



ELSEVIER

European Journal of Operational Research 138 (2002) 332–348

EUROPEAN
JOURNAL
OF OPERATIONAL
RESEARCH

www.elsevier.com/locate/dsw

An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI

Luís Dias ^{a,b,*}, Vincent Mousseau ^c, José Figueira ^b, João Clímaco ^{a,b}

^a Instituto de Engenharia de Sistemas e Computadores – Coimbra, Rua Antero de Quental, 199, 3000-033 Coimbra, Portugal

^b Faculdade de Economia, Universidade de Coimbra, Av. Dias da Silva, 165, 3004-512 Coimbra, Portugal

^c LAMSADE, Université Paris-Dauphine, Place du Maréchal De Lattre de Tassigny, 75775 Paris Cedex 16, France

Abstract

ELECTRE TRI is a well-known method to assign a set of alternatives to a set of predefined categories, considering multiple criteria. Using this method requires setting many parameters, which is often a difficult task. We consider the case where the decision makers (DMs) in the decision process are unsure of which values should each parameter take, which may result from uncertain, imprecise or inaccurately determined information, as well as from lack of consensus among them. This paper discusses the synergy between two approaches developed independently to deal with this difficulty. The first approach infers the value of parameters from assignment examples provided by the DMs, as an elicitation aid. Each assignment example originates mathematical constraints that the parameter values should satisfy. The second approach considers a set of constraints on the parameter values reflecting the imprecise information that the DMs are able to provide. Then, it computes the best and worst categories for each alternative compatible with constraints, in order to present robust conclusions. Both approaches avoid asking for precise values for the parameters. Rather, they proceed to solve the problem in a way that requires from the DMs much less effort. By integrating these two approaches, this paper proposes a new interactive approach, where the insight obtained during robustness analyses guides the DMs during the elicitation phase. © 2002 Published by Elsevier Science B.V.

Keywords: Multiple criteria analysis; Sorting problems; Aggregation/disaggregation; Robustness and sensitivity analysis

1. Introduction

Many real world decision problems can be formalised using a multiple criteria approach, i.e.,

by defining a set of criteria evaluating the alternatives' performances. This approach requires the decision makers (DMs) to provide some preference information in order to build a model that evaluates the alternatives. Eliciting preference information from the DMs and formalising it through preferential parameters is hence a crucial phase in a multiple criteria decision aid model. In the various aggregation procedures, these parameters take the form of weights, aspiration levels,

* Corresponding author.

E-mail addresses: ldias@inescc.pt (L. Dias), mousseau@lamsade.dauphine.fr (V. Mousseau), figueira@fe.uc.pt (J. Figueira), jclimaco@inescc.pt (J. Clímaco).

thresholds, etc. The values assigned to these parameters will determine how the evaluation of the alternatives on the different criteria should be combined.

However, DMs have difficulties in defining precise values for preferential parameters. There are various reasons for such problems concerning preference elicitation: the data considered in the decision problem might be imprecise or uncertain; DMs may have a vague understanding of what the parameters represent and their point of view can evolve during the elicitation process. Moreover, in group decisions, lack of consensus among DMs can be also a critical issue.

Some authors (Mousseau, 1995; Bana e Costa and Vansnick, 1994; Jacquet-Lagrèze and Siskos, 1982; Belton and Vickers, 1990) have proposed preference elicitation techniques (PETs) to support the analyst in assigning values to preferential parameters. Usually PETs proceed indirectly through a questioning procedure and “translate” the DMs’ answers into values for the preferential parameters (by applying the specific aggregation rule in use).

Such PETs often may provide the analyst with several acceptable combinations of values for the parameters, although they try to select a single combination according to some measure of appropriateness. If we consider all those acceptable combinations of values for the parameters, then one idea is to find conclusions that are valid for all of (or most of) these combinations. Such a study concerning the impact of the values of preferential parameters on the overall preferences is called robustness analysis (see Roy, 1998), typically conducted as the last phase of a decision aid process.

It seems obvious that these two stages of a decision aiding process may interact. Robustness analysis uses a set of acceptable parameter values as input, while the elicitation of values for preferential parameters should be considered in relation with the impacts of the parameters’ values on the results. However, methodologies proposed in the literature do not consider preference elicitation and robustness analysis within an integrated approach.

The purpose of this paper is to show how these two phases of a decision aiding process can be

viewed within a single integrated approach. Within such an approach, the preference elicitation process is a sequence of questions and answers. The DMs are provided with the solution(s) derived by the model using their answers to the preceding questions. Hence, they can react interactively and control their preference information with regard to the conclusions that are robust. We will restrict the analysis to the ELECTRE TRI method (see Mousseau et al., 1999, 2000b; Roy and Bouyssou, 1993). However, we deem that the basic idea underlying our work can be applied to a large class of methods.

The paper is organised as follows. The next section provides a brief reminder on the well-known ELECTRE TRI method. Section 3 presents two approaches developed independently: the first one consists of inferring values for the ELECTRE TRI method’s parameters from assignment examples (see Mousseau and Slowinski, 1998; Mousseau et al., 2000a), whereas the second one consists of computing robust assignments (i.e., the best and the worst assignment for each alternative) consistent with the constraints on parameters (see Dias and Clímaco, 1999, 2000). Both approaches avoid asking for precise values for the parameters. Rather, they proceed to solve the problem in a way that requires much less cognitive effort from the DMs. Section 4, after introducing the concept of constructive learning procedures, shows how these two approaches can be combined into an integrated methodology to determine robust assignments interactively, within an aggregation/disaggregation approach. The computational aspects are presented in Section 5 and an illustrative example is provided in Section 6. Section 7 draws some conclusions and issues for further research.

2. Brief presentation of the ELECTRE TRI method

This section gives a very brief overview of the ELECTRE TRI method and defines some notations that will be used along the paper. For more details, see Mousseau et al. (1999, 2000b) and Roy and Bouyssou (1993).

ELECTRE TRI is a multiple criteria sorting method, i.e., a method that assigns alternatives to pre-defined ordered categories delimited by reference alternatives called profiles. The assignment of an alternative a results from the comparison of a with the profiles defining the limits of the categories. Let $F = \{1, 2, \dots, n\}$ denote the set of the indices of the criteria g_1, g_2, \dots, g_n and let $B = \{1, 2, \dots, p\}$ denote the set of indices of the profiles b_1, b_2, \dots, b_p defining $p + 1$ categories, b_h being the upper limit of category C_h and the lower limit of category C_{h+1} , $h = 1, 2, \dots, p$.

ELECTRE TRI builds a fuzzy outranking relation S whose meaning is “at least as good as”. Preferences restricted to the significance axis of each criterion are defined through pseudo-criteria (see Roy and Vincke, 1984, for details on this double-threshold preference representation). The indifference and preference thresholds, $q_j(b_h)$ and $p_j(b_h)$, respectively, constitute the intra-criterion preferential information. Two types of inter-criteria preference parameters intervene in the construction of S :

- the set of weight-importance coefficients $w = (w_1, w_2, \dots, w_n)$ is used to aggregate the single-criterion concordance indices $c_j(a, b_h)$ ($j = 1, \dots, n$), which indicate how much each criterion agrees with the assertion aSb_h , into a global concordance index $c(a, b_h)$, which quantifies the relative importance of the coalitions of criteria being in favour of the assertion aSb_h (similarly, $c(b_h, a)$ is computed concerning the assertion b_hSa);
- the set of veto thresholds $(v_1(b_h), v_2(b_h), \dots, v_n(b_h))$ is used to compute discordance indices; $v_j(b_h)$, represents the smallest difference of performances in criterion g_j incompatible with the assertion aSb_h (and b_hSa).

As the assignment of alternatives to categories does not result directly from the relation S , an exploitation phase is necessary; it requires the relation S to be “defuzzified”: the assertion aSb_h is considered to be valid if the credibility index of the fuzzy outranking relation is greater than a “cutting level” λ (with $\lambda \in [0.5, 1]$).

Two assignment procedures (optimistic and pessimistic) are available to analyse the way alternative a compares to the profiles, in order to

determine its category. The result of these two assignment procedures differs when the alternative a is incomparable with at least one profile b_h .

3. Preference elicitation and robustness analysis for ELECTRE TRI

3.1. Main features of the inferring procedure

Mousseau and Slowinski (1998) proposed an inference procedure using the paradigm of aggregation/disaggregation, outlined in Fig. 1. Its aim is to find an ELECTRE TRI model as compatible as possible with the assignment examples given by the DMs. The assignment examples concern a subset $A^* \subset A$ of alternatives that the DMs can easily assign to a category (or a range of consecutive categories), taking into account their evaluation on all criteria. The compatibility between the ELECTRE TRI model and the assignment examples is understood as an ability of the ELECTRE TRI method using this model to reassign the alternatives of A^* in the same way as the DM did.

In order to minimise the differences between the assignments made by ELECTRE TRI and the assignments made by the DM, an optimisation procedure is used. The DMs can tune up the model in the course of an interactive procedure. They may either revise the assignment examples (i.e., remove and/or add some alternatives from/to A^* , change the assignment of some alternatives of A^*) or define constraints for some model parameters (i.e., ordinal information on the importance of criteria, incomplete definition of some profiles defining the limits between categories, ...) basing on their own intuition.

When the model is not perfectly compatible with the assignment examples, the procedure can detect all “hard cases”, i.e., the alternatives for which the assignment computed by the model strongly differs from the DMs’ assignment. The DMs could then be asked to reconsider their judgement.

The interaction procedure presented in Fig. 1 (grey items refer to the phases in which the intervention of DMs is required) stops when the DM is satisfied with the values proposed for the param-

eters. These values should be compatible with a set of assignment examples and possibly with additional constraints on the parameters value. The interaction provides the DMs a so-called “constructive learning” context in which they can improve their understanding on how the assignment model is affected by the values of the parameters and find a set of parameters values that are consistent with the assignment examples.

This methodology can be quite helpful when eliciting preferences in practical decision situations. However, Mousseau and Slowinski (1998)

do not provide much support to the DMs in the management of the interaction. There is a lack of control of the interaction process which is due to the absence of features that would ensure a certain form of “cognitive convergence” of the procedure.

3.2. A procedure to obtain robust conclusions

We consider in this section a situation where several combinations of values for the parameters are acceptable. Such a context is usually design-

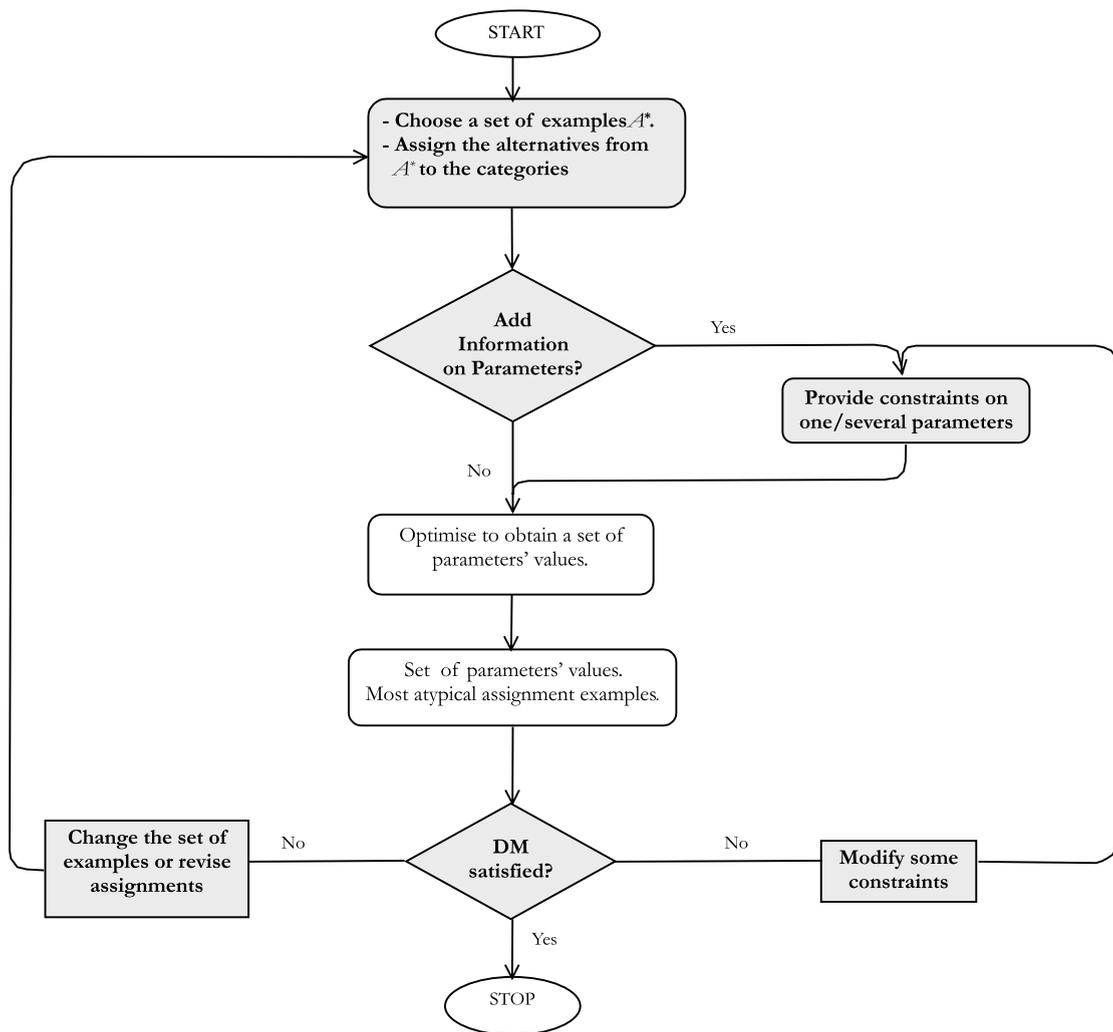


Fig. 1. General scheme of the inference procedure.

nated as a situation of “imprecise information” (see Athanassopoulos and Podinovski, 1997), “incomplete information” (see Weber, 1987), “partial information” (see Hazen, 1986) or “poor information” (see Bana e Costa and Vincke, 1995). We will use the expression “imprecise information”, meaning that it does not impose a precise combination of values for the parameters. The set of possible combinations is expressed as constraints, rather than as a discrete set of values. The constraints may be explicitly provided by the DMs or inferred from holistic comparisons (as in Mousseau, 1993, 1995). Although constraints are not always easy to provide, requiring precise values for the parameters is obviously more demanding.

Let T represent the set of all acceptable combinations of parameter values. One can determine the range of any credibility index subject to T (for details, see Dias and Clímaco, 1999), which may be used to determine the best and worst categories to which an alternative may be assigned, subject to T

(for details, see Dias and Clímaco, 2000). This approach determines the best and the worst categories, $B(a_i, T)$ and $W(a_i, T)$, respectively, to which alternative a_i can be assigned by ELECTRE TRI subject to the constraints defining T . If $B(a_i, T)$ coincides with $W(a_i, T)$, then the method is able to assign a_i to a single category, despite the imprecision regarding the input.

Three types of information are considered in Dias and Clímaco’s approach:

1. Robust conclusions as regards T : stating that a_i belongs to a category no worse than $W(a_i, T)$ is a robust conclusion, and so is stating that a_i belongs to a category no better than $B(a_i, T)$.
2. Identification of the alternatives that are more affected by the imprecision of data. Indeed, it may be interesting to know that some alternatives have a wide range of categories to which they may be assigned, in contrast with some other alternatives that are precisely assigned to a single category.

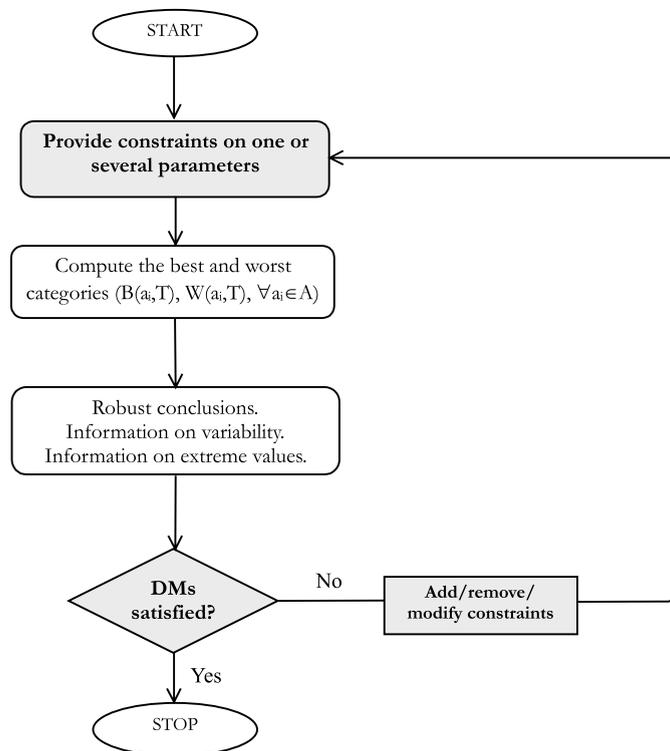


Fig. 2. General scheme of the robustness analysis procedure.

3. “Extreme” combinations of parameter values corresponding to best-case and worst-case assignments.

We may consider the use of this type of approach to be an interactive learning process, where the results of the analysis may stimulate the DMs to discuss and revise their inputs, as outlined in Fig. 2. In general, DMs could begin with little information (starting with a set T not too constrained) and then progressively enrich that information (reducing T) as they form their judgements. This type of approach then tries to identify conclusions that can be accepted as valid, despite the lack of precision present in the information they provide.

Dias and Clímaco (1999, 2000) addressed a general case that involves solving many non-linear maximisation/minimisation problems, which become linear programs when veto thresholds are fixed or constrained to intervals. They did not address explicitly the problem of obtaining those constraints from the DMs. Therefore, the integrated approach presented in the next section can be seen as a complement to the robustness analysis approach, to the same extent that it may be seen as a complement to the inference approach in Section 3.1.

4. A new constructive learning procedure

4.1. The constructive learning process

The process that leads to the definition of a multiple criteria sorting model can be analysed according to different perspectives. The role of assignment examples, constraints on preference parameters and the way the model can be validated are very different when they are conceived according to a descriptive or constructive process.

Defining a sorting model can be viewed in a *descriptive learning* perspective, i.e., as a process in which the model reproduces a class of input–output behaviour accounting for a set of learning examples. Machine learning (Michalski, 1983; Quilan, 1986) typically enters this perspective in which the definition of the model should optimise

the explanation of the examples. Hence, the validity of the obtained model is grounded on the input data and its ability to reproduce similar classifications. The role of the DMs in the definition of the model is reduced to providing learning examples. Different models that equally explain the data are considered equivalent.

On the other hand, a *constructive learning* perspective regards the definition of the sorting model as a result between the interaction of the assignment examples and the DMs’ points of view. The process should result in a model that both explains the examples and accounts for the DMs’ perception of the problem. During the construction process, the DMs might modify some assignment examples while increasing their understanding of the problem. The very nature of this constructive learning process is to provide the DMs with an interactive context in which they may check the impact of assignment examples on the model parameters, in order to find a model that fits both a set of learning examples and their perception of the sorting problem.

The combined approach proposed in this paper falls into the context of constructive learning, as the integration of the DMs in the process of defining the sorting model is considered a crucial point.

4.2. A combined approach

We can combine the two approaches presented in Sections 3.1 and 3.2 to take advantage of both of them. Although the methodologies proposed by Mousseau and Slowinski (1998) and Dias and Clímaco (1999, 2000) apply when all preferential parameters are considered, we restrict the presentation of our ideas to the case where only the weights $w = (w_1, w_2, \dots, w_n)$ and cutting level λ are variables and when no discordance occurs (the veto thresholds do not play any role). Hence, the optimisation problems to be solved are linear programs, which are much easier to solve than the nonlinear problems that usually arise when all the parameters are being inferred at the same time. However we do not believe that a procedure that infers all the pa-

parameters simultaneously is the best way of eliciting preferences: such procedure gives a poor control of the preference elicitation during the interaction with the DM. We rather recommend to elicit interactively subsets of preferential parameters, the other parameters being fixed.

4.2.1. Input information required from the DMs

The procedure we propose does not require the DMs to provide precise values for the preferential parameters. At iteration k , the input may consist of:

1. Explicit preferential information i.e., linear constraints on the parameter values, which are always written in the same manner. Specifically take the form

$$\sum_{j=1}^n w_j \cdot c_j(a, b) \leq \sum_{j=1}^n w_j \cdot c_j(b, a),$$

where a and b correspond to any real or fictitious alternative. These constraints usually correspond to pair-wise comparisons of coalitions of criteria. These constraints define a domain T_k^w .

2. Assignment examples concerning a subset of alternatives $A^* \subset A$ that the DMs are able to assign holistically to a category (or a range of categories), defining a domain T_k^a .

Both types of input information take the form of constraints on the values of preferential parameters, since assignment examples are expressed in the model through a set of linear constraints (using the ELECTRE TRI pessimistic assignment rule). Let $T_k = T_k^w \cap T_k^a$ denote the set (a polyhedron) of all acceptable combinations $t = (w_1, \dots, w_n, \lambda)$ corresponding to the input information provided by the DMs at the iteration k (either through assignment examples or constraints on the parameter values). At each iteration the input information may be inconsistent or not, i.e., the set T_k may be empty or not.

4.2.2. Output information provided to DMs

The information provided to the DMs at iteration k is different depending on the consistency of the input information (details concerning

the computation of these results are given in Section 5):

(a) The input information is considered as inconsistent if T_k is empty, i.e., there is no combination of parameter values that fully restores the assignment examples and simultaneously conforms to the additional constraints. In this case we can provide the DMs with:

1. A combination of parameter values that least violates the constraints that define T_k . It can occur that multiple optimal solutions exist. However, the choice of one among these optima is not a crucial issue in our methodology. All of these optima restore the decision maker's examples (when the information is consistent); moreover the DM can revise his/her judgments after each iteration. Hence, he/she can select an appropriate set of parameter values, during the iteration process, by revising the constraints and as a result of a learning process.
2. The constraints defining T_k^a that are the most difficult to satisfy, which yield the alternatives from A^* that are the most difficult to assign.
3. The constraints defining T_k^w that are the most difficult to satisfy.
4. Proposals of modification of T_k so that this polyhedron does not become empty.

(b) The input information is consistent, i.e., T_k is not empty. In this case we can provide the DMs with:

1. A combination of parameter values $t_k^* \in T_k$ that best match the provided information.
2. The worst category $W(a_i, T_k)$ and the best category $B(a_i, T_k)$ to which each alternative $a_i \in A$ can be assigned to for any combination of parameter values in T_k ;

(Note: Dias et al. (2000) show that when an alternative a_i is such that $c_j(a_i, b_h) = c_j(a_i, b_{h+1}) \forall j$ (b_h and b_{h+1} defining a category C_h in the range $[W(a_i, T_k), B(a_i, T_k)]$), there exists no $t \in T_k$ such that $a_i \rightarrow_t C_h$ (where the symbol \rightarrow denotes the assignment operation)).

3. The category in $[W(a_i, T_k), B(a_i, T_k)]$ corresponding to t_k^* for each alternative $a_i \in A$.
4. "Extreme" combinations $t_k^B(a_i)$ and $t_k^W(a_i)$ leading to the best and worst case assignment, respectively, for each alternative $a_i \in A$.

5. For each category C_h to which any alternative $a_i \in A$ can be assigned, a “typical” combination of parameter values leading to this assignment.

4.2.3. Interaction process

The proposed procedure is designed to be used interactively, i.e., the output at a given

iteration k is used to guide the DMs’ discussion and revision of the input for the following iteration (T_{k+1}). Fig. 3 presents the general scheme of the procedure. The nature of the interaction depends on the consistency of the input information provided at the current iteration.

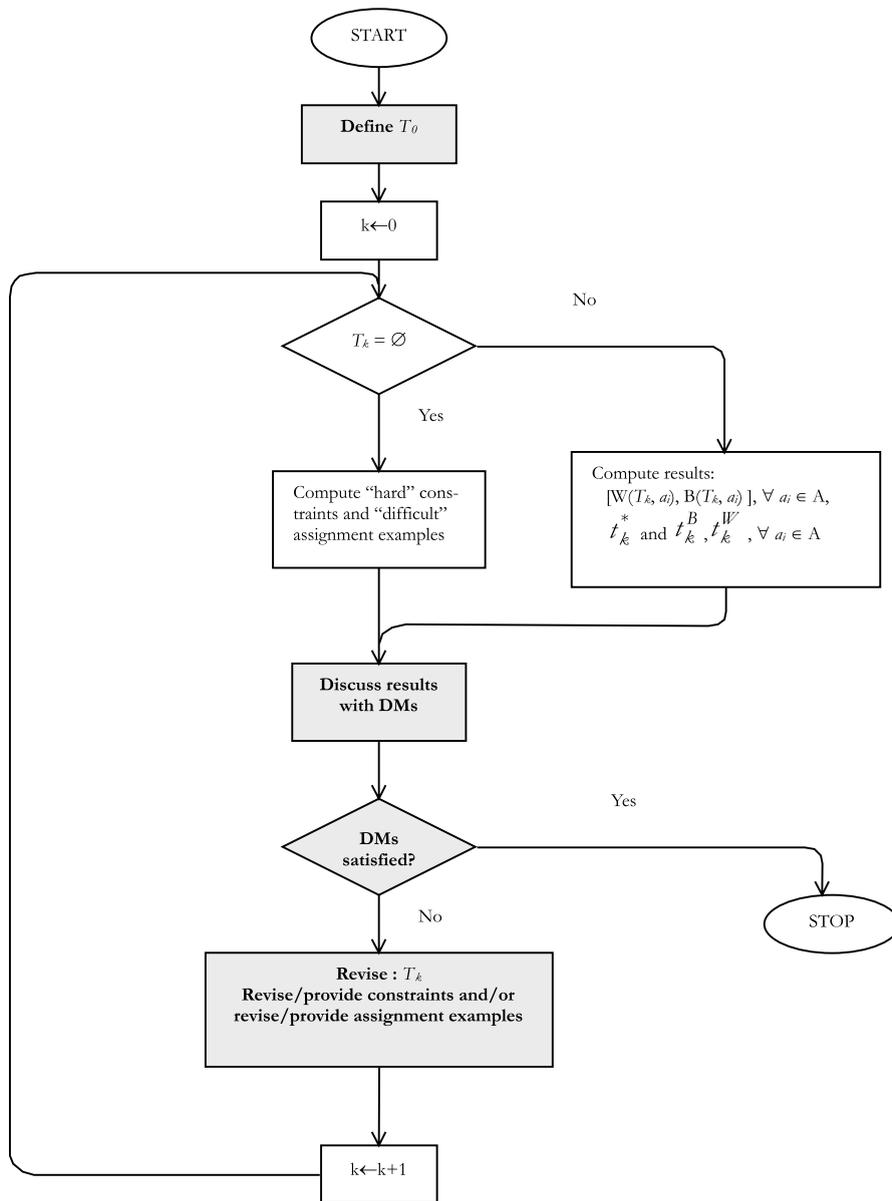


Fig. 3. The proposed procedure.

(a) If $T_k = \emptyset$, the interaction aims at eliminating the inconsistency. The DMs should react by a modification or a deletion of a constraint. Several elements can support them in such revision of their input information:

- the list of assignment examples that are the most difficult to restore in the ELECTRE TRI model are the ones that should be considered first for an eventual modification or deletion;
- similarly, the constraints defining T_k^w that are the most difficult to respect should be considered first;
- the deletion of specific subsets of constraints can lead to a new polyhedron T_{k+1} which is not empty. Several such sets can be computed and presented to the DMs for them to choose.

(b) If $T_k \neq \emptyset$, the interaction aims at reducing the set of possible combinations of parameter values. Hence, the DMs should react in order to modify a constraint and/or add a new one. Several elements can support the DMs in such revisions of their input information:

- the combinations of parameters values t_k^* , $t_k^B(a_i)$ and $t_k^W(a_i)$ may suggest additional constraints for the DMs define T_{k+1} ; for instance, if t_k^* is such that $\lambda = 0.5$, the DMs might react by stating that $\lambda \geq 0.7$;
- the ranges of $[W(a_i, T_k), B(a_i, T_k)]$ for all $a_i \in A$ provide the DMs with useful information for selecting an alternative to be added to A^* ; alternatives with the widest ranges are the most affected by the imprecision; hence, giving additional information about their assignment should be highly informative; we conjecture that on average there ought to be less iterations to reach acceptably precise assignments when DMs choose alternatives with larger variability as assignment examples.

This interactive procedure is designed in such a way that the DMs can start the first iteration with very little information. Each iteration will provide opportunity to add, delete or modify a specific supplementary constraint. Adding only a single piece of information at each iteration facilitates the control of the information supplied by the DMs.

As the DMs add constraints during the successive iterations, the domain T_k becomes more

constrained and the range of categories $[W(a_i, T_k), B(a_i, T_k)]$ (for each $a_i \in A$) is reduced or at least remains invariant (unless the domain T_k becomes empty, meaning that the new constraints contradict some previous ones and that DMs should revise T_k). The interactive procedure stops when DMs are satisfied and the set T_k as well as the assignment of alternatives matches their view of the decision problem. The final results are:

- a set of constraints and assignment examples defining a set T_k of acceptable combination of parameter values;
- an inferred combination of parameter values t_k^* defining a model in a precise manner;
- a precise assignment or range of assignments for each alternative in A that is robust with respect to T_k .

However, we want to emphasise the fact that this interactive process does not aim only at providing the above-mentioned results. During the interaction, the DMs will increase the insight on their view of the problem and will possibly modify their opinions. This interaction will provide them with an opportunity to learn about their preferences and increase their understanding of the decision problem.

5. Computational aspects

At each iteration k , results are computed according to T_k in order to provide the DMs results on the basis of which they can refine the input information T_k and define T_{k+1} . The computations are different depending on the consistency of the input information T_k .

5.1. Computations when T_k is inconsistent

5.1.1. Parameter values that least violates the constraints that define T_k

In order to determine the combination of values for w and λ that least violate the constraints defining T_k , we solve the following linear program (LP1):

LP1 Max α

s.t.

$$\alpha \leq x_i \quad \forall i \text{ such that } a_i \in A^*, \quad (1)$$

$$\alpha \leq y_i \quad \forall i \text{ such that } a_i \in A^*, \quad (2)$$

$$\alpha \leq s_q \quad \forall q = 1, \dots, n_c, \quad (3)$$

$$\alpha \leq s_\lambda^{\min}, \quad (4)$$

$$\alpha \leq s_\lambda^{\max}, \quad (5)$$

$$\sum_{j=1}^n w_j c_j(a_i, b_{l_i-1}) - x_i = \lambda$$

$$\forall i \text{ such that } a_i \in A^*, \quad (6)$$

$$\sum_{j=1}^n w_j c_j(a_i, b_{u_i}) + y_i = \lambda + \epsilon$$

$$\forall i \text{ such that } a_i \in A^*, \quad (7)$$

$$\sum_{j=1}^n w_j c_j(a', a'')$$

$$- \sum_{j=1}^n w_j c_j(a'', a') - z_q = 0$$

$$\forall q = 1, \dots, n_c$$

$$\forall a', a'' \text{ real or fictitious alternatives}, \quad (8)$$

$$\lambda - \lambda^{\min} - s_\lambda^{\min} = 0, \quad (9)$$

$$\lambda - \lambda^{\max} + s_\lambda^{\max} = 0, \quad (10)$$

$$\epsilon' \leq w_j \leq 0.5 - \epsilon', \quad j \in F,$$

$$\sum_{j=1}^n w_j = 1, \quad (11)$$

$$0.5 \leq \lambda \leq 1, \quad (12)$$

$$x_i, y_i, s_q, s_\lambda^{\min}, s_\lambda^{\max}, \alpha \text{ free}, \quad (13)$$

where

- l_i (u_i , respectively) is the index of the lowest category (highest category, respectively) to which the alternatives $a_i \in A^*$ can be assigned, according to the opinion of the DMs ($l_i = u_i$, means that the DMs are able to assign the alternatives a_i to a single category).
- λ^{\min} (λ^{\max} , respectively) is the lower bound (upper bound, respectively) for the cutting level λ . This information stated by the DMs is such that $0.5 \leq \lambda^{\min} \leq \lambda^{\max} \leq 1$.
- T_k^a is the set of combinations of weights that

conform to the constraints implied by the assignment examples at iteration k . In the mathematical program below, T_k^a corresponds to constraints (6) and (7).

- T_k^w is the set of combinations of parameters values that conform to the additional information provided by the DMs at iteration k . T_k^w is defined by n_c constraints (8) of the form

$$\sum_{j=1}^n w_j c_j(a', a'') - \sum_{j=1}^n w_j c_j(a'', a') \geq z_q$$

$$\forall q = 1, \dots, n_c,$$

(where a' and a'' denotes any real or fictitious alternative) and constraints (9) and (10) defining bounds for λ .

- ϵ and ϵ' represents an arbitrary small positive value.
- x_i and y_i ($\forall a_i \in A^*$), s_q ($q = 1, \dots, n_c$), s_λ^{\min} and s_λ^{\max} , are the slack variables that account for the “degree to which” a constraint is fulfilled; if negative, the constraint is violated and T_k is empty.

5.1.2. Alternatives from A^* that are the most difficult to assign

In LP1, these alternatives correspond to the a_i such that $x_i = \alpha$ and/or $y_i = \alpha$ at the optimum.

5.1.3. Constraints defining T_k^w that are the most difficult to respect

In LP1, these constraints correspond to the ones for which $s_q = \alpha$ at the optimum.

5.1.4. Proposals of modification of T_k

Proposals of modification of T_k correspond to subsets of constraints that, when removed, lead to a consistent information (see for example Chinneck, 1996).

5.2. Computations when T_k is consistent

5.2.1. Parameter values $t_k^* \in T_k$ that best match the provided information

The set of parameter values for w and λ is contained in the optimal solution of LP1 (see Section 5.1.1).

Table 1
Assignment ranges after the first iteration

	C ₁	C ₂	C ₃	C ₄	C ₅		C ₁	C ₂	C ₃	C ₄	C ₅
a ₂₈				X		a ₂₇				X	
a ₁₄				X		a ₂₉				X	
a ₃₆				X		a ₃₀				X	
a ₃₁		X				a ₃₂					X
a ₂					X	a ₃₇				X	
a ₃			X			a ₂₄		X			
a ₄				X		a ₃₄			X		
a ₈				X		a ₃₈			X		
a ₉					X	a ₃₉			X		
a ₁₀				X		a ₃₅		X			
a ₁₁					X	a ₁				X	
a ₁₂				X		a ₅					X
a ₁₃				X		a ₆					X
a ₁₆				X		a ₇					X
a ₁₈				X		a ₁₅				X	
a ₁₉				X		a ₁₇				X	
a ₂₀				X		a ₂₃			X		
a ₂₁				X		a ₂₆			X		
a ₂₂				X		a ₃₃			X		
a ₂₅				X		a ₀					X

5.2.2. Best and worst categories to which each alternative $a_i \in A$ can be assigned considering T_k

The alternative a_i belongs to the highest category C_h such that $a_i S b_{h-1}$, i.e.,

$$a_i \rightarrow_t C_h \iff \sum_{j=1}^n w_j c_j(a_i, b_h) - \lambda < 0 \wedge \sum_{j=1}^n w_j c_j(a_i, b_{h-1}) - \lambda \geq 0.$$

To determine the best possible assignment $B(a_i, T_k)$ for the alternative a_i considering the domain T_k of possible values for w and λ , we perform successive tests by maximising the credibility ($h = 1, 2, \dots, p$):

$$\text{LP2} \quad \max \sum_{j=1}^n w_j c_j(a_i, b_h) - \lambda$$

s.t. $t = (w, \lambda) \in T_k.$

To determine the worst possible assignment $W(a_i, T_k)$ for the alternative a_i , we perform the tests by minimising the credibility ($h = 1, 2, \dots, p$):

$$\text{LP3} \quad \min \sum_{j=1}^n w_j c_j(a_i, b_h) - \lambda$$

s.t. $t = (w, \lambda) \in T_k.$

5.2.3. “Extreme” combinations $t_k^B(a_i)$ and $t_k^W(a_i)$ leading to the best and worst case assignment

$t_k^B(a_i)$ and $t_k^W(a_i)$ are obtained directly by solving LP2 and LP3.

Table 2
Assignment ranges after the second iteration

	C ₁	C ₂	C ₃	C ₄	C ₅		C ₁	C ₂	C ₃	C ₄	C ₅
a ₂			X		X	a ₁₈			X		
a ₃				X		a ₁₉			X		
a ₉			X			a ₂₀			X		
a ₁₀			X			a ₂₁			X		
a ₁₁				X		a ₂₂				X	
a ₃₂			X			a ₂₃				X	
a ₁₄		X	X			a ₂₅			X		
a ₃₄			X			a ₂₆			X		
a ₃₆		X				a ₂₇			X		
a ₃₈		X				a ₂₉				X	
a ₃₉		X				a ₃₀				X	
a ₃₅	X	X				a ₃₃			X		
a ₅				X		a ₃₇			X		
a ₆				X		a ₂₄		X	X		
a ₇					X	a ₃₁	X				
a ₄			X			a ₀					X
a ₈				X		a ₁				X	
a ₁₂			X			a ₁₅				X	
a ₁₃			X			a ₁₇				X	
a ₁₆			X			a ₂₈	X				

5.2.4. For each category C_h to which any alternative $a_i \in A$ can be assigned, a “typical” combination of values

This result is obtained by the resolution of LP1 (see Section 5.1.1) to which the constraints corresponding to the assignment of a_i to C_h are added.

5.2.5. For each category C_h to which $a_i \in A$ cannot be assigned, the constraints that make impossible for a_i to be assigned to C_h

In order to compute the constraints that forbid the assignment of an alternative $a_i \in A$ to a category C_h , we may proceed as follows:

1. Add the assignment example $a_i \rightarrow C_h$, to T_k . The information becomes inconsistent.
2. Compute proposals of modification of T_k that make the information consistent, while containing the constraints corresponding to the assignment example $a_i \rightarrow C_h$.

6. An illustrative example

In order to illustrate the proposed methodology, let us consider the data from a real world application in the banking sector (see Dimitras et al., 1995). We consider a model that aims at assigning 40 alternatives to five categories on the basis of their evaluations on seven criteria. The data are also presented in Dias et al. (2000).

Using this data we simulate a posteriori the modelling process that could have taken place using our methodology. We consider a hypothetical preference elicitation process, supposing that the DMs were able to elicit directly the limits of categories b_h and the thresholds $q_j(b_h)$ and $p_j(b_h)$, but had difficulties with assigning values for the importance coefficients w_j and the cutting level λ .

Let us suppose that the only initial information that the DMs are able to express is that the most important criterion is g_2 (i.e., $w_2 \geq w_j, j \neq 2$), and

Table 3
Assignment ranges after the third iteration

	C ₁	C ₂	C ₃	C ₄	C ₅		C ₁	C ₂	C ₃	C ₄	C ₅
a ₃₁			X			a ₂₄			X		
a ₁₀					X	a ₀					X
a ₁₂					X	a ₁					X
a ₁₃				X		a ₂					X
a ₁₄				X		a ₃					X
a ₁₅					X	a ₄					X
a ₁₆					X	a ₅					X
a ₁₇					X	a ₆					X
a ₁₈				X		a ₇					X
a ₁₉				X		a ₈					X
a ₂₀				X		a ₉					X
a ₂₁				X		a ₁₁					X
a ₂₂					X	a ₂₃				X	
a ₂₅				X		a ₂₆				X	
a ₂₇				X		a ₃₃				X	
a ₂₉					X	a ₃₄				X	
a ₃₀					X	a ₃₈				X	
a ₃₂					X	a ₃₉				X	
a ₃₆				X		a ₃₅			X		
a ₃₇						a ₂₈	X				

no assignment example is provided by DMs. Suppose also that $\lambda \in [0.5, 0.99]$. This information is obviously consistent and t_0^* is such that $w_2 = 0.49$, $w_j = 0.085 \forall j \neq 2$, and $\lambda = 0.5$. Ranges $[W(a_i, T_0), B(a_i, T_0)] \forall a_i \in A$, are presented in Table 1 in which the grey cells correspond to possible assignments and the marked cells correspond to the assignments made by ELECTRE TRI pessimistic rule using t_0^* . Alternatives are sorted according to their assignment range.

Analysing these first results, the DMs are surprised with the wide range (C₁ to C₅, except C₂ as $c_j(a_{28}, b_1) = c_j(a_{28}, b_2) \forall j \in F$, see Dias et al., 2000) of possible assignments for a_{28} . Considering evaluations of a_{28} on the criteria and their expertise of the problem, the DMs state that a_{28} should be assigned to category C₁. Considering T_1 (the information is still consistent), computations show that t_1^* is such that $w_2 = 0.49$, $w_5 = 0.25$,

$w_7 = 0.22$, $w_j = 0.01$ for all other criteria and $\lambda = 0.99$. Ranges $[W(a_i, T_1), B(a_i, T_1)] \forall a_i \in A$, are presented in Table 2.

Considering the result presented in Table 2, the DMs are surprised that the alternative a_1 is assigned to category C₄. They consider that this alternative should be assigned to category C₅. However, it is clear that adding the assignment example “ $a_1 \rightarrow C_5$ ” would make T_2 inconsistent (see Table 2). Though, we determine which constraints contradict such assertion, i.e., the subsets of constraints with minimal cardinality whose deletion from T_2 makes the information consistent. In our case, the deletion of one constraint can make T_2 consistent. This constraint is one among the following:

- $a_1 \rightarrow C_5$;
- $a_{28} \rightarrow C_1$;
- $w_2 \geq w_5$.

Table 4
Assignment ranges after the fourth iteration

	C ₁	C ₂	C ₃	C ₄	C ₅		C ₁	C ₂	C ₃	C ₄	C ₅
a ₁₀				X	X	a ₁					X
a ₁₂				X	X	a ₂					X
a ₁₃				X	X	a ₃					X
a ₁₄				X	X	a ₄					X
a ₁₅				X	X	a ₅					X
a ₁₆				X	X	a ₆					X
a ₁₇				X	X	a ₇					X
a ₁₉				X	X	a ₈					X
a ₂₀				X	X	a ₉					X
a ₂₁				X	X	a ₁₁					X
a ₂₂				X	X	a ₁₈				X	
a ₂₅				X	X	a ₂₃				X	
a ₂₇				X	X	a ₂₆				X	
a ₂₉				X	X	a ₃₃				X	
a ₃₀				X	X	a ₃₄				X	
a ₃₂				X	X	a ₃₈				X	
a ₃₆				X	X	a ₃₉				X	
a ₃₇				X	X	a ₃₅			X		
a ₂₄			X			a ₃₁		X			
a ₀				X		a ₂₈	X				

Let us suppose that DMs are rather confident in the two assignment examples but feel less confident about the constraint concerning w_2 and w_5 , which they accept to remove. Considering the updated information (T_3 is not empty), t_3^* is such that $w_5 = 0.49$, $w_2 = 0.235$, $w_4 = w_6 = 0.1225$, $w_j = 0.01$, $j = 1, 3, 7$, and $\lambda = 0.6225$. Ranges $[W(a_i, T_3), B(a_i, T_3)] \forall a_i \in A$, are presented in Table 3.

At this stage, the DMs would like to add a new assignment example by stating that the alternative a_{31} is assigned to category C_2 . The new information is consistent and t_4^* is such that $w_5 = 0.48$, $w_2 = 0.282$, $w_4 = 0.198$, $w_j = 0.01$, $j = 1, 3, 6, 7$, and $\lambda = 0.604$. Ranges $[W(a_i, T_4), B(a_i, T_4)] \forall a_i \in A$, are presented in Table 4.

The DMs then add a supplementary constraint concerning the weights $w_3 \geq w_4$. Adding this constraint makes the information inconsistent (T_5 is

empty). It is however possible to determine a set of parameters that least violates the information contained in T_5 : $w_5 = 0.333$, $w_2 = w_3 = 0.160$, $w_4 = w_6 = 0.166$, $w_1 = w_7 = 0.01$ and $\lambda = 0.667$. In this case, we determine which constraints contradict such assertion, i.e., the subsets of constraints with minimal cardinality whose deletion from T_5 makes the information consistent. In our case, the deletion of one constraint can make T_5 consistent. This constraint is one among the following:

- $a_1 \rightarrow C_5$;
- $a_{31} \rightarrow C_2$;
- $w_3 \geq w_4$.

Let us suppose that the DMs decide to revise their views concerning alternative a_1 in such a way that $a_1 \rightarrow (C_4 \text{ or } C_5)$. Results are computed considering this updated information. T_6 is consistent and t_6^* is such that $w_5 = 0.406$, $w_2 = 0.396$,

Table 5
Assignment ranges after the sixth iteration

	C ₁	C ₂	C ₃	C ₄	C ₅		C ₁	C ₂	C ₃	C ₄	C ₅
a ₃			X		X	a ₁₃			X	X	
a ₈				X		a ₁₈				X	
a ₉					X	a ₁₉				X	
a ₁₀				X		a ₂₀				X	
a ₁₁					X	a ₂₁				X	
a ₁₂				X		a ₂₂				X	
a ₁₆				X		a ₂₃				X	
a ₂₉				X		a ₂₅				X	
a ₃₂					X	a ₂₆				X	
a ₁₄		X		X		a ₂₇				X	
a ₃₄		X		X		a ₃₀				X	
a ₃₆		X		X		a ₃₃				X	
a ₃₈		X		X		a ₃₇				X	
a ₃₉		X		X		a ₂₄		X			
a ₃₅	X		X			a ₀					X
a ₅				X		a ₁				X	
a ₆				X		a ₁₅				X	
a ₇				X		a ₁₇				X	
a ₂			X	X		a ₃₁		X			
a ₄			X	X		a ₂₈	X				

$w_3 = 0.158$, $w_j = 0.01$, $j = 1, 4, 6, 7$ and $\lambda = 0.743$. Ranges $[W(a_i, T_6), B(a_i, T_6)] \forall a_i \in A$, are presented in Table 5.

Afterwards, the DMs add a new assignment example $a_{16} \rightarrow C_3$, yielding T_7 (consistent) and later they add the assignment example $a_{14} \rightarrow C_2$. The new information is still consistent (T_8 is not empty) and t_8^* is such that $w_5 = 0.225$, $w_2 = 0.295$, $w_3 = 0.225$, $w_7 = 0.225$, $w_j = 0.01$, $j = 1, 4, 6$ and $\lambda = 0.846$. Ranges $[W(a_i, T_8), B(a_i, T_8)] \forall a_i \in A$, are presented in Table 6.

The interaction process stops here, as the DMs are satisfied with this assignment table and the corresponding values of preferential parameters.

At the end of this interactive elicitation process, the DMs have a set of weights and a value for the cutting level defining an assignment model.

Moreover, these values for w and λ are justified by the information they have provided in the process, i.e., the assignment examples and constraints on the parameters.

This example of a fictitious elicitation process shows how the interaction is made possible through the proposed procedure. We would like to emphasise that the procedure is designed in such a way that DMs can learn about their preferences during the process. For instance, their opinion concerning the assignment of alternative a_1 changed during the process.

Another important issue illustrated through this example deals with the “amount of information” provided at each iteration. In our example, only a single piece of information is added at each iteration. In this way, the DMs keep a good control of the interaction. This feature is

Table 6
Assignment ranges after the eighth iteration

	C ₁	C ₂	C ₃	C ₄	C ₅		C ₁	C ₂	C ₃	C ₄	C ₅
a ₂₂				X		a ₁₃			X		
a ₂₉				X		a ₁₆			X		
a ₃₀				X		a ₁₈			X		
a ₂₄		X				a ₁₉			X		
a ₃₅		X				a ₂₀			X		
a ₀					X	a ₂₁			X		
a ₁				X		a ₂₃			X		
a ₅				X		a ₂₅			X		
a ₆				X		a ₂₆			X		
a ₇				X		a ₂₇			X		
a ₁₅				X		a ₃₂			X		
a ₁₇				X		a ₃₃			X		
a ₂			X			a ₃₇			X		
a ₃			X			a ₁₄		X			
a ₄			X			a ₃₁		X			
a ₈			X			a ₃₄		X			
a ₉			X			a ₃₆		X			
a ₁₀			X			a ₃₈		X			
a ₁₁			X			a ₃₉		X			
a ₁₂			X			a ₂₈	X				

particularly important for understanding the inconsistencies.

7. Conclusion

A classical view of a multiple criteria decision aiding process is frequently the following: (1) definition of the criteria and alternatives, (2) choice of an aggregation model and preference elicitation, (3) use of the aggregation model, (4) robustness analysis and recommendations to the DMs.

Preference elicitation and robustness analysis are usually considered separately. In this paper, we show how these two phases of a decision aiding process can be viewed within a single integrated approach. We present a new approach to elicit an ELECTRE TRI model in a way that integrates the

preference elicitation phase and the construction of robust conclusions: in our interactive procedure, the information provided by the DMs for preference elicitation purposes is considered as compared to its implications on the robust conclusions that can be derived.

Although this work is dealing with the ELECTRE TRI method only, we claim that the basic ideas underlying our work can be applied to a large class of sorting methods such as additive utility sorting methods (UTADIS), rough sets based classifiers, etc.

Further work should be pursued in order to support the DM in the resolution of inconsistencies appearing during the elicitation process. Moreover, a software implementation is being developed in order to facilitate the use of the methodology in real world case studies.

Acknowledgements

This work was supported by the Luso-French grant no. 328J4 (ICCTI/Ambassade de France au Portugal). João Clímaco and Luís Dias acknowledge the support of PRAXIS grant PSCH/C/CEG/28/96.

References

- Athanassopoulos, A.D., Podinovski, V.V., 1997. Dominance and potential optimality in MCDA with imprecise information. *Journal of the Operational Research Society* 48, 142–150.
- Bana e Costa, C., Vansnick, J.-Cl., 1994. MACBETH – an interactive path toward the construction of cardinal value function. *International Transactions in Operational Research* 1 (4), 489–500.
- Bana e Costa, C.A., Vincke, Ph., 1995. Measuring credibility of compensatory preference statements when trade-offs are interval determined. *Theory and Decision* 39, 127–155.
- Belton, V., Vickers, S., 1990. The use of simple multi-attribute value function incorporating visual interactive sensitivity analysis for multiple criteria decision aid. In: Bana e Costa, C. (Ed.), *Readings in Multiple Criteria Decision Aid*. Springer, Berlin, pp. 319–334.
- Chinneck, J.W., 1996. An effective polynomial time heuristic for the minimum cardinality IIS set-covering problem. *Annals of Mathematics and Artificial Intelligence* 17, 127–144.
- Dias, L.C., Clímaco, J.N., 1999. On computing ELECTRE's credibility indices under partial information. *Journal of Multi-Criteria Decision Analysis* 8, 74–92.
- Dias, L.C., Clímaco, J.N., 2000. ELECTRE TRI for groups with imprecise information on parameter values. *Group Decision and Negotiation* 9 (5), 355–377.
- Dias, L.C., Mousseau, V., Figueira, J., Clímaco, J.N., 2000. An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI. *Cahier no. 174, LAMSADE, Université Paris-Dauphine*.
- Dimitras, A., Zopounidis, C., Hurson, C., 1995. Multicriteria decision aid method for the assessment of business failure risk. *Foundations of Computing and Decision Sciences* 20 (2), 99–112.
- Hazen, G.B., 1986. Partial information, dominance and potential optimality in multiattribute utility theory. *Operations Research* 34 (2), 297–310.
- Jacquet-Lagrèze, E., Siskos, J., 1982. Assessing a set of additive utility functions for multicriteria decision making, the UTA method. *European Journal of Operational Research* 10 (2), 151–164.
- Michalski, R.S., 1983. A theory and methodology of inductive learning. *Artificial Intelligence* 20, 11–116.
- Mousseau, V., 1993. Problèmes liés à l'évaluation de l'importance relative des critères en aide multicritère à la décision: Réflexions théoriques, expérimentation et implémentation informatique. Thèse de doctorat, Université Paris-Dauphine.
- Mousseau, V., 1995. Eliciting information concerning the relative importance on criteria. In: Pardalos, P., Siskos, Y., Zopounidis, C. (Eds.), *Advances in Multicriteria Decision Aid*. Kluwer Academic Publishers, Dordrecht, pp. 17–43.
- Mousseau, V., Slowinski, R., 1998. Inferring an ELECTRE TRI model from assignment examples. *Journal of Global Optimization* 12, 157–174.
- Mousseau, V., Slowinski, R., Zielniewicz, P., 1999. ELECTRE TRI 2.0a Methodological Guide and User's Manual. Document no. 111, LAMSADE, Université Paris-Dauphine.
- Mousseau, V., Figueira, J., Naux, 2000a. Using assignment examples to infer weights for the Electre Tri method: some experimental results. *European Journal of Operational Research* 130 (2), 263–275.
- Mousseau, V., Slowinski, R., Zielniewicz, P., 2000b. A user-oriented Implementation of the ELECTRE-TRI method integrating preference elicitation support. *Computers and Operations Research* 27, 757–777.
- Quilan, J.R., 1986. Induction of decision trees. *Machine Learning* 1, 81–106.
- Roy, B., 1998. A missing link in OR/DA: Robustness analysis. *Foundations of Computing and Decision Sciences* 23 (3), 141–160.
- Roy, B., Bouyssou, D., 1993. Aide multicritère à la décision: Méthodes et Cas. *Economica*.
- Roy, B., Vincke, Ph., 1984. Relational systems of preferences with one or more pseudo-criteria: Some new concepts and results. *Management Science* 30 (11), 1323–1334.
- Weber, M., 1987. Decision making with incomplete information. *European Journal of Operational Research* 28, 44–57.