congestion. It is for this reason that this study associates to the Maximum Coverage Location Model formulations related to the queues theory.

This type of problems, that can arise both in the context of the public or private sector, involving different types of formulation as maximum distance modelos or total/average distance models. The methodology associated with each specific problem should be carefully proposed and one should compare the results obtained with others reciprocated by traditional models.

Additionally to the "*Greedy Randomized Adaptive Search Procedure*" (or GRASP), was also used in the heuristic method developed the *p-minmax* Regret (Minimum Relative Regret) proposed by Daskin [2]. The use of these processes is in line with previous research and aims its integrate in order to explore new methodologies that enhance or better adapt to the circumstances of the cases studied. Thus, the developed models can be considered adequate to address the type of issue proposed in the current work. By varying the limits in terms of waiting times and maximum distance, limits of demand processing capabilities and network sizes, one can notice significant changes in the final solutions.

There are numerous real situations where the waiting time is an important factor when considering the length of service (time or distance traveled plus the waiting time). In those cases, taking into account the determination of a location pattern, the waiting time is to be regarded as essential in the respective system modeling. May also interfere with the processing time the number of centers to locate at a certain network scale, as well as the capacity of facilities in providing the sought service.

The proposed meta-heuristic reciprocates near optimal results demonstrating significant savings in computation time. Given the initial data, was with the use of simulation that in the present study the demand levels associated with each population data were obtained.

Regarding the application of Greedy heuristics to these formulations, these show acceptable behavior to the extent that the near-optimal solutions are sensitive to the worked examples and problematic situations proposed in each case.

On the other hand, according to the theory and the numerical examples obtained, suggest that the solutions become less sensitive to the model parameters as the system becomes less busy. In the case that, for instance, the distance limit between the demand node and the service facility is smaller or when there are less service centers to locate, one can assume that now there is a greater congestion associated with the model. The latter are precisely those cases where the heuristic has given results not identical to the initial solutions used as input to our algorithm.

As for generalizing conclusions regarding the computational experiment conducted, special care should be taken. The tested models and their various examples were obtained using random number generation. In many cases, it is worth noting different results but there are others where the proposed formulation does not produce significant differences in the results. As already mentioned, in general, the most "tight" systems, in other words, when the distance limit is smaller, the number of service centers to locates is smaller or for inferior processing capabilities, location decisions are more sensitive to pre-defined parameters for the model.

In conclusion, having simulated populations and the respective demand frequencies, with this work it is possible to highlight the paramount importance, as in real life, of consider systems congestion in its various forms as a determining factor in location and allocation decisions.

References

1. Church, R. and C.R. VELLE, *The maximal covering location problem.* Papers in Regional Science, 1974. **32**(1): p. 101-118.
2. Daskin, M.S., S.M. Hesse, and C.S. Revelle, *α-reliable p-minimax regret: A new model for strategic facility location modeling.* Location Science, 1997. **5**(4): p. 227-246.
3. Current, J., M. Daskin, and D. Schilling, *3 D iscrete N etw ork Location M odels.* 2002.
4. Marianov, V. and D. Serra, *Probabilistic, maximal covering location-allocation models for congested systems.* Journal of Regional Science, 1998. **38**(3): p. 401-424.
5. Silva, F. and D. Serra, *Locating emergency services with different priorities: the priority queuing covering location problem.* Journal of the Operational Research Society, 2008. **59**(9): p. 1229-1238.
6. Hakimi, S.L., *Optimum locations of switching centers and the absolute centers and medians of a graph.* Operations Research, 1964: p. 450-459.

7. Hakimi, S., *Optimum distribution of switching centers in a communication network and some related graph theoretic problems.* Operations Research, 1965. **13**(3): p. 462-475.
8. Sheppard, E., *A conceptual framework for dynamic location-allocation analysis.* Environment and Planning A, 1974. **6**(5): p. 547-564.
9. ReVelle, C. and K. Hogan, *The maximum availability location problem.* Transportation Science, 1989. **23**(3): p. 192-200.
10. Hogan, K. and C. Revelle, *Concepts and applications of backup coverage.* Management Science, 1986. **32**(11): p. 1434-1444.
11. Daskin, M.S., *APPLICATION OF AN EXPECTED COVERING MODEL TO EMERGENCY MEDICAL SERVICE SYSTEM DESIGN*.* Decision Sciences, 1982. **13**(3): p. 416-439.
12. Daskin, M.S., *A Maximum Expected Covering Location Model - Formulation, Properties and Heuristic Solution.* Transportation Science, 1983. **17**(1): p. 48-70.
13. Larson, R.C., *A hypercube queuing model for facility location and redistricting in urban emergency services.* Computers & Operations Research, 1974. **1**(1): p. 67-95.
14. Batta, R., *A queueing-location model with expected service time dependent queueing disciplines.* European Journal of Operational Research, 1989. **39**(2): p. 192-205.
15. Marianov, V. and D. Serra, *4 Location Problems in the Public Sector.* 2002.
16. Chiu, S.S., O. Berman, and R.C. Larson, *Locating a mobile server queueing facility on a tree network.* Management Science, 1985: p. 764-772.
17. Batta, R. and O. Berman, *A Location Model for a Facility Operating as a M/G/K Queue.* Networks, 1989. **19**(6): p. 717-728.
18. Brandeau, M.L. and S.S. Chiu, *A center location problem with congestion.* Annals of Operations Research, 1992. **40**(1): p. 17-32.
19. Marianov, V. and C. Revelle, *The queuing probabilistic location set covering problem and some extensions.* Socio-Economic Planning Sciences, 1994. **28**(3): p. 167-178.
20. Kleinrock, L., *Queueing Systems. Volume 1: Theory.* 1975.
21. Berman, O. and D. Krass, *11 Facility Location Problems with Stochastic Demands and Congestion.* Facility location: Applications and theory, 2002: p. 329.
22. Rayward-Smith, V.J., et al., *Modern heuristic search methods*1996: Wiley New York.
23. Feo, T.A. and M.G.C. Resende, *Greedy randomized adaptive search procedures.* Journal of global optimization, 1995. **6**(2): p. 109-133.
24. Teitz, M.B. and P. Bart, *Heuristic methods for estimating the generalized vertex median of a weighted graph.* Operations Research, 1968. **16**(5): p. 955-961.
25. Festa, P. and M.G.C. Resende, *An annotated bibliography of GRASP.* Operations Research Letters, 2004. **8**: p. 67-71.
26. ReVelle, C., *THE MAXIMUM CAPTURE OR "SPHERE OF INFLUENCE" LOCATION PROBLEM: HOTELLING REVISITED ON A NETWORK*.* Journal of Regional Science, 1986. **26**(2): p. 343-358.
27. Balinski, M.L., *Integer programming: methods, uses, computations.* Management Science, 1965. **12**(3): p. 253-313.