A Multiobjective Approach for a Dynamic Simple Plant Location Problem under Uncertainty

Joana Dias[1,2], Maria do Céu Marques[2,3]

[1]Faculdade de Economia, Universidade de Coimbra, 3004-512 Coimbra, Portugal
[2]Inesc-Coimbra, Rua Antero de Quental, 199, 3000-033 Coimbra, Portugal
[3]Instituto Politécnico de Coimbra, ISEC, DFM, Rua Pedro Nunes, 3030-199 Coimbra,
`joana@fe.uc.pt, cmarques@isec.pt`

Abstract. Location problems are, by nature, strategic decisions since facilities will usually be in operation in the medium and long terms. It is necessary to decide today, given the available information, knowing that the consequences of today's decisions will remain in time. Having to include in the decision making process data that will only be known with certainty in the future does not advise the use of deterministic models: the existing uncertainty should be explicitly included in the models. The notion of *optimal solution* becomes fragile: it will be difficult to find a single solution that is the best in all possible future realizations of uncertainty. In this paper we consider a dynamic simple plant location problem, where uncertainty is explicitly considered through the use of scenarios. We advocate the use of a multiobjective approach as a valuable tool in guiding the decision-making process, iteratively or as an *off-line* generation procedure.

Keywords: Location Problems, Uncertainty, Scenarios, Multiobjective, Pareto-efficient

1   Introduction

Plant location problems are strategic problems by nature, usually associated with significant investments, and with medium to long term consequences. Whenever a decision-maker (DM) has to decide where to locate a warehouse, a plant, a store, the amount of data that should be considered in the decision making process is not only huge but often not known with certainty at the time the decision has to be made. Furthermore, location decisions are often associated with other problems like inventory management, transportation or assignment problems. As a simple example, let us consider the problem of locating a warehouse that will serve as supplier for a set of customers. There are only two potential locations for the warehouse: one location has very low construction and other fixed costs, but it is very far away from any of the customers (implying important transportation costs); the other location is near most customers, but the fixed costs incurred are very high. When deciding, we cannot split the location and transportation/assignment problems in two distinct problems, or we

would end up with a suboptimal solution. They have to be considered simultaneously in the decision making process. However, dealing with those two problems at the same time is not as straightforward as it might seem, especially when we are considering that the facility to be located will be in operation during a given time horizon. Although it is not easy to change the location of warehouses or other facilities from one time period to the next, because of the direct and indirect costs incurred, it is not that difficult to change the way the demands of customers are being satisfied and the goods transported. If there is a change in the road infrastructure, or an important increase or decrease in fuels' prices, or there is a road that became tolled, the decision maker will be able to adjust the assignment/transportation decisions to the new reality. A similar reasoning could be made if we were thinking of other problems like inventory management or production scheduling.

In plant location problems we are thus faced with having to simultaneously make decisions that are deeply interconnected but that have very different natures: on one hand, location decisions that are strategic by nature and difficult to reverse, on the other hand, other related decisions that can be easily reverted whenever new information is available. The strategic nature of location problems is clearly described and emphasized in [1], where not only the classical static and deterministic models are defined, but also dynamic and stochastic location problems are discussed.

In this paper a dynamic simple plant location problem under uncertainty is considered. There is a set of potential locations for opening facilities, and there is a set of clients with a given demand that has to be satisfied by opened facilities. There is a planning horizon, and it will be necessary to decide which facilities to open and when to open them. The only data that is known for sure is the one related with the present moment: if the facilities are opened today, and the assignment of clients to facilities is made today, then all fixed location costs and assignment costs for the present time period are known with certainty. However, for all other future time periods, fixed opening costs are not known for sure, as well as assignment costs. In the future, even the set of clients or the potential locations for new facilities can be uncertain. In the present work, uncertainty is represented through a set of possible future scenarios. Two different stages for decision-making are considered: on one hand, location decisions determining when and where facilities should be opened during the planning horizon are set at the present moment and cannot be changed throughout the planning horizon (first stage); on the other hand, the assignment of clients to facilities that is made in each time period considering the existing opened facilities and the assignment costs for that particular period and for the scenario that came to occur (second stage).

Whenever uncertainty is explicitly considered, the notion of both feasible and optimal solution becomes very fragile (see, for instance, [2]). Regarding feasibility, it can be the case that it is not possible to guarantee feasibility under all possible realizations of the uncertain parameters. In the present work, as we are dealing with an uncapacitated problem, a solution that is feasible for one scenario will be feasible for all others. Regarding optimality, it is often the case that no single solution will be optimal under all possible future scenarios. A possible approach would be to consider a single objective function that would represent the expected objective function value

over all scenarios. Optimizing the expected value of a given objective function can be the best thing to do if we are facing a decision making process that has to be repeated over and over again, always under uncertainty. Only in this case does the expected value represent the DM's payoff in the long run. Location problems do not fit this profile. Most of the times they are "once in a lifetime" decisions. Furthermore, we cannot talk about uncertainty without thinking about risk, and different DMs have different attitudes towards risk [3]. If a DM is neutral towards risk, then an expected value approach could be defendable. But if the DM is averse to risk, then he will probably be concerned with the worst of all scenarios only: he would be interested in making the worst possible outcome the best possible. This approach is known as the min-max approach. Many other possible ways of dealing with uncertainty can be disguised, some under an umbrella usually known as robust optimization. A review of facility location under uncertainty can be found in [4].

There is a whole set of publications dedicated to multiobjective stochastic programming, usually tackling the problem by reducing it to a single objective stochastic program or transforming it to a deterministic multiobjective program (see, for instance, [5, 6, 7, 8, 9, 10, 11]). The approach described in this paper is different: we tackle a single objective location problem under uncertainty by resorting to a multiobjective approach. The concept of Pareto-efficiency is thus applied in the context of a single objective problem under uncertainty. Without making any kind of assumptions regarding the attitude of the decision maker towards risk, it is possible to consider the dynamic plant location problem under uncertainty as a multiobjective problem, where each scenario will give rise to one objective. The DM will only be interested in Pareto-efficient solutions. A Pareto-efficient solution can be understood as a solution where it is not possible to improve one objective without deteriorating at least one other. In this context, a solution will be Pareto-efficient if it is not possible to improve the objective function value under one scenario without deteriorating its value under at least one other scenario. The multiobjective approach can be used embedded in an interactive decision-making process or as an *off-line* generation procedure, where the whole set of efficient solutions is calculated.

Actually, it is very difficult to find the concept of Pareto efficiency being applied in this context. Iancu and Trichakis [12] have, in a recent paper, introduced the Pareto efficiency concept in robust optimization, showing that some robust solutions are not efficient, and apply their methodology to portfolio optimization, inventory management and project management. Nevertheless, their focus is solely on Pareto efficient worst-case solutions. Kaläi et al [13] introduces the concept of lexicographic α-robustness, that can be seen as an extension of lexicographic programming used to tackle multiobjective problems.

This paper is organized as follows: the next section presents the mathematical model, and introduces the main concepts used throughout the text. Section 3 describes the multiobjective approach, and shows how some "robust solutions" can be, in fact, Pareto-efficient solutions. Section 4 illustrates the procedure through the use of some illustrative examples. In Section 5 the main conclusions are stated, and paths for future research are delineated.

2   Mathematical Modelling

Consider the set of potential locations for facilities denoted by $J = \{1,...,j,...,M\}$, and the set of possible customers' locations denoted by $I = \{1,...,i,...,N\}$. The planning horizon is defined as a set of time periods $\Gamma = \{1,...,t,...,T\}$. There are fixed costs associated with opening a facility at a given location and keeping this facility open until the end of the planning horizon. There are also assignment costs related to the assignment of customers to opened facilities during the planning horizon. The uncertainty considered in this problem is present not only in the fixed and assignment costs, but also in the existence of each possible client in each time period, and in the possibility of opening a facility in each potential location in the beginning of each time period. The uncertainty is represented by a set of possible future scenarios $\Phi = \{1,...,s,...S\}$. Suppose that each scenario s will occur with probability p_s such that $\sum_{s \in \Phi} p_s = 1$. The decision maker can be aware of these probabilities or not. Consider the following notation:

$$\delta_{its} = \begin{cases} 1, & \text{if customer } i \text{ has a demand to be fulfilled during} \\ & \text{period } t \text{ under scenario } s \\ 0, & \text{otherwise} \end{cases} \quad , i \in I, t \in \Gamma, s \in \Phi;$$

$$\beta_{jts} = \begin{cases} 1, & \text{if it is possible to open facility } j \text{ at the beginning} \\ & \text{of period } t \text{ under scenario } s \\ 0, & \text{otherwise} \end{cases} \quad , j \in J, t \in \Gamma, s \in \Phi;$$

$f_{jts} = $ fixed cost of opening facility j in the beginning of period t plus operation and maintenance costs until the end of the planning horizon (and possibly closing costs if the facility is to be closed after T) under scenario s, $j \in J, t \in \Gamma, s \in \Phi$;

$c_{ijts} = $ cost of assigning customer i to facility j during period t under scenario s, $i \in I, j \in J, t \in \Gamma, s \in \Phi$.

The decision variables to be considered are defined as follows:

$$x_{ji} = \begin{cases} 1, & \text{if facility } j \text{ is opened at the beginning of period } t \\ 0, & \text{otherwise} \end{cases} \quad , j \in J, t \in \Gamma;$$

$$y_{ijts} = \begin{cases} 1, & \text{if customer } i \text{ is assigned to facility } j \text{ in period } t \\ & \text{under scenario } s \\ 0, & \text{otherwise} \end{cases} \quad , i \in I, t \in \Gamma, s \in \Phi.$$

Notice that location decisions represented by x_{ji} do not depend on the realizations of the uncertain parameters, but the assignment decisions do: their values are only determined when the uncertainty is resolved for period t.

Representing location variables x_{ji} by a vector $\boldsymbol{x}$ and assignment variables by a vector $\boldsymbol{y}$, the simple dynamic location problem under uncertainty can be formulated as follows.

SDLPU:

$$"min" G(\boldsymbol{x}, \boldsymbol{y}) \qquad (1)$$

Subject to:

$$\sum_{j \in J} y_{ijts} = \delta_{its}, i \in I, t \in \Gamma, s \in \Phi \qquad (2)$$

$$\sum_{\tau=1}^{t} x_{j\tau} \geq y_{ijts}, i \in I, j \in J, t \in \Gamma, s \in \Phi \qquad (3)$$

$$\sum_{t \in \Gamma} x_{jt} \leq 1, j \in J \qquad (4)$$

$$x_{jt} \leq \beta_{jts}, j \in J, t \in \Gamma, s \in \Phi \qquad (5)$$

$$x_{jt} \in \{0,1\}, j \in J, t \in \Gamma \qquad (6)$$

$$y_{ijts} \in \{0,1\}, i \in I, j \in J, t \in \Gamma, s \in \Phi \qquad (7)$$

Constraints (2) guarantee that each client with demand will be assigned to a facility in each period t and scenario s. Constraints (3) guarantee that clients can only be assigned to opened facilities. Constraints (4) state that a facility can only be opened once during the whole planning horizon. Constraints (5) guarantee that it is only possible to open facilities in a given time period if that is allowed by all scenarios. If we consider that if β_{jts} is equal to 0 then f_{jts} is equal to $+\infty$, then constraints (5) can be dropped.

After fixing variables x_{jt}, a simple assignment problem has to be solved for each time period t and scenario s. We assume that the assignment problem is solved only when we know what was the scenario that came to occur. It is worth noticing that these assignment problems will always have a feasible solution as long as at least one facility is open at each time period, since the model does not consider capacity constraints.

Although the meaning of constraints is clear, this problem is still not well-defined. Since we are in the presence of possible future scenarios, how should we interpret "min"? Let us consider that we are interested in minimizing total costs. Even so, different decision makers will have different interpretations of (1).

When a decision has to be taken under uncertainty, and the consequences of that decision depend on the realization of uncertainty that comes to occur, the objective function to consider is not independent of the risk profile of the DM. DMs that are risk averse will tend to look for min-max solutions: solutions that minimize the worst

possible outcome. DMs that like to take risks could prefer making a decision that could lead to the most advantageous payoff, even if this "best of all" payoff only has a tiny change of occurring. One way of interpreting (1) could be to consider the expected value of a given objective function over all scenarios. If this is the case, and we are interested in the minimization of the expected total cost, then (1) could be interpreted as (8).

$$\min \sum_{s \in \Phi} \sum_{t \in \Gamma} \sum_{j \in J} p_s f_{jts} x_{jt} + \sum_{s \in \Phi} \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} p_s c_{ijts} y_{ijts} \tag{8}$$

It is also possible to imagine that we are interested in guaranteeing that the worst payoff is the best possible (min-max approach):

$$\min_s \left\{ \max \sum_{t \in \Gamma} \sum_{j \in J} f_{jts} x_{jt} + \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} c_{ijts} y_{ijts} \right\} \tag{9}$$

We can also think that what really matters is to consider the minimization of the total cost for the most probable scenario s^*:

$$\sum_{t \in \Gamma} \sum_{j \in J} f_{jts^*} x_{jt} + \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} c_{ijts^*} y_{ijts^*} \tag{10}$$

Or minimize the maximum regret, where F_s^* represents the optimal objective function value when considering only scenario s:

$$\min_s \left\{ \max \left(\sum_{t \in \Gamma} \sum_{j \in J} f_{jts} x_{jt} + \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} c_{ijts} y_{ijts} - F_s^* \right) \right\} \tag{11}$$

In reality, the DM will be interested in compromise solutions: it is seldom the case where a single solution will be the best under all possible realizations of uncertainty. Making a decision will always give a better result under some circumstances than under others, and the DM is not capable of controlling what these circumstances will be. If it is not possible to do any kind of assumptions regarding the risk profile of the decision maker, or about his preferences, then one possible approach could be to interpret (1) not as one single objective function but as a set of objective functions instead. Let $F_s(x, y)$ represent the total cost incurred under scenario s when solution (x, y) is considered (12).

$$F_s(x, y) = \sum_{t \in \Gamma} \sum_{j \in J} f_{jts} x_{jt} + \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} c_{ijts} y_{ijts}, s \in \Phi \tag{12}$$

SDLPU would then become a multiobjective problem (SDLPU_MO), where (1) is interpreted as (13).

SDLPU_MO:

$$\min\left\{F_1(\boldsymbol{x},\boldsymbol{y}),\cdots,F_s(\boldsymbol{x},\boldsymbol{y}),\cdots,F_S(\boldsymbol{x},\boldsymbol{y})\right\} \quad (13)$$

Subject to: (2)-(7)

Independently of the preferences or profile of the DM and assuming only that the DM is rational, he will only be interested in solutions such that it is not possible to improve the objective function of one given scenario without deteriorating the objective function of at least one other scenario. This means that the DM will only be interested in Pareto-efficient solutions (known as non-dominated solutions if we consider the objective space).

Definition 1: Consider $(\boldsymbol{x},\boldsymbol{y})$ an admissible solution for SDLPU_MO. $(\boldsymbol{x},\boldsymbol{y})$ is a Pareto efficient solution if and only if there is no other solution $(\boldsymbol{x}',\boldsymbol{y}')$ such that $F_s(\boldsymbol{x}',\boldsymbol{y}') \leq F_s(\boldsymbol{x},\boldsymbol{y})$ and $F_s(\boldsymbol{x}',\boldsymbol{y}') < F_s(\boldsymbol{x},\boldsymbol{y})$ for at least one scenario s. The image of an efficient solution in the objective space is known as a *non-dominated solution*.

3   Multiobjective Approach

There are several different ways of dealing with a multiobjective programming problem. One such way is the so-called interactive approach. The interactive approach considers interchanging calculation and dialogue phases. In the calculation phase a non-dominated solution is calculated and showed to the DM. The DM will then react by giving some new information that will guide the calculation of the new non-dominated solution to be calculated in the next iteration. The process continues until the DM is satisfied with a given solution or the total set of non-dominated solutions is found (see, for instance, [14]). The major drawback of this approach has to do with the possibility of having calculation phases taking too much computational time, not promoting a real-time interaction and making the process not attractive to the DM. The main advantage has to do with the ability of searching areas of the solutions' surface that are interesting to the DM, not wasting time or resources calculating solutions that the DM will simply discard. Moreover, whenever a non-dominated solution is encountered, there is a region in the objective space that is no longer interesting (the one that is dominated by this solution), and another region where there cannot be any admissible solutions (or else this solution would not be non-dominated). So, it is possible, in each iteration, to eliminate regions from further searches.

Another way of dealing with multiobjective problems considers the *a priori* and *off-line* calculation of the whole set (or a significant number) of non-dominated solutions. The solutions can then be presented to the DM, all at the same time, or using an interactive approach similar to the one previously described. The set of non-dominated solutions can even be analyzed by using multicriteria decision-making techniques ([15, 16]). One of the advantages of this approach is that the computational burden of calculating the solutions is made *a priori*, promoting a faster action-reaction

interaction with the decision maker since no optimizations will be done. Furthermore, it will be possible to show more information to the DM, like statistics or the fulfillment of other criteria not explicitly considered in the model.

The choice between an interactive or a generation approach should be done considering several aspects of the problem: What is the dimension of the problem? How long does it take to calculate a solution? Whatever is the choice in a particular situation, there has to be some kind of procedure to calculate non-dominated solutions. There are several auxiliary programming problems that can be used to calculate non-dominated solutions, mostly relying on the optimization of some mono-objective programming problem, being the most well known the consideration of a weighted sum of the objective functions. When dealing with integer or mixed-integer problems, care has to be taken to guarantee that the chosen procedure is capable of calculating non-supported non-dominated solutions (lying inside duality gaps).

In this paper we resort to a result due to Ross and Soland [17], considering an auxiliary mono-objective programming problem AUX, where $\lambda_s > 0$ is a weight associated with scenario s such that $\sum_{s \in \Phi} \lambda_s = 1$, and M_s is an upper bound to the objective function related to scenario s.

AUX:

$$\min \sum_{s \in \Phi} \sum_{t \in \Gamma} \sum_{j \in J} \lambda_s f_{jts} x_{jt} + \sum_{s \in \Phi} \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} \lambda_s c_{ijts} y_{ijts} \tag{14}$$

Subject to: (2)-(7)

$$\sum_{t \in \Gamma} \sum_{j \in J} f_{jts} x_{jt} + \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} c_{ijts} y_{ijts} \leq M_s, s \in \Phi \tag{15}$$

Let $M \in \Re^S$ represent the vector of M_s values.

Proposition 1 (based on [17]): For any vector λ such that $\sum_{s \in \Phi} \lambda_s = 1$ and $\lambda_s > 0, s \in \Phi$, (x, y) is an efficient solution of SDLPU_MO if and only if it is the optimal solution of AUX for some vector $M \in \Re^S$.

Based on Proposition 1, it is now possible to verify whether some of the most common interpretations of (1) will lead to efficient solutions or not. Considering objective function (8), if constraints (15) are added to SDLPU, we can guarantee that the corresponding solution will be an efficient solution.

Proposition 2: If $\lambda_s = p_s, \forall s$, then AUX will calculate a solution considering the minimization of the expected total cost over all scenarios. This solution is a non-dominated solution to SDLPU_MO.

Proof: The proof follows directly from Proposition 1. If $\lambda_s = p_s, \forall s$ and if M_s is made large enough, then we are actually minimizing the expected total cost. We can thus conclude that this solution is a non-dominated solution of SDLPU_MO.

Let us define a solution as α-robust, $0 < \alpha < 1$, if and only if the objective function value under any of the scenarios is not more than $\alpha\%$ worst than the best possible solution for that particular scenario [18]. If M_s are properly chosen, then AUX can be used to calculate an α-robust solution, and it is possible to guarantee that the α-robust solution is indeed efficient. Let F_s^* be the optimal objective function value if only scenario s is considered.

Proposition 3: The solution obtained by solving AUX such that $M_s = (1+\alpha)F_s^*, s \in \Phi$ and $0 < \alpha < 1$, is a α-robust non-dominated solution of SDLPU_MO.

Proof: The result follows directly from Proposition 1.

It has been proven that the solution obtained by considering the minimization of the worst outcome can be dominated [12]. Nevertheless, AUX could also be used to calculate an efficient min-max solution. In a first stage, it would be necessary to solve the problem of minimizing the maximum cost under all scenarios. This could be done by solving the following programming problem:
MIN-MAX:

$$\min \delta \qquad\qquad (16)$$

Subject to:

(2)-(7)

$$\sum_{t \in \Gamma} \sum_{j \in J} f_{jts} x_{jt} + \sum_{t \in \Gamma} \sum_{j \in J} \sum_{i \in I} c_{ijts} y_{ijts} \le \delta, s \in \Phi \quad (17)$$

After solving this problem, it will be possible to guarantee the calculation of an efficient solution that is also min-max optimal by considering appropriate values for M_s.

Proposition 4: Let δ represent the worst case objective function, calculated by solving MIN-MAX. If M_s is defined such that $M_s = \delta, s \in \Phi$, then AUX will generate an efficient min-max solution.

Proof: After solving MIN_MAX, δ will represent the worst case objective that we are willing to accept. This means that we will only be interested in solutions such that the objective function value for any scenario s will be less than or equal to δ. Comparing (17) with (15), it is easy to see that if M_s is defined such that $M_s = \delta, s \in \Phi$, then any efficient solution calculated will also be a min-max solution.

A similar reasoning could be applied if we are interested in minimizing the maximum regret. Consider the following restrictions:

$$\sum_{t\in\Gamma}\sum_{j\in J} f_{jts} x_{jt} + \sum_{t\in\Gamma}\sum_{j\in J}\sum_{i\in I} c_{ijts} y_{ijts} - F_s^* \leq \delta, s \in \Phi \tag{18}$$

Proposition 5: Let δ represent the maximum allowed regret calculated by solving MIN-MAX, with restriction (17) replaced by (18). If M_s is defined such that $M_s = F_s^* + \delta, s \in \Phi$, then AUX will generate an efficient solution that minimizes maximum regret.

Proof: It follows the previous proof.

It is quite easy to embed the use of AUX in both an interactive and an *off-line* generation procedure. In an interactive approach, the dialogue phase with the DM consists in defining new M_s values. Notice that, in reality, these values do no more no less than defining a region of search. In a generating approach, M_s values can be automatically generated in a way that guarantees that the whole objective space is explored. The automatic generation of vector $\boldsymbol{M}$ can be done resorting to two simple data structures: a binary tree, with as much levels as the number of scenarios, and a matrix. Each time a new solution is calculated based on a given vector $\boldsymbol{M}$, a binary tree is generated such that it will define all possible future vectors $\boldsymbol{M}$. These vectors are then saved in a matrix so that they can be retrieved in future iterations. To give a simple example of this procedure, imagine a situation with three scenarios. The initial vector $\boldsymbol{M}$ is equal to $\left[M_1^1, M_2^1, M_3^1\right]$. By using this vector, we obtain a non-dominated solution with the following values for each of the 3 objective function values: $\left[F_1^1, F_2^1, F_3^1\right]$, with $F_1^1 \leq M_1^1, F_2^1 \leq M_2^1, F_3^1 \leq M_3^1$. Based on these two vectors, a binary tree can be built (Fig. 1), where 8 possible search regions are defined.
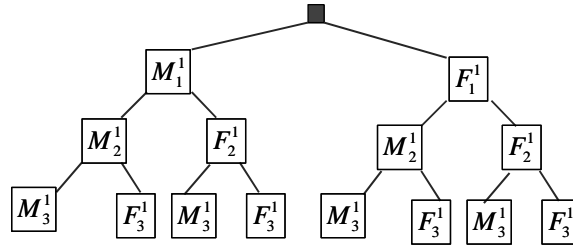

Fig. 1. Binary tree for automatic generation of vector $\boldsymbol{M}$.

The path from the root to each node of the tree will define a new future vector $\boldsymbol{M}$. These values can be stored in a matrix, so that they can be retrieved in a future iteration of the algorithm (Table 1). Whenever a new solution is calculated, a new binary tree is built and the corresponding vectors added to the matrix. In each iteration, vec-

tor M will be determined by the next column of this matrix. Some of the problems will be unfeasible and should not be considered in the matrix. Using $\left[F_1^1, F_2^1, F_3^1\right]$, for instance, will not be interesting because it corresponds to an unfeasible problem (if that was not so, $\left[F_1^1, F_2^1, F_3^1\right]$ would not be a non-dominated solution). Other vectors will end up with optimal solutions that are already known ($\left[M_1^1, M_2^1, M_3^1\right]$, for instance, is not an interesting vector). Furthermore, knowing that one given problem is impossible will allow us to conclude that other M vectors will also lead to impossible problems (if $[M_1^1, F_2^1, M_3^1]$ leads to a problem that is impossible, then it is not worth to explore region $[M_1^1, F_2^1, F_3^1]$, for instance). This search method is easily implementable and will guarantee that the whole objective space is explored.

Table 1. Example of a table for automatic definition of vector M.

M_1^1	M_1^1	M_1^1	F_1^1	F_1^1	F_1^1
M_2^1	F_2^1	F_2^1	M_2^1	M_2^1	F_3^1
F_3^1	M_3^1	F_3^1	M_3^1	F_3^1	M_3^1

It should be stressed that the weights vector λ that is used in (14) does not represent any kind of DM's preferences. These weights can and should be changed in accordance with vector M in order to help decreasing the computational time needed to calculate a solution [14]. If M is more demanding for a given scenario, meaning that M_s is close to the best objective function value F_s^*, then the respective objective function weight should be increased. One simple way of doing this is setting λ as follows:

$$\lambda_s = 1 - \frac{M_s - F_s^*}{F_s^*}, s \in \Phi$$

$$\lambda_s = \frac{\lambda_s}{\sum\limits_{s \in \Phi} \lambda_s}, s \in \Phi \tag{19}$$

Actually, the AUX formulation presented can result in a computationally heavy integer programming problem. It is a NP-hard problem, and the computational time needed to calculate a given solution will be heavily dependent on the dimension of the problem, especially the number of scenarios and the number of potential locations for facilities.To solve AUX we can resort to general solvers or use dedicated procedures, both exact and heuristic procedures. Although the latter will not be able to guarantee the optimality of the calculated solution, they can be a very good choice especially in the presence of an interactive procedure, where the most important thing will be to define a region of interest for the DM. It is even possible to think of using a heuristic procedure in a first stage, and then an exact procedure to actually guarantee the optimality of the solution of interest. Although any number of scenarios can be consid-

ered, in many real life situations DMs look at small number of possible future realizations of uncertainty (usually the worst, best and most probable scenarios).

4   Illustrative Examples

In this section we will illustrate the multiobjective approach to the simple plant location problem under uncertainty by resorting to two small examples of randomly generated problems. The problems were generated using the algorithm described in [19]. All AUX problem instances were solved by a general solver (Cplex V12.6). In the first example, we consider a problem with 25 potential locations for facilities, 100 potential clients, 10 time periods and 2 scenarios. Let us consider the use of an interactive procedure based on [14]. The first thing to do is to calculate the optimal solution for both scenarios. The best possible value of the objective function for each scenario will delineate the region of interest. These solutions are shown in Fig. 2, in the objective space. The DM is free to set the vector M as he wishes. Let us assume that he does not want to explore any particular region, so he decides to define $M_1 = 218195$ and $M_2 = 153313$. These limits are defined by the two non-dominated solutions already calculated (that correspond to the optimum solution of each scenario). The weights are considered to be $\lambda_1 = 0.8$ and $\lambda_1 = 0.2$ (according to (19)). The solution reached is shown in Fig. 3.
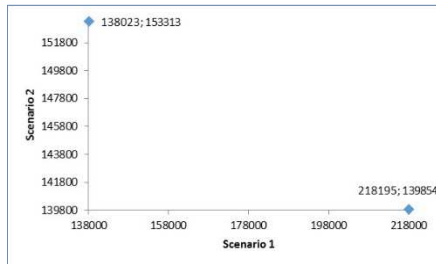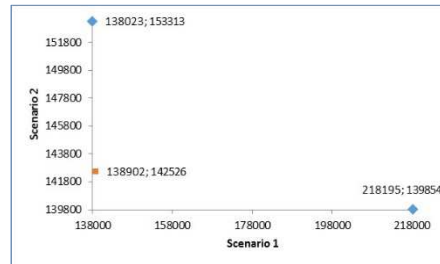

Fig. 2. Optimal solutions for each scenario.

Fig. 3. The first non-dominated solution calculated.

Let us look at more detail at Fig. 3. Considering the newly calculated non-dominated solution, it is easy to see that two regions of the objective space are no longer interesting. This is shown in Fig. 4. Region A will only have solutions that are dominated by the solution calculated. Region B has only non-admissible solutions. The decision-maker can now decide whether to explore region C or region D. Let us suppose that he would explore region D. Then M_1 will remain equal to 218195 and M_2 will be set to 142526 (given by the new non-dominated solution just calculated). Fig. 5 shows the new solution calculated. The procedure would be repeated until the DM is satisfied, or the whole objective space has been explored. The whole set of non-dominated solutions found is shown in Fig. 6. It is possible to observe the compromises that exist between the two scenarios. Moreover, it is also possible to calcu-

late the non-dominated solution that minimizes the maximum regret or that minimizes the maximum cost under both scenarios. These solutions are highlighted in Fig. 7.
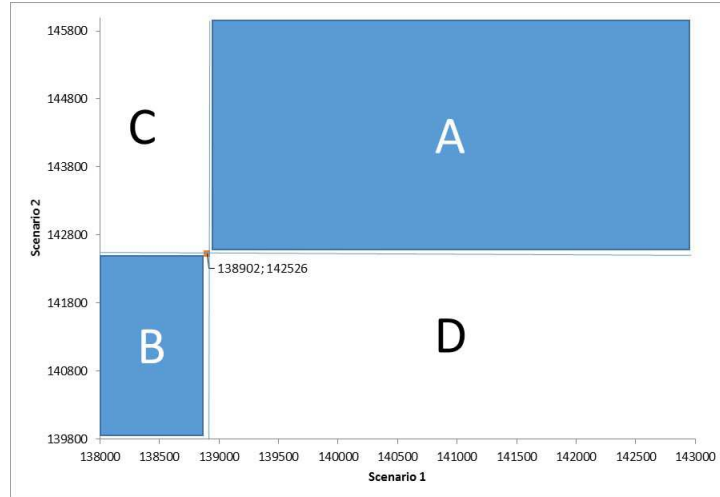

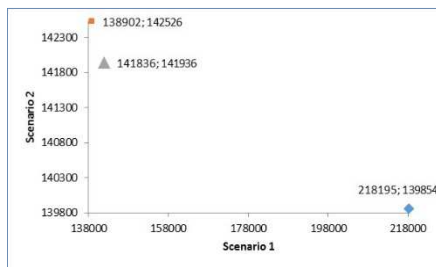**Fig. 4.** The shaded areas A and B are no longer of interest.

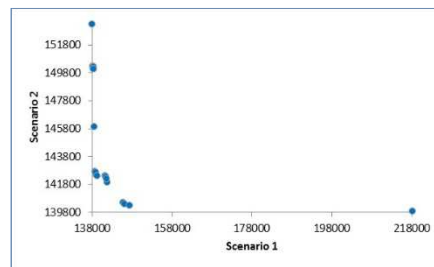

Fig. 5. A new non-dominated solution.

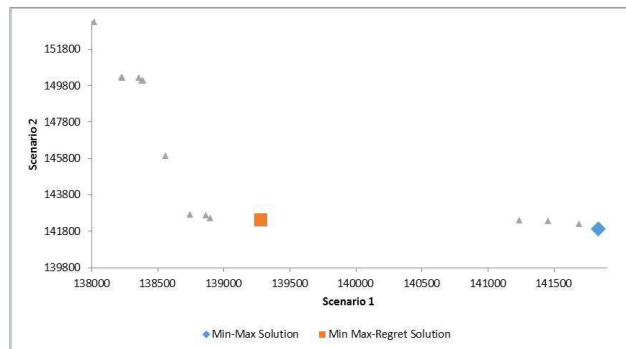**Fig. 6.** The set of non-dominated solutions.

Fig. 7. The Min-Max and Min-Max Regret Solutions

It can also be interesting to analyze the solutions by looking at the opened facilities. In this problem, a set of 7 facilities is opened exactly in the same time period in all solutions calculated. Table 2 shows, for each non-dominated solution, which are the facilities to be opened and when should they be opened.

Table 2. Time period in which each facility is opened in each solution.

Objective Function		Opened Facilities														
s=1	s=2	2	3	4	5	6	7	9	11	14	16	18	20	22	24	25
138023	153313	1	1	-	7	6	4	2	2	3	2	4	6	3	2	-
138228	150276	1	1	-	7	6	4	2	2	3	2	4	6	3	2	1
138237	150257	1	1	-	7	6	4	2	2	3	3	4	6	3	2	1
138360	150238	1	1	-	7	6	4	2	2	3	3	4	-	3	2	1
138384	150093	1	1	-	-	6	4	2	2	3	2	4	6	3	2	1
138393	150074	1	1	-	-	6	4	2	2	3	3	4	6	3	2	1
138564	145957	1	1	-	7	6	4	2	5	3	2	4	6	3	2	-
138720	145827	1	1	-	-	6	4	2	5	3	2	4	6	3	2	-
138746	142709	1	1	-	7	6	4	2	5	3	2	4	6	3	2	1
138869	142690	1	1	-	7	6	4	2	5	3	2	4	-	3	2	1
138902	142526	1	1	-	-	6	4	2	5	3	2	4	6	3	2	1
139281	142430	1	1	7	-	6	4	2	5	3	2	4	6	3	2	1
141238	142389	-	1	-	7	6	4	2	5	3	2	4	6	3	2	1
141457	142365	-	1	-	-	6	4	2	5	3	2	4	6	3	2	1
141695	142200	-	1	7	-	6	4	2	5	3	2	1	6	3	2	1
141836	141936	-	1	7	-	6	4	2	5	3	2	4	6	3	2	1
145742	140500	1	1	-	-	6	4	2	5	3	2	4	2	3	2	1
146121	140404	1	1	7	-	6	4	2	5	3	2	4	2	3	2	1
147507	140307	-	1		7	6	4	2	5	3	2	4	2	3	2	1
218195	139854	-	1	7	-	6	4	2	5	3	2	4	2	3	2	1

In Fig. 8, the non-dominated solutions calculated by using the *off-line* generation procedure are shown, considering a randomly generated problem with 10 potential locations for facilities, 50 potential clients, 5 time periods and 3 scenarios. With more than 3 scenarios, the visualization of non-dominated solutions in the objective space is no longer possible, and it could be better to resort to the use of multicriteria analysis tools.
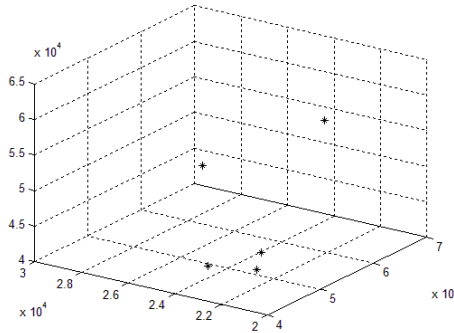

Fig. 8. Non-dominated solutions in 3D objective space

5 Conclusions

In this paper we describe a multiobjective approach for decision-making applied to a simple dynamic location problem under uncertainty. This approach has several ad-

vantages when compared with the more traditional robust approaches. Regarding the DM, the only assumption made is that the DM is rational, thus will only be interested in Pareto-efficient solutions. It is not necessary to estimate any kind of probabilities associated with the possible future scenarios. A set of solutions is calculated, instead of only one. The DM will thus have a much broader view of the compromises that exist among the possible scenarios. Many of the classic robust solutions, like minimizing the worst result or the maximum regret can be considered as special cases. Making available to the DM a set of solutions will make it possible to apply other robustness measures that are more difficult to incorporate directly in a mathematical programming problem (like, for instance, the *bw*-robustness measure [20]).

The major drawback of the described approach has to do with the fact that it implies the sequential optimization of NP-hard problems. Depending on the problem's dimension, this can be computationally expensive. One way of circumventing the problem is through the use of efficient heuristics in a first stage, and the use of exact procedures in a second phase where the preferred region is already delineated. Another possibility is the use of metaheuristics that work with populations of solutions (like genetic algorithms) and that are capable of generating sets of efficient solutions in each generation (see, for instance, [21]).

In the problem considered, any solution that is admissible for one given scenario will be admissible for all others. This situation does not always occur. Defining the concept of Pareto-efficiency for single objective dynamic location problems where feasibility can also be affected by uncertainty is an interesting path of research.

Although we have applied this approach to a simple dynamic location problem under uncertainty, it can easily be generalized to many other linear, integer or mixed-integer programming problems.

References

1. Owen, S.,Daskin, M.: Strategic Facility Location: A Review. European Journal of Operational Research. 111, 423-447 (1998)
2. Mulvey, J.M., Vanderbei, R.J.,Zenios, S.A.: Robust optimization of large-scale systems. Operations research. 43, 264-281 (1995)
3. Rieger, M.O., Wang, M.,Hens, T.: Risk Preferences Around the World. Management Science. Published online in Articles in Advance 21 Feb 2014, (2014)
4. Snyder, L.V.: Facility location under uncertainty: a review. IIE Transactions 38, 537–554 (2006)
5. Hulsurkar, S., Biswal, M.P.,Sinha, S.B.: Fuzzy programming approach to multi-objective stochastic linear programming problems. Fuzzy Sets and Systems. 88, 173-181 (1997)

6. Teghem Jr, J., Dufrane, D., Thauvoye, M.,Kunsch, P.: STRANGE: an interactive method for multi-objective linear programming under uncertainty. European Journal of Operational Research. 26, 65-82 (1986)

7. Urli, B.,Nadeau, R.: PROMISE/scenarios: An interactive method for multiobjective stochastic linear programming under partial uncertainty. European Journal of Operational Research. 155, 361–372 (2004)

8. Abdelaziz, F.B.: Solution approaches for the multiobjective stochastic programming. European Journal of Operational Research. 216, 1-16 (2012)

9. Gutjahr, W.J.: Two metaheuristics for multiobjective stochastic combinatorial optimization. In: O. Lupanov, O.M. Kasim-Zade, A.V. Chaskin, K. Steinhöfel (eds) Stochastic Algorithms: Foundations and Applications. LNCS, vol. 3777, pp. 116-125. Springer, (2005)

10. Guillén, G., Mele, F.D., Bagajewicz, M.J., Espuna, A.,Puigjaner, L.: Multiobjective supply chain design under uncertainty. Chemical Engineering Science. 60, 1535-1553 (2005)

11. Cardona-Valdés, Y., Álvarez, A.,Ozdemir, D.: A bi-objective supply chain design problem with uncertainty. Transportation Research Part C. 19, 821–832 (2011)

12. Iancu, D.A.,Trichakis, N.: Pareto Efficiency in Robust Optimization. Management Science. 60, 130-147 (2014)

13. Lamboray, C.,Vanderpooten, D.: Lexicographic alfa-robustness: An alternative to min-max criteria. European Journal of Operational Research. 220, 722-728 (2012)

14. Dias, J., Captivo, M.E.,Clímaco, J.: An interactive procedure dedicated to a bicriteria plant location model. Computers & Operations Research. 30, 1977-2002 (2003)

15. Mladineo, N., Margeta, J., Brans, J.P.,Mareschal, B.: Multicriteria Ranking of Alternative Locations for Small Scale Hydro Plants. European Journal of Operational Research. 31, 215-222 (1987)

16. Barda, O.H., Dupuis, J.,Lencioni, P.: Multicriteria Location of Thermal Power Plants. European Journal of Operational Research. 45, 332-316 (1990)

17. Ross, T.,Soland, R.: A Multicriteria Approach to the Location of Public Facilities. European Journal of Operational Research. 4, 307-321 (1980)

18. Lim, G.J.,Sonmez, A.D.: Y-Robust Facility Relocation Problem. European Journal of Operational Research. 229, 67-74 (2013)

19. Marques, M.d.C.,Dias, J.: Simple dynamic location problem with uncertainty: a primal-dual heuristic approach. Optimization. 62, 1379–1397 (2013)

20. Roy, B.: Robustness in Operational Research and Decision Aiding: a multi-faceted issue. European Journal of Operational Research. 200, 629-638 (2010)

21. Dias, J., Captivo, M.E.,Clímaco, J.: A Memetic Algorithm for Multi-objective Dynamic Location Problems. Journal of Global Optimization. 42, 221-253 (2008)