Article:
Proll, LG, Salhi, A and Rios Insua, D (2001) Improving an optimization-based framework for sensitivity analysis in multi-criteria decision-making. Journal of Multi-Criteria Decision Analysis, 10 (1). pp. 1-9. ISSN 1057-9214

https://doi.org/10.1002/mcda.283

Improving an Optimisation-Based Framework for Sensitivity Analysis in Multi-Criteria Decision Making

L.G.Proll[+], A.Salhi[†], and D.Rios Insua[*]

[+] Constraint Programming and Operational Research Group, School of Computing, University of Leeds, UK

[†] Dpt. of Mathematics, University of Essex, UK

[*] Dpt. de Ciencias Experimentales e Ingenieria, Universidad Rey Juan Carlos, Spain

Abstract

The framework for sensitivity analysis in discrete multi-criteria decision analysis developed by Rios Insua and French allows simultaneous variation of all parameters and applies to many paradigms for decision analysis. However its computational load may inhibit use, particularly in the context of a decision conference where results are required in near real time. In order to improve on the current algorithm and its implementation, we investigate, on the one hand, an opportunistic approach aimed at reducing the number of optimisation problems solved in the original framework and, on the other, an alternative framework based on distance analysis. Computational results on linear and bilinear models are reported.

KEYWORDS: multi-criteria decision analysis, sensitivity analysis

1 Introduction

Multi-criteria decision models rely on two forms of numeric input: objective data defining the physical aspects of the alternatives, states and consequences of the decision model, and judgemental data relating to the decision-maker(DM)'s beliefs and preferences. In this context, sensitivity analysis allows exploration of the effects of variations in judgemental input on the ranking of alternatives. However it also plays the role of an aid to the elicitation process of decision analysis by focussing discussion and reflection on the judgemental data, (French, 1992).

Rios Insua and French (1991) have developed a conceptual framework for sensitivity analysis in multi-criteria decision making with a discrete set of alternatives which allows simultaneous variation of judgemental data and which applies to many paradigms for decision analysis. Extension of this framework to the case of a continous set of alternatives is discussed in (Rios Insua *et al*, 1997) and its description from a statistical decision theory perspective is given in (French, 1995; Rios Insua *et al*, 2000).

The framework is largely based on mathematical programming and involves solving a potentially large number of mathematical programmes of various types, some of which are nonlinear and nonconvex. Consequently the computational load may inhibit use of the framework particularly when decision analyses are performed in the context of decision conferences, where it is desirable for sensitivity analyses to be conducted in near real time, preferably on a PC. In principle variations in the objective data could also be handled within the framework but our principal concern is with judgemental data, which is inherently more uncertain and whose examination aids problem understanding.

An alternative approach to sensitivity analysis in the multi-criteria decision making context is through simulation (see, fror example, Butler *et al*, 1997). Simulation approaches may also suffer computational problems if constraints on the values of the judgemental data of the form envisaged in (Rios Insua, 1990) are present due to the difficulty of sampling in continuous spaces defined by general constraints (Rinnooy Kan and Timmer, 1986).

The present work is an attempt to address the question of computational load in the Rios Insua-French framework through two avenues:

1. Given that many of the problems in the filtering process of the original algorithm are existence problems rather than optimisation problems, use an opportunistic approach to exploit this observation and reduce the overall number of optimisation problems to be solved.

2. Modify the basic algorithm for sensitivity analysis given in (Rios Insua, 1990) to concentrate on distance analysis, through which immediate contenders for optimality are detected.

Note that a parallel processing approach to handling the computational load has been investigated previously, (Salhi *et al*, 1995). Results with parallel implementations will be discussed.

This paper is organised as follows. Section 2 recalls the concept of the original algorithm for sensitivity analysis. Section 3 discusses implementations of the original framework, including parallel implementations which were considered in the past as an obvious way of handling computational load. In Section 4 the alternative framework is given. Section 5 discusses an opportunistic approach to reducing the number of optimisation problems. Section 6 is a conclusion and discussion of the experimental results obtained with the two approaches.

2 The Original Algorithm Concept

As perceived in (Rios Insua, 1990), sensitivity analysis is a crucial step of the decision making iterative process. This step filters the alternatives through four phases:

- The dominance (D) phase;
- The potential optimality (PO) phase;
- The adjacent potential optimality (APO) phase;
- The distance analysis (DA) phase.

A detailed algorithm for sensitivity analysis can be found in (Proll *et al*, 1993). It is phrased in terms of two major problem descriptors: a function $\psi_j(\omega)$ which evaluates alternative j for a given vector, ω, of judgemental data values and a set S delimiting the possible values of ω. S may comprise such conditions as monotonicity of utilities, coherence of probabilities and normalisation of weights, in addition to the decision maker's judgements on the likely ranges of values of the various parameters. The questions posed in the above phases are all formulated as constrained optimisation problems. The classes of mathematical programme requiring solution vary between phases and depend on the form of the evaluation function, $\psi_j(\omega)$, for the decision model adopted, the nature of the set S and, in the distance analysis phase, on the metric chosen. Given an initial estimate, ω^0, of ω elicited by the decision analyst, a current best alternative can be identified. The first three phases above identify the competitors of this alternative, i.e. alternatives which may become optimal as ω changes away from ω^0. The distance analysis phase identifies, for each metric, the nearest competitor and provides an index of the sensitivity of the

decision, thus giving a focus to further discussion and elicitation. Suppose p is the number of alternatives under consideration and assume that the ranking of these alternatives at ω^0 is strict, then, in the worst case, the D phase requires solution of $p(p-1)/2$ problems, the PO and APO phases require solution of $(p-1)$ problems and the DA phase requires the solution of p problems for each metric. Each of these problems is of size proportional to n, the number of judgemental parameters. If the initial ranking is not strict, additional problems may need to be solved in the D phase.

The algorithm outlined above is embedded naturally in a cycle of modelling, optimisation and sensitivity analysis until the model is requisite (Phillips, 1984). This implies that the mathematical programmes may need to be generated and solved several times. Thus implementation of the Rios Insua framework requires care as the computational load is potentially heavy.

2.1 Optimisation Issues

In the linear model, all mathematical programmes are linear (or transformable to linear) with the exception of the minimum L_2 distance problem, which is a convex quadratic programme, and the maximum L_2 and L_∞ distance problems, which are nonconvex. For the bilinear model, the formulations in (Rios Insua, 1990) yield nonlinear and, in general, nonconvex programmes for all problems except the maximum L_∞ distance problem which is solvable by linear programming, (Proll and Salhi, 1994). For the general model, all problems are nonlinear programmes and are potentially nonconvex. Thus many of the problems which we have to solve may have multiple local optima. Use of a local optimiser only could convey a false impression of insensitivity. For example, consider the bicriteria decision problem given in Table 1 the alternatives of which are described by their component utilities.

	States of Nature	
Alternative	θ_1	θ_2
a1	(1,1)	(1,1)
a2	(2,1)	(1,0)
a3	(5/2,3/2)	(3/2,1/2)
a4	(0,2)	(0,0)
a5	(1,1)	(2,0)
a6	(71/36,35/36)	(35/36,35/36)
a7	(73/72,73/72)	(73/72,73/72)

Table 1: A Bicriteria Decision Problem

Let the probability of states θ_1, θ_2 be p_1, p_2 respectively and let the weights assigned to the two criteria be λ_1, λ_2 respectively. Then we know that:

$$\begin{aligned} \lambda_1 \ + \ \lambda_2 \ &= \ 1 \\ p_1 \ + \ p_2 \ &= \ 1 \\ \lambda_i, \ p_i \ &\geq \ 0 \end{aligned}$$

Letting

$$\begin{aligned} \lambda_1 &= \lambda, \lambda_2 = 1-\lambda \\ p_1 &= p, \; p_2 = 1-p \end{aligned}$$

the evaluation of an alternative described by ((a,c),(b,d)) is

$$p\lambda(a-c-b+d)+(c-d)p+(b-d)\lambda+d$$

Therefore

$$\begin{aligned} \psi_1(\lambda,p) &= 1 \\ \psi_2(\lambda,p) &= \lambda+p \\ \psi_3(\lambda,p) &= \lambda+p+1/2 \\ \psi_4(\lambda,p) &= -2p\lambda+2p \\ \psi_5(\lambda,p) &= -2p\lambda+p+2\lambda \\ \psi_6(\lambda,p) &= p\lambda+35/36 \\ \psi_7(\lambda,p) &= 73/72 \end{aligned}$$

Suppose that we have been able to elicit the following judgements from the DM,

$$\begin{aligned} a_2 &\preceq a_1 \\ a_1 &\preceq a_3 \\ a_4 &\preceq a_5 \end{aligned}$$

Then we deduce that:

$$\begin{aligned} \psi_2(\lambda,p) \leq \psi_1(\lambda,p) &\implies p+\lambda \leq 1 \\ \psi_1(\lambda,p) \leq \psi_3(\lambda,p) &\implies p+\lambda \geq 1/2 \\ \psi_4(\lambda,p) \leq \psi_5(\lambda,p) &\implies p \leq 2\lambda \end{aligned}$$

To check whether a_6 dominates a_7, we have to solve:

$$\begin{array}{lllll} \min & p\lambda-3/72 & & \\ \text{s.t.} & p+\lambda & \leq & 1 \\ & p+\lambda & \geq & 1/2 \\ & p-2\lambda & \leq & 0 \\ & 0 \leq & \lambda, p & \leq 1 \end{array}$$

This problem has two local optimal solutions at (1/2, 0) and (1/6, 2/6) with values -3/72 and 1/72 respectively. If the optimisation routine returns the first of these, we conclude that a_6 does not dominate a_7; if it returns the second, we conclude that a_6 dominates a_7. Clearly the filtering phase may then result in different sets of alternatives being submitted to the distance analysis phase, and hence to differing values of the sensitivity index.

Thus to provide reliable sensitivity information we need to provide global optimisation routines. It is important also to note that, of necessity, we have to solve the problems generated by the sensitivity analysis framework unseen. Further, although methods are available for some classes of problem of interest to us, e.g. the

maximum L_2 distance problem which is a concave quadratic programme, (Pardalos and Rosen, 1987), it is impractical from a software maintenance viewpoint to implement a different algorithm for each of the many classes of problem which arise. Ideally we need a robust, quick general-purpose method. Stochastic methods for constrained global optimisation (Schoen, 1991) would appear to offer the best hope. However, despite the development of many such methods, experimental evidence of their effectiveness is patchy.

3 Prototype Implementations

3.1 Sequential implementation

Rios Insua (1990) constructed a prototype implementation in Fortran to test the framework for each of three types of decision model:

1. Linear, in which $\psi_j(\omega)$ is linear and S is defined by linear constraints. Such models may arise, for example, in multi-attribute value methods when there is imprecision in the weights.

2. Bilinear, in which $\psi_j(\omega)$ is bilinear and S is defined by linear constraints. Such models may arise, for example, in multi-attribute value methods when there is imprecision in both the weights and the scores.

3. General, in which $\psi_j(\omega)$ and S have general form. Such models may arise, for example, from decision trees, influence diagrams and multiattribute utility models with imprecision in both utilities and probabilities.

The programme ran on an Amdahl 5860 mainframe and required additional code for each specific problem instance. Distance analysis was supported in any L_p metric. All constituent mathematical programmes were formulated in a smooth manner, (Fletcher, 1987), and were solved using optimisation routines from Chapter E of the NAG library. Only local optima were sought for the nonconvex programmes.

As an indication of the computational load imposed by sensitivity analysis, Rios Insua (1990) quotes an instance of a linear model involving 8 alternatives and 10 criteria which required the solution of 54 linear programmes, 4 convex quadratic programmes, 1 nonconvex quadratic programme and 1 nonconvex nonlinear programme with distance analysis performed in the L_1, L_2, and L_∞ metrics.

Our development of the prototype implementation addressed the question of whether the Rios Insua-French framework is viable within the environment of a decision conference, i.e.

a) can the computational load be reduced sufficiently to allow analysis of models of realistic size in near real-time on a PC?

b) can software implementing the framework be made sufficiently general that no on-site coding is required by the decision analyst and that communication with commercially available decision aiding packages is possible?

c) can sufficiently robust optimisation routines be developed to enable the sensitivity analysis software to be reliable?

Issues a) and c) are connected and have been approached from a number of directions including exploitation of parallelism within the framework, choice of local and global optimisation algorithms for particular subproblems and alternative formulations of such subproblems. Issue c) also is important since the sensitivity analysis algorithm, and hence the optimisation problems underpinning it, appears as a 'black box' to the decision analyst. Issue b) implies the construction of problem generators for the various subproblems arising in the framework and also depends on the choice of formulation of such problems.

Our implementation effort has been concentrated on linear and bilinear models because they are more likely to arise in a decision conference and because they offer the best hope of successful implementation. For such models, issue b) is readily answered in the affirmative as all problem data can be expressed in matrix form. However, much of what is reported here, with the exception of the global optimisation issue, is applicable to the general model. Distance analysis is restricted to the L_1, L_2, and L_∞ metrics.

3.2 Parallel implementation

The general algorithm for sensitivity analysis is inherently sequential due to the precedence relationships between the phases. However, parallelism is present within each phase as each requires the solution of a set of independent problems, one for each (or each pair of) alternative. Given the potentially large number of mathematical programmes to be solved, large grain parallelisation, in which each optimisation is considered as a single task, seems appropriate. We adopt the processor farm approach (Fox *et al*, 1988) on a network of processor nodes. Individual mathematical programmes arising at the level of each phase are farmed out to individual nodes by a master process which resides on the root node. The worker process is replicated on each node of the network. The master process generates the problems, transmits them to the network and collects and displays the results. The worker process communicates with the master and solves mathematical programmes. Three implementations of this approach have been constructed on differing platforms, a Meiko Computing Surface using CSTools, a PC enhanced with a transputer board using 3L Parallel Fortran and a network of Sun workstations using PVM. Fuller details of the implementations are given in (Proll *et al*, 1993a, 1993b) and (Salhi *et al*, 1995). In all cases, good speedup was achieved within each phase, but the overall speedup on a 5 node network was approximately 2.5 for linear models and 3 for bilinear models. Essentially, this is due to the high ratio of communication time to computation time as the mathematical programmes are small and relatively easy to solve and also because of the precedence relationships between phases. The parallel approach may be more fruitful for general models in which the mathematical programmes are more difficult to solve.

4 Reducing the Number of Optimisation Problems

The principal concern in the computational load is the number and difficulty of the mathematical programmes which need to be solved. In this section, we concentrate on reducing the number of problems which need to be solved in the D and PO phases and in one component of the DA phase. In the D phase we pose the question : Does alternative j dominate alternative k? via the mathematical programme:

$$\begin{aligned} & \min \quad \psi_j(\omega) - \psi_k(\omega) \\ \text{s.t.} \quad & \omega \in S \end{aligned} \tag{1}$$

We need to solve up to $p(p-1)/2$ such problems, where p is the number of alternatives, if the ordering is strict and possibly more if it is not. In all other phases the number of problems to be solved is linear in p, e.g. in the potential optimality phase, we solve for each non-dominated alternative j, the mathematical programme:

$$\begin{aligned} \min \quad & \max\{\psi_j(\omega) - \psi_k(\omega) : k \neq j\} \\ \text{s.t.} \quad & \omega \in S \end{aligned} \tag{2}$$

However we may observe that

- alternative j cannot be dominated by alternative k if there exists $\omega \in S$ such that $\psi_j(\omega) > \psi_k(\omega)$;
- alternative j is potentially optimal if there exists $\omega \in S$ such that $j = arg\max_k \psi_k(\omega)$.

Thus observation of points at which such conditions occur removes the need to solve the corresponding optimisation problem. How do we generate a set of test points? Our solution is an opportunistic one, i.e. each time we obtain the optimal solution to a dominance problem, we test the above conditions at that point for each (pair of) alternative(s) for which the dominance/potential optimality position is not yet resolved. Random sampling is an obvious alternative but is not used for the following reasons:

- generating random points in a convex polytope takes a non-trivial amount of time, generating random points in a nonconvex set even more so;
- initial tests showed that the sample size required to give similar results to the opportunistic strategy made random sampling uncompetitive in time;
- dominated and non-potentially optimal alternatives cannot be identified by observation at a point so the solution of some optimisation problems is still likely to be required.

Table 2 below shows the success of this strategy on a set of linear and bilinear problems; problems 1 - 6 are linear, problems 7 - 11 are bilinear. Problems 1, 2 and 5 arise from decision conferences on radiation protection and computer selection conducted by Simon French. Problems 3, 8, 10 and 11 are taken from the flood-plain management, portfolio selection and road selection cases in (Rios Insua, 1990). Problem 4 is taken from (Phillips, 1988). Problem 7 is taken from (French and Rios Insua, 1989). Problem 9 arises in the location of power plants (Barda *et al*, 1990).

	Original Version		Opportunistic Strategy	
Problem	D Problems	PO Problems	D Problems	PO Problems
1	3	1	3	0
2	20	4	5	0
3	17	5	8	0
4	5	1	5	0
5	17	5	7	0
6	26	6	10	0
7	3	2	1	1
8	8	3	5	0
9	10	3	5	2
10	21	6	2	4
11	25	6	7	4
Total	153	42	58	11

Table 2: Effect of the Opportunistic Strategy on the Number of Optimisations Performed

Thus, as Table 2 shows, the opportunistic strategy affords a substantial reduction in the number of optimisation problems solved. This is achieved at negligible additional cost, requiring only the evaluation function for each non-dominated alternative to be computed.

Further reduction can be achieved in respect of one of the problems arising in the distance analysis phase, the maximum L_∞ distance problem, i.e.

$$\begin{aligned} \max \quad & \max_{1 \le i \le n} |\omega_i - \omega_i^0| \\ \text{s.t.} \quad & \omega \in S \end{aligned} \tag{3}$$

where n is the number of parameters and ω_i^0 are known constants, the initial estimates of these parameters.

Rios Insua (1990) gives an algorithm for this problem which requires the solution of $2n$ subproblems. For linear and bilinear models, these are (small) linear programmes and are not very costly to solve but, for general models, the subproblems are nonlinear and possibly nonconvex, thus requiring global optimisation. Proll and Salhi (1994), show empirically, by using simple bounding ideas, that the number of subproblems needing to be solved can be reduced by 60-95% on a set of test cases. As, in each maximum L_∞ distance problem, the subproblems are all of the same size and structure, time savings of the same order are realised.

5 An Alternative Framework

The D phase involves pair-wise comparison of the alternatives and thus may require the solution of $O(p^2)$ optimisation problems. This is the dominant term in the total number of problems to be solved. Although it is useful to know the set of alternatives which are nondominated, one can argue that it is those nondominated alternatives which are immediate contenders for optimality which are of 'real' interest to the analyst. These candidates for optimality can be detected using distance tools, thus avoiding a systematic check for dominance. This idea is translated into the following algorithm, on which the alternative framework for sensitivity analysis is based.

Algorithm

1. Rank alternatives according to $\psi_\cdot(\omega^0)$;

2. Consider the first ranking alternative as current optimum, a_*, if it is nondominated. (Note that the first ranking alternative is non-dominated if it does not tie with another alternative. If there are several alternatives which tie for first ranking, at least one will be (strictly) PO, in which case that one can be chosen, or the DM is indifferent to these alternatives, in which case all but one of them can be deleted.)

3. Find least changes leading to it being outranked by other alternatives a_j. This can be done through the solution of distance problems of the form:

$$\begin{array}{rll} d_j = & \min & d(\omega, \omega^0) \\ \text{s.t.} & & \psi_j(\omega) - \psi_*(\omega) = 0 \\ & & \omega \in S \end{array} \tag{4}$$

   where $d(.,.)$ is some continuous metric;

4. Rank alternatives according to minimum distance;

5. Find first alternative in this ranking which is nondominated. This is the nearest competitor of a_*.

The above algorithm only checks for dominance the (one or more) 'nearest competitors' of the current optimal alternative. The price paid for avoiding the optimisations associated with the D, PO and APO phases however is an increase in the number of optimisations in the DA phase. The questions posed in the D, PO and APO phases can frequently be answered by local optimisation only since they are fundamentally existence problems rather than inherently optimisation problems. For instance, suppose we are testing whether alternative j dominates alternative k and that a local minimum of $\psi_j(\omega) - \psi_k(\omega)$ subject to $\omega \in S$ has value < 0, then clearly k cannot be dominated by j and it is unnecessary to find the global minimum of $\psi_j(\omega) - \psi_k(\omega)$. Thus our implementation performs a local optimisation first and only performs global optimisation if necessary in each problem arising in

the D, PO and APO phases. DA problems however always require global optimisation and generally are more costly to solve. Thus, in addition to not providing as much information to the analyst, the alternative algorithm turns out to be worse computationally than the original one, as evidenced in Table 4 and Table 5.

5.1 Experimental Results with the Alternative Algorithm

A Fortran code for the algorithm has been run on the 5 bilinear models, Problems 7 - 11 of Table 2. Relevant statistics for these problems can be found in Table 3. #D, #PO, #APO are the number of nondominated, potentially optimal and adjacent potentially optimal alternatives respectively.

Problem	n	p	#D	#PO	#APO
7	5	3	3	2	1
8	9	6	4	4	3
9	14	6	4	3	2
10	12	7	7	5	4
11	20	8	7	7	6

Table 3: Bilinear models: statistics

Results are of two types. In Table 4 the code has been run with the option to consider all optimisation problems as seeking to find a local optima only. In Table 5 it has been run with the option to solve the problems in the global sense. The difference in the recorded CPU times for the two options is clear.

Problem	Original	Alternative
7	3.02	4.07
8	7.80	9.07
9	45.27	36.81
10	20.11	26.43
11	175.06	129.73

Table 4: Bilinear models: Local Optimisation Mode

Problem	Original	Alternative
7	123.75	184.36
8	224.89	358.24
9	1409.56	2335.88
10	589.12	1322.86
11	5245.88	> 7000

Table 5: Bilinear models: Global Optimisation Mode

6 Global Optimisation and Reformulation

In section 2.1, we remarked that a proper implementation of the framework for all models requires a robust, yet quick, general purpose method for constrained global optimisation. It is doubtful if currently available algorithms satisfy this requirement for problems of the size we need to solve and in the context in which many of our applications need to be tackled. However we have developed a simulated annealing method (Salhi *et al*, 2000) which has successfully solved problems occurring in the linear and bilinear models. Its success arises from exploiting the structure of the problem constraints to ensure that random neighbours of the current point are always feasible. Suppose we are optimising over the set $A = \{\mathbf{x} : \mathbf{a_j} * \mathbf{x}\ (\leq, =)\ b_j, j = 1, 2, ..., m\}$ we use the following routine:

1. Find a feasible point $\mathbf{x} \in A$.

2. Generate a direction vector $\mathbf{v}$ with equal probability from one of the n coordinate vectors, i.e. generate a random index k in 1,...,n. Set $v_k = 1$, $v_j = 0$ for $j \neq k$ from one of the n coordinate vectors, i.e. generate a random index k in 1,...,n. Set $v_k = 1$, $v_j = 0$ for $j \neq k$

3. If x_k has non-zero coefficient in equality constraint i, generate a random index l, $(l \neq k)$ in 1,...,n such that x_l has non-zero coefficient in constraint i.
   Set $v_l = -a_{ik}/a_{il}$.

4. For each inequality j, compute $\lambda_j = (b_j - \mathbf{a_j} * \mathbf{x})/(\mathbf{a_j} * \mathbf{v})$

5. If $\exists j$ such that $\lambda_j = 0$ and $\mathbf{a_j} * \mathbf{v} > 0$, set $\lambda^+ = 0$ else $\lambda^+ = \min\{\lambda_j : \lambda_j > 0\}$
   If $\exists j$ such that $\lambda_j = 0$ and $\mathbf{a_j} * \mathbf{v} < 0$, set $\lambda^- = 0$ else $\lambda^- = \max\{\lambda_j : \lambda_j < 0\}$.

6. Generate u from a uniform distribution on [0,1] and set
   $\mathbf{y} = \mathbf{x} + (\lambda^- + u(\lambda^+ - \lambda^-))\mathbf{v}$.

The scheme above guarantees that $\mathbf{y} \in A$ provided that any equations present in the linear constraint set A involve mutually exclusive sets of variables. The set S delimiting the parameters may involve equations only for the normalisation of weights and the coherence of probabilities and thus has this structure. For the D, PO and maximum distance problems (see (1), (2), (3) respectively), A is exactly S. During other phases of the sensitivity analysis constraints involving the evaluation functions are added which may destroy this property. This does not cause any difficulty for the linear model but does so in the APO (5) and minimum distance problems(4) when the evaluation function is bilinear.

In (Rios Insua, 1990) the APO problem for alternative j is formulated as:

$$\begin{array}{rc} \max & \psi_j(w) - \psi_*(w) \\ \text{s.t.} & w \in S \\ & \psi_j(w) - \psi_*(w) \leq 0, \forall i \end{array} \tag{5}$$

That is, we try to find a point at which alternatives a_j and a_* are jointly optimal. We can also do this by solving:

$$\begin{array}{ll} \min & ((\psi_j(w) - \psi_*(w))^2, \max\{\psi_i(w) - \psi_*(w) : i \neq j\}) \\ \text{s.t.} & w \in S \end{array}$$

a_j is a.p.o. if the optimal value of the mathematical programme is 0. In this form the APO problem now has the required structure for the simulated annealing method.

The minimum distance problems do not have the required structure because of the single nonlinear constraint $\psi_j(\omega) - \psi_*(\omega) = 0$. We deal with this by taking this constraint into the objective function using an exact penalty function (Fletcher, 1987) of the form:

$$\begin{array}{rl} \min & d(\omega, \omega^0) - p * min(0, \psi_j(\omega) - \psi_*(\omega)) \\ \text{where} & p = d(x, \omega_*)/(\psi_j(x) - \psi_*(x)) \end{array}$$

and x is a point at which a_j is potentially optimal.

7 Conclusions

The improvements to the sensitivity analysis algorithm described in this paper, together with technological improvements in the computing environment, make analysis of the linear and bilinear models viable in near real-time. Complete analyses of each of the problems using all three distance metrics, L_1, L_2, L_∞ can be performed in less than two minutes on a 233 MHz Pentium II processor. Analysis of general models remains much more problematic due both to the difficulty of specifying the model to a general package and to the need for a suitable global optimisation method, since the method described here for the linear and bilinear models is not applicable. Constrained global optimisation is a vibrant research area, which offers us some hope here. However it is perhaps less likely that such models would arise in a context which required results in near real-time.

Acknowledgements

We are grateful to Professor Simon French for his helpful discussions and provision of data. We are happy to acknowledge funding for this research by Engineering and Physical Sciences Research Council(UK) Grant GR/F45141, by CICYT(Spain), by NATO Collaborative Research Grant CRG910546 and by British Council/MEC Acciones Integradas Grant 667.

References

Barda, O.H., Dupuis, J. and Lenciono, P. Multicriteria location of thermal power plants, *European Journal of Operations Research*, **45**, 332-346 (1990).

Butler, John, Jia, Jianmin and Dyer, James, Simulation techniques for the sensitivity analysis of multi-criteria decision models, *European Journal of Operations Research*, **103**, 531-546 (1997).

Fletcher, R. *Practical Methods of Optimization*, 2nd edition, John Wiley & Sons, Chichester (1987).

Fox, G., Johnson, M., Lyzenga, G., Otto, S, Salmon, J. and Walker, D. *Solving Problems on Concurrent Processors*, Prentice-Hall, Englewood Cliffs, NJ (1988).

French, S., The role of sensitivity analysis in decision analysis, in Holtham, C. (editor) *Executive Information Systems and Decision Support*, 99-123, Chapman and Hall, London (1992).

French, S., The framing of statistical decision theory: a decision analytic view, in Bernardo,J. M., Berger, J. O., Dawid, A. P. and Smith, A. F. M. (editors) *Bayesian Statistics 5*, 147-164, Oxford University Press, Oxford.

French, S. and Rios Insua, D. Partial information and sensitivity analysis in multi-objective decision-making, in Lockett, A.G. and Islei, G. (editors) *Improving Decision Making in Organisations*, 424-433, Springer Verlag, Berlin (1989).

Pardalos, P.M. and Rosen, J.B., *Constrained Global Optimization: Algorithms and Applications*. Springer Verlag, Berlin (1987).

Phillips, L.D., A theory of requisite decision models, *Acta Psycholgica*, **56**, 29-48 (1984).

Phillips, L.D., People-centred group decision support, in Doukidis, G. Land, F. and Miller, G. (editors) *Knowledge Based Management Systems*, Ellis Horwood, Chichester (1988).

Proll, L.G., Rios Insua, D. and Salhi, A., Mathematical programming and the sensitivity of multi-criteria decisions. *Annals of Operations Research*, **43**, 109-122 (1993).

Proll, L.G. and Salhi, A., A note on a maximum distance problem, *Computers and Operations Research*, **21**, 1069-1072 (1994).

Proll, L.G. Salhi, A. and Rios Insua, D., A parallel implementation of a mathematical programming framework for sensitivity analysis in MCDM. Presented at *APMOD93 - Applied Mathematical Programming and Model ling*, Budapest (1993).

Rinnooy Kan, A. H. G. and Timmer, G. T., The multi-level single linkage method for unconstrained and constrained global optimization, in Griffiths, D.F. and Watson, G. A. (editors), *Numerical Analysis*, 173–186. Longman Scientific & Technical, Harlow (1986).

Rios Insua, D., *Sensitivity Analysis in Multi-objective Decision making*, Springer Verlag, Berlin (1990).

Rios Insua, D. and French, S., A framework for sensitivity analysis in discrete multi-objective decision making, *European Journal of Operations Research*, **54**, 176-190 (1991).

Rios Insua, David, Martin, Jacinto, Proll, Les, French, Simon and Salhi, Abdellah, Sensitivity analysis in statistical decision theory: a decision analytic view, *Journal of Statistical Computation and Simulation*, **57**, 197-218 (1997).

Rios Insua, David, Ruggeri, Fabrizio, and Martin, Jacinto, Bayesian Sensitivity Analysis, in Saltelli, A., Chan, K. and Scott, E. M. (editors) *Sensitivity Analysis*, 225-244, John Wiley and Sons, Chichester (2000).

Salhi, A., Proll, L.G., Rios Insua, D. and French, S., Implementation of a framework for sensitivity analysis in multi-criteria decision-making on a cluster of workstations, *University of Edinburgh, Department of Business Studies Working Paper Series* 95/5 (1995).

Salhi, A., Proll, L.G., Rios Insua, D. and Martin, J., Experiences with stocahstic methods for a class of global optimisation problems, *RAIRO Operations Research*, **34**,183-197 (2000).

Schoen,F., Stochastic techniques for global optimization: a survey of recent advances, *Journal of Global Optimization*, **1**, 207-228 (1991).