4OR 4: 145–158 (2006)

DOI: 10.1007/s10288-005-0076-8

Dealing with inconsistent judgments in multiple criteria sorting models

Vincent Mousseau[1], **Luis C. Dias**[2], **and José Figueira**[2,3]

[1] LAMSADE, Université Paris-Dauphine, Place du Maréchal De Lattre de Tassigny, 75775 Paris, France (e-mail: mousseau@lamsade.dauphine.fr)
[2] INESC Coimbra / Faculty of Economics, University of Coimbra, Av. Dias da Silva 165, 3004-512 Coimbra, Portugal (e-mail: {lmcdias,figueira}@fe.uc.pt)
[3] DIMACS, Rutgers University, CoRE Building, 96 Frelinghuysen Road, Piscataway, NJ 08854-8018, USA

Received: September 2004 / Revised version: June 2005

Abstract. Sorting models consist in assigning alternatives evaluated on several criteria to ordered categories. To implement such models it is necessary to set the values of the preference parameters used in the model. Rather than fixing the values of these parameters directly, a usual approach is to infer these values from assignment examples provided by the decision maker (DM), i.e., alternatives for which (s)he specifies a required category. However, assignment examples provided by DMs can be inconsistent, i.e., may not match the sorting model. In such situations, it is necessary to support the DMs in the resolution of this inconsistency. In this paper, we extend algorithms from Mousseau et al. (2003) that calculate different ways to remove assignment examples so that the information can be represented in the sorting model. The extension concerns the possibility to relax (rather than to delete) assignment examples. These algorithms incorporate information about the confidence attached to each assignment example, hence providing inconsistency resolutions that the DMs are most likely to accept.

Key words: Multicriteria decision aiding, sorting problem, inconsistency analysis

MSC classification: 90B50, 91B08, 90C05

Correspondence to: Vincent Mousseau

4OR A Quarterly Journal of Operations Research

1 Introduction

Many real-world decision problems can be represented by a model stating explicitly the multiple points of view from which alternatives under consideration should be evaluated, through the definition of n_{crit} criterion functions $g_1, g_2, \ldots, g_j, \ldots, g_{n_{crit}}$. Given a set $A = \{a_1, a_2, \ldots, a_i, \ldots, a_{n_{alt}}\}$ of potential alternatives evaluated on the criteria, the analyst conducting the decision aiding study may formulate the problem in different terms. Roy (1996) distinguishes three problem statements (choosing, sorting, and ranking) that may guide the analyst in structuring the decision problem (see also Bana and Costa 1996).

In this paper, we are interested in sorting problems which consist in assigning each alternative to one of the pre-defined categories C_1, C_2, ..., C_k, ..., $C_{n_{cat}}$. The assignment of an alternative a_i results from its intrinsic evaluation on all criteria with respect to the norm defining the categories. Several methods have been proposed to handle multiple criteria sorting problems (MCSP), e.g., Trichotomic Segmentation (Moscarola and Roy 1977), N-TOMIC (Massaglia and Ostanello 1991), ORCLASS (Larichev and Moskovich 1994), ELECTRE TRI (Roy and Bouyssou 1993), PROAFTN (Belacel 2000), UTADIS (Zopounidis et al. 2001) and a general class of filtering methods (Perny 1998).

One of the major difficulties that an analyst must face when interacting with a decision maker (DM) in order to build a sorting model is the elicitation of various preference parameters used by the method. Even when these parameters can be interpreted, it is difficult to fix their values directly and to have a clear understanding of the implications of these values in terms of the output of the model. In order to avoid direct elicitation of the parameters, several authors have designed disaggregation procedures which make it possible to infer parameter's values from holistic judgments, an approach firstly introduced in the UTA method (Jacquet-Lagrèze and Siskos, 1982). Such procedures have been defined for MCSP (e.g., Zopounidis et al. 2001 for UTADIS and Mousseau and Slowinski 1998 for ELECTRE TRI).

The holistic judgments required to infer sorting models are called assignment examples and correspond to alternatives (real or fictitious) for which the DM can express a desired assignment, e.g., "a_i should be assigned to C_3" ($a_i \rightarrow C_3$), or "a_i should be assigned to C_1 or C_2" ($a_i \rightarrow [C_1, C_2]$, i.e., imprecise assignment examples can be considered). In some sorting methods (namely UTADIS and ELECTRE TRI when only the weights of criteria are inferred) such assignment examples define linear constraints on the model parameters.

When the assignment examples provided by the DM can be fully represented in the preference model, one may infer values for the models parameters that restore these examples. However, this is not always possible and the preference model cannot represent all examples simultaneously. Such a situation can be understood according to two perspectives: either the examples provided by the DM contradict each other, or the preference model is not flexible enough to account for the way the DM assigns alternatives holistically. In the first case, the DM would acknowledge a misjudgment and would agree to reconsider his/her examples; in the second case, the DM would not agree to change the examples and the preference model should be changed. In both cases, we refer to an inconsistency situation. In any case, the DM needs to know what causes inconsistency, i.e., which judgments should be changed if the aggregation model is to be kept. Very few multiple criteria methods incorporate such inconsistency analysis: MACBETH (Bana e Costa et al. 2005) provides such feature to build an additive model from qualitative judgments about difference of value.

Consider a problem in which a DM has specified assignment examples inducing linear inequalities on the preference parameters. This is namely the case with UTADIS (Zopounidis et al. 2001) and ELECTRE TRI (Dias et al. 2002) when only the weights of criteria are inferred. The assignment examples define a polyhedron of possible values for the parameters; when an inconsistent set of assignment examples is provided by the DM, this polyhedron is empty. There exist various ways by which the set of assignment examples can be modified so that the polyhedron becomes non-empty.

The problem is then to identify all the "minimal" subsets (in the sense of the inclusion) that resolve inconsistency, i.e., subsets among which the DM must choose in order to make his/her information consistent. In Mousseau et al. (2003), two algorithms are proposed to identify all the minimal subsets $S_q, q = 1, \dots, Q$ to be deleted (sorted by cardinality) that resolve inconsistency and whose cardinality is lower than (or equal to) `maxcount` (`maxcount` is an input to the algorithms that states the maximum number of solutions to be computed).

In this paper, we complement Mousseau et al. (2003) by proposing alternative ways to resolve inconsistencies stemming from a set of assignment examples. Namely, instead of deleting assignment examples, we consider relaxing them, i.e., enlarging the interval of the possible assignments for an alternative. Moreover, we consider that the DM may provide confidence levels associated with the assignment examples.

The paper is organized as follows. Section 1 defines inconsistency relaxation and shows that the algorithms proposed by Mousseau et al. (2003) still apply when considering constraints relaxation rather than constraints deletion. An illustrative example is introduced within the context of the ELECTRE TRI method. Section 2 considers the case in which the DM is able to provide confidence levels associated to the assignment examples, and suggests two ways to account for such information in order to rank the solutions according to the confidence levels provided by the DM. The example introduced in Sect. 1 is extended to this case. Finally, conclusions and suggestions for future research are provided.

2 Inconsistency resolution via constraints relaxation

Resolving the inconsistencies can be performed by deleting a subset of constraints. Let $x_1, x_2, \dots, x_j, \dots, x_n$ denote the n parameters of the considered sorting model. Let us denote $I = \{1, 2, ..., m\}$ the set of indices of the constraints and $T_\emptyset =$

$\{x \in \mathbb{R}^n : \sum_{j=1}^{n} \alpha_{ij} x_j \geq \beta_i, \forall i \in I\}$ the initial empty polyhedron, i.e., the polyhedron with all the initial constraints. Let $S \subseteq I$ denote a subset of indices of constraints. We will say that S resolves the inconsistency if and only if the polyhedron $T_S = \{\sum_{j=1}^{n} \alpha_{ij} x_j \geq \beta_i, \forall i \in I \setminus S\}$ is not empty.

In Mousseau et al. (2003) two algorithms are proposed to compute alternative ways to restore consistency by constraints deletion. We would like to consider here the case in which consistency can be solved by relaxing constraints rather than deleting them.

2.1 Defining constraints relaxations

Considering an infeasible system of linear inequalities (that can correspond to assignment examples), relaxing constraints rather than deleting them (in order to restore feasibility) has already been studied in the general case by (e.g., León and Liern 2001 and Roodman 1979). The relaxations considered by these authors are continuous and deal with the right-hand-side of the constraints only. In our case, we will define the relaxations differently:

- the relaxations will be performed by changing the technical coefficients α_{ij} of the constraints rather than the right-hand-side;
- a discrete set of relaxations will be considered which have a meaning with respect to the sorting model, namely increasing the interval of categories to which an alternative can be assigned.

Suppose the DM has specified a set of assignment examples, i.e., a subset of alternatives $A^* \subseteq A$ such that each $a_i \in A^*$ is associated with $max(a_i)$ ($min(a_i)$, respectively) the index of the maximum (minimum, respectively) category to which a_i should be assigned according to his/her holistic preferences ($a_i \rightarrow [C_{min(a_i)}, C_{max(a_i)}]$, $a_i \in A^*$). From the DM's perspective, $max(a_i)$ represents the statement "a_i *should be assigned at most to category* $C_{max(a_i)}$" (denoted $C(a_i) \leq max(a_i)$) and, $min(a_i)$ express that "a_i *should be assigned at least to category* $C_{min(a_i)}$" (denoted $C(a_i) \geq min(a_i)$). For each $a_i \in A^*$, $min(a_i)$ and $max(a_i)$ induce two constraints. As mentioned earlier, when considering UTADIS or ELECTRE TRI (the limits of categories and associated thresholds being known) these constraints are linear.

Let us consider the assignment example $a_i \rightarrow [min(a_i), max(a_i)]$, $a_i \in A^*$. A relaxation of this assignment example consists in assigning a_i to a wider interval $[C_k, C_{k'}]$ such that $k \leq min(a_i)$ or $k' \geq max(a_i)$, with at least one strict inequality. Let us consider the system of inequalities containing the constraints corresponding to *all* the possible relaxations of the assignment example $a_i \rightarrow [C_{min(a_i)}, C_{max(a_i)}]$

(it also contains the constraints corresponding to the original assignment example):

$$\begin{cases} C(a_i) \geq \quad min(a_i) \\ C(a_i) \geq min(a_i) - 1 \\ \quad \vdots \\ C(a_i) \geq \quad 1 \\ C(a_i) \leq \quad max(a_i) \\ C(a_i) \leq max(a_i) + 1 \\ \quad \vdots \\ C(a_i) \leq \quad n_{cat} \end{cases}$$

Consider S^* the set of all indices of constraints induced from a set of assignment examples corresponding to a relaxation of the initial assignment examples. It should be noticed that all the constraints corresponding to a relaxation of one of the two initial constraints are redundant (therefore, S^* contains many redundancies). It should also be remarked that the constraints $C(a_i) \geq 1$ and $C(a_i) \leq n_{cat}$ are trivial and can be removed.

2.2 Illustrative example

Let us consider a situation in which a set of 40 alternatives has to be assigned to 5 categories using the ELECTRE TRI pessimistic method. Each alternative is evaluated on the basis of a set of 7 criteria (see Appendix A). The limits of the categories are known but the criteria importance coefficients are to be defined (see Appendix B). No discordance situation (veto) is considered. Suppose the DM provides assignment examples, where $a_i \rightarrow [C_k, C_{k'}]$ means that alternative a_i must be assigned to a category between C_k and $C_{k'}$, $(k \leq k')$:

- $a_1 \rightarrow C_5$
- $a_{18} \rightarrow C_4$
- $a_{23} \rightarrow [C_2, C_3]$
- $a_{24} \rightarrow [C_2, C_3]$
- $a_{26} \rightarrow C_5$
- $a_{30} \rightarrow C_1$
- $a_{31} \rightarrow C_5$
- $a_{35} \rightarrow [C_1, C_2]$
- $a_{36} \rightarrow C_4$
- $a_{38} \rightarrow C_4$
- $a_{39} \rightarrow C_3$

From these assignment examples, it is possible to define relaxations as in §1.1. These assignment examples and their relaxations generate a set of 41 constraints on the criteria weights w_j, $j = 1, ..., 7$ and cutting level λ.

For instance, $C(a_1) \geq 5$ is equivalent to stating that a_1 is at least as good as b_4 (b_4 being the limit between C_4 and C_5) which amounts to stating the following constraint according to ELECTRE TRI pessimistic rule:

$$\sum_{i=1}^{7} w_i c_i(a_1, b_4) \geq \lambda \Leftrightarrow w_4 + w_5 + w_6 - \lambda \geq 0$$

where w_i denotes the weight of criterion i, λ is the required majority and $c_i(a_1, b_4) \in [0, 1]$ represents the degree to which criterion i agrees that a_1 is at least as good as b_4.

The first 17 constraints correspond to the original assignment examples and the remaining ones correspond to the relaxations (the complete list of constraints is provided in Appendix C). The linear system associated with the assignment examples is infeasible, which means that the information provided by the DM is inconsistent, i.e. there is no way to represent the information in the ELECTRE TRI sorting model.

If we apply the algorithms proposed by Mousseau et al. (2003) (i.e., inconsistency resolution via constraints deletion) to the set S, the solutions correspond to constraints relaxation and/or deletion. It follows from the preceding remark that it is possible to use the algorithm proposed by Mousseau et al. (2003) to solve inconsistencies by relaxation (rather than deletion) of assignment examples. Hence, in the rest of the paper, we will talk in terms of constraints deletion, knowing that it embraces the case of constraints relaxation.

Considering this infeasible linear system, there exist 11 minimal subsets of constraints that resolve the inconsistency, where $I = \{1, 2, ..., 41\}$. These 11 subsets (ordered by cardinality) are listed below. Let us remark that due to the limited size of this example, we have computed all the solutions. When dealing with a large size problem, the DM ought to indicate a maximum number of solutions to be computed.

- $S_1 = \{5, 8, 9, 10, 11, 25, 28, 29\}$
- $S_3 = \{1, 5, 8, 9, 10, 14, 17, 28, 29\}$
- $S_4 = \{1, 5, 8, 9, 10, 14, 25, 28, 29\}$
- $S_5 = \{1, 8, 9, 10, 14, 25, 28, 29, 31\}$
- $S_6 = \{1, 8, 9, 10, 11, 25, 28, 29, 31\}$
- $S_7 = \{5, 7, 9, 10, 11, 13, 17, 28, 29, 30\}$
- $S_8 = \{1, 8, 9, 10, 12, 14, 16, 25, 28, 31, 38\}$
- $S_9 = \{5, 8, 9, 10, 11, 12, 14, 16, 25, 28, 31, 38\}$
- $S_{10} = \{3, 5, 7, 9, 11, 13, 15, 17, 23, 24, 28, 29, 30, 41\}$
- $S_{11} = \{1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 19, 20, 21, 22, 25, 26, 27, 31, 32, 33, 36, 37, 38, 39, 40\}$

To solve the inconsistency, the DM should choose one among these 11 alternative solutions. In order to be presented to the DM, these solutions should be formulated in terms of relaxation of the assignment examples. For instance, S_1 corresponds to:

$$\begin{cases} \text{relax } a_{23} \to [C_2, C_3] & \text{to } a_{23} \to [C_2, C_4] \\ \text{relax } a_{26} \to C_5 & \text{to } a_{26} \to [C_3, C_5] \\ \text{relax } a_{30} \to C_1 & \text{to } a_{30} \to [C_1, C_4] \\ \text{relax } a_{31} \to C_5 & \text{to } a_{31} \to [C_4, C_5] \\ \text{relax } a_{35} \to [C_1, C_2] & \text{to } a_{35} \to [C_1, C_3] \end{cases}$$

We can observe that some of these relaxations correspond to the deletion of one constraint from I (relax $a_{23} \to [C_2, C_3]$ to $a_{23} \to [C_2, C_4]$, constraint 5), while others require the deletion of several constraints from I (relax $a_{26} \to C_5$ to $a_{26} \to [C_3, C_5]$, constraints 8 and 25).

3 Attributing confidence levels to assignment examples

For each assignment example, the DM may be more or less confident in his/her statements. In this section, we will suppose that the DM is able to express confidence judgments. Such confidence judgments can be taken into account by algorithms that identify alternative ways for solving inconsistencies. Intuitively, these algorithms should provide solutions in an order such that the least confident constraints are relaxed/deleted with a higher priority than solutions relaxing high-confidence statements.

3.1 Defining confidence levels

Let us consider a confidence scale Ψ ($\psi_0 \prec \psi_1 \prec \ldots \prec \psi_p \prec \ldots \prec \psi_\tau$). The semantics of this qualitative scale is such that, when facing an inconsistency situation, an assignment example is less likely to be relaxed/deleted when its confidence level is high.

From the DM's perspective, the statements "*a_i should be assigned at most to category $C_{max(a_i)}$*" and "*a_i should be assigned at least to category $C_{min(a_i)}$*" induce two constraints. The DM can attach a confidence level to each of the above mentioned statements. The information will be interpreted as confidence levels attached to the corresponding constraints (for example, $a_1 \to C_2$ implies "a_1 should be assigned at least to C_2" and "a_1 should be assigned at most to C_2" and the DM may have different confidence levels concerning these two statements, e.g., (s)he may say that if a_1 is not assigned to C_2, then it is more likely to be assigned to a higher category than to a lower one). For each relaxed constraint, the attached confidence level is considered to be equal to the confidence level of the original constraint from which it was derived (unless the DM provides specific information).

3.2 A lexicographic ranking procedure

Let us consider an inconsistent set of assignment examples provided by the DM and the set of linear constraints associated with them. Any relaxation of these assignment examples (see §1) will be also considered here. Following the notation introduced previously, m denotes the total number of constraints and $I = \{1, 2, \ldots, i, \ldots, m\}$ denotes the set of indices of these constraints. The resulting polyhedron is, $T_\emptyset = \{x \in \mathbb{R}^n : \sum_{j=1}^n \alpha_{ij} x_j \geq \beta_i, \forall i \in I\} = \emptyset$.

Let I^p denote the subset of constraints whose confidence level is equal to ψ_p. Hence, $I^0, I^1, ..., I^p, ..., I^\tau$ define a partition of I. Furthermore, we will denote $I^{\leq p} = \bigcup_{l=0}^p I^l$ the set of constraints whose confidence level is lower than or equal to ψ_p. Now, consider $S^l \subseteq I^{\leq l}$ a subset of indices of constraints whose confidence level is lower than or equal to ψ_l. We will say that S^l resolves the inconsistency at a confidence level ψ_l if and only if $T_{S^l} = \{x \in \mathbb{R}^n : \sum_{j=1}^n \alpha_{ij} x_j \geq \beta_i, \forall i \in I \setminus S^l\} \neq \emptyset$.

A simple way to account for the confidence level attached to each constraint is to proceed as follows:

1. Identify (by increasing order of cardinality) all minimal subsets $S_1^0, S_2^0, ..., S_{q_0}^0$ that resolve the inconsistency at level ψ_0 (i.e., relaxations whose confidence level is equal to ψ_0 that make the original system of inequalities feasible).
2. Then, identify (by increasing order of cardinality) all minimal subsets $S_1^1, S_2^1, ..., S_{q_1}^1$ that resolve the inconsistency at level ψ_1.
3. Proceed in the same way until finding minimal subsets $S_1^\tau, S_2^\tau, ..., S_{q_\tau}^\tau$ that resolve the inconsistency at level ψ_τ or finding a total number of subsets equal to `maxcount`.

The algorithms presented in Mousseau et al. (2003) can be rather easily adapted to account for the confidence levels lexicographically as outlined above (for a detailed description of the adapted algorithm see Mousseau et al. 2004).

3.3 A penalty based ranking procedure

Another approach to our problem consists in defining a penalty function $\pi(S)$ associated to each subset of constraints indices $S \subseteq I$, and in ranking the subsets that resolve the inconsistency by decreasing penalty order: the larger $\pi(S)$, the greater the dissatisfaction of the DM in removing S from I. This approach generalizes the lexicographic ranking as it is possible to define the penalty function π in such a way that the penalty ranking coincides with the lexicographic ranking.

Given a subset $S \subseteq I$, $S \cap I^p$ denotes the subset of S corresponding to constraints indices whose confidence level is equal to ψ_p, $p = 0, ..., \tau$. Let $|S \cap I^p|$ denote the cardinality of S whose confidence level is equal to ψ_p. In order to define the semantic of the penalty function π, we impose a few suitable conditions on π:

Condition 3.1 *(non-negativity)*
$\forall S \subseteq I,\ \pi(S) \geq \pi(\emptyset) = 0.$

Condition 3.2 *(anonymity)*
$\forall S, S' \subseteq I$, *if* $|S \cap I^p| = |S' \cap I^p|,\ \forall p = 0, ..., \tau$ *then* $\pi(S) = \pi(S')$.

Condition 3.3 *(confidence monotonicity)*
$\forall S, S' \subseteq I$ *such that* $|S \cap I^p| = |S' \cap I^p|,\ p = 1, \ldots, \tau, p \neq u, p \neq v$, *it holds:*

$$\left.\begin{array}{l} |S' \cap I^u| = |S \cap I^u| + 1 \\ |S' \cap I^v| = |S \cap I^v| - 1 \\ u < v \end{array}\right\} \Rightarrow \pi(S) > \pi(S')$$

Condition 3.4 *(cardinality monotonicity)*
$\forall S, S' \subseteq I$, *if* $\forall p = 0, ..., \tau,\ |S \cap I^p| \geq |S' \cap I^p|$ *then* $\pi(S) \geq \pi(S')$.

Among many possible penalty functions, one of the simplest can be defined considering that each constraint in S of a given confidence level ψ_p contributes to increasing $\pi(S)$ by an amount Δ_p (the values Δ_p are to be defined by the DM):

$$\pi(S) = \sum_{p=0}^{\tau} \Delta_p\ |S \cap I^p| \tag{3.1}$$

Given a penalty function π, it is necessary to define an algorithm to rank by increasing penalty the subsets of I that, if removed, lead to a consistent system. The algorithms presented in Mousseau et al. (2003) provide the set of the solutions ordered by cardinality without considering confidence levels. Obviously, the smallest cardinality solutions might not correspond to those of the smallest penalty. Therefore, we can proceed by computing the solutions by increasing cardinality and stop when we are sure that the solutions of a higher cardinality have a greater penalty than the ones we have already obtained. Let S_{tail} be the solution of highest penalty in a list of `maxcount` elements and let $S^{x,p}$ denote an arbitrary set of x constraints, whose confidence levels are all equal to ψ_p. Proposition 2.1 allows us to define the stopping condition $\pi(S_{tail}) \leq \pi(S^{|S_q|,0})$). For a detailed description of this algorithm see Mousseau et al. (2004).

Proposition 3.1. $\forall S \subseteq I,\ S' \subseteq I : |S| \geq |S'|$, *it holds* $\pi(S) \geq \pi(S^{|S'|,0})$ *i.e., the penalty of any solution after the q-th is not lower than the penalty that would be awarded to the q-th solution if all the constraints indexed by* S_q *were of the lowest confidence.*

Proof. From repeatedly using Condition 2.3, $\pi(S) \geq \pi(S^{|S|,0})$. From Condition 2.4, since $|S| \geq |S'|$, it holds that $\pi(S^{|S|,0}) \geq \pi(S^{|S'|,0})$.

3.4 Illustrative example

Let us consider the same example as in §1.2, but now assuming the DM provides associated confidence judgements on a 3-point scale (absolutely confident $\succ$ quite confident $\succ$ not so confident), as follows:

- $a_{35} \to [C_1, C_2]$, absolutely confident
- $a_{18} \to C_4$, $a_{24} \to [C_2, C_3]$, $a_{26} \to C_5$, $a_{36} \to C_4$, quite confident
- $a_1 \to C_5$, $a_{23} \to [C_2, C_3]$, $a_{30} \to C_1$, $a_{31} \to C_5$, $a_{38} \to C_4$, $a_{39} \to C_3$, not so confident

Considering these confidence judgments, the lexicographic ranking provides the solutions, considering first the solutions that remove constraints which are "*not so confident*" (i.e., $I^{\leq 0}$), then the solutions removing constraints that are "*not so confident*" or "*quite confident*" (i.e., $I^{\leq 1}$) and finally the remaining ones. In each group, the solutions are computed by increasing order of cardinality. In our example, no solution exists removing only "*not so confident*" constraints; the first five solutions computed only involve the deletion of constraints that are "*not so confident*" or "*quite confident*":

- $S_3 = \{1, 5, 8, 9, 10, 14, 17, 28, 29\}$
- $S_4 = \{1, 5, 8, 9, 10, 14, 25, 28, 29\}$
- $S_5 = \{1, 8, 9, 10, 14, 25, 28, 29, 31\}$
- $S_7 = \{5, 7, 9, 10, 11, 13, 17, 28, 29, 30\}$
- $S_8 = \{1, 8, 9, 10, 12, 14, 16, 25, 28, 31, 38\}$

If the DM provides a penalty function $\pi(.)$, then the penalty based procedure presented in §2.3 may also be used. For instance, let us consider a penalty function as in (3.1), with $\Delta_0 = 1$, $\Delta_1 = 2$ and $\Delta_2 = 3$. The five best solutions are:

- $S_5 = \{1, 8, 9, 10, 14, 25, 28, 29, 31\}$, $\pi(S_5) = 10$
- $S_1 = \{5, 8, 9, 10, 11, 25, 28, 29\}$, $\pi(S_1) = 11$
- $S_3 = \{1, 5, 8, 9, 10, 14, 17, 28, 29\}$, $\pi(S_3) = 11$
- $S_4 = \{1, 5, 8, 9, 10, 14, 25, 28, 29\}$, $\pi(S_4) = 11$
- $S_7 = \{5, 7, 9, 10, 11, 13, 17, 28, 29, 30\}$, $\pi(S_7) = 11$

Using either of these rankings of solutions, the DM is to chose among the "most promising" solutions to solve inconsistency. Hence, this largely reduces the cognitive effort of the DM to resolve inconsistency.

Conclusion

In this paper, we considered the problem of supporting the DM in the resolution of inconsistent judgments expressed in the form of assignment examples in multiple criteria sorting model. We have proposed the concept of relaxation of an assignment example, which is helpful in this context. To resolve the inconsistency, it is useful to obtain from the DM confidence statements associated with the assignment examples. We have proposed procedures that account for the information to assist the DM in finding the most relevant ways to restore consistency. Although we used ELECTRE TRI for illustrative purposes, our procedures apply to any sorting method for which assignment examples generate linear constraints on the preference-related parameters.

An interesting extension of this work consists in considering the possibility of associating different confidence levels to the original assignment examples constraints and their corresponding relaxation. This extension amounts to considering that the various relaxations of an assignment example are not judged as equivalent as regards their confidence levels.

Acknowledgements. This work has benefited from the Luso-French grant n^o 500B4 (ICCTI / Ambassade de France au Portugal) and program PESSOA 2004 (GRICES/EGIDE). Project POCI/EGE/58371/2004 (FCT / FEDER) also partially supported this research. The third author was also supported by the grant SFRH/BDP/6800/2001.

Appendices

Appendix A: Evaluation matrix (assignment examples only)

	$g_1(a_i)$	$g_2(a_i)$	$g_3(a_i)$	$g_4(a_i)$	$g_5(a_i)$	$g_6(a_i)$	$g_7(a_i)$
a_1	16.4	14.5	59.8	7.5	5.2	5	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
a_{18}	11.7	10	42.1	12.2	4.3	5	2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
a_{23}	12.9	1.9	65	14	7.5	4	3
a_{24}	5.9	-27.7	77.4	16.6	12.7	3	2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
a_{26}	16.7	13.1	73.5	11.9	4.1	2	2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
a_{30}	29.5	8.6	41.8	5.2	6.4	2	3
a_{31}	7.3	-64.5	67.5	30.1	8.7	3	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
a_{35}	-13.3	-31.1	63	21.2	29.1	2	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
a_{36}	6.2	-3.2	46.1	4.8	10.5	2	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
a_{38}	0.1	-9.6	42.5	12.9	12.4	1	1
a_{39}	13.6	9.1	76	17.1	10.3	1	1

Appendix B: Fixed parameters

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
$g_j(b_1)$	−10.0	−60.0	90.0	28.0	40.0	1.0	0.0
$q_j(b_1)$	1.0	4.0	1.0	1.0	0.0	0.0	0.0
$p_j(b_1)$	2.0	6.0	3.0	2.0	3.0	0.0	0.0
$g_j(b_2)$	0.0	−40.0	75.0	23.0	32.0	2.0	2.0
$q_j(b_2)$	1.0	4.0	1.0	1.0	0.0	0.0	0.0
$p_j(b_2)$	2.0	6.0	3.0	2.0	3.0	0.0	0.0
$g_j(b_3)$	8.0	−20.0	60.0	18.0	22.0	4.0	3.0
$q_j(b_3)$	1.0	4.0	1.0	1.0	0.0	0.0	0.0
$p_j(b_3)$	2.0	6.0	3.0	2.0	3.0	0.0	0.0
$g_j(b_4)$	25.0	30.0	35.0	10.0	14.0	5.0	4.0
$q_j(b_4)$	1.0	4.0	1.0	1.0	0.0	0.0	0.0
$p_j(b_4)$	2.0	6.0	3.0	2.0	3.0	0.0	0.0

Appendix C: Constraints stemming from the assignment examples

1. $C(a_1) \geq 5 \Leftrightarrow -\lambda + w_4 + w_5 + w_6 \geq 0$
2. $C(a_{18}) \geq 4 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 \geq 0$
3. $C(a_{18}) \leq 4 \Leftrightarrow \lambda - w_5 - w_6 \geq \varepsilon$
4. $C(a_{23}) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
5. $C(a_{23}) \leq 3 \Leftrightarrow \lambda - w_1 - w_2 - w_4 - w_5 - w_6 - w_7 \geq \varepsilon$
6. $C(a_{24}) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
7. $C(a_{24}) \leq 3 \Leftrightarrow \lambda - w_4 - w_5 \geq \varepsilon$
8. $C(a_{26}) \geq 5 \Leftrightarrow -\lambda + 0.1w_4 + w_5 \geq 0$
9. $C(a_{30}) \leq 1 \Leftrightarrow \lambda - w_1 - w_2 - w_3 - w_4 - w_5 - w_6 - w_7 \geq \varepsilon$
10. $C(a_{31}) \geq 5 \Leftrightarrow -\lambda + w_5 \geq 0$
11. $C(a_{35}) \leq 2 \Leftrightarrow \lambda - w_2 - w_3 - w_4 - w_5 - w_6 \geq \varepsilon$
12. $C(a_{36}) \geq 4 \Leftrightarrow -\lambda + 0.2w_1 + w_2 + w_3 + w_4 + w_5 \geq 0$
13. $C(a_{36}) \leq 4 \Leftrightarrow \lambda - w_4 - w_5 \geq \varepsilon$
14. $C(a_{38}) \geq 4 \Leftrightarrow -\lambda + w_2 + w_3 + w_4 + w_5 \geq 0$
15. $C(a_{38}) \leq 4 \Leftrightarrow \lambda - w_5 \geq \varepsilon$
16. $C(a_{39}) \geq 3 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 \geq 0$
17. $C(a_{39}) \leq 3 \Leftrightarrow \lambda - w_1 - w_2 - w_4 - w_5 \geq \varepsilon$
18. $C(a_1) \geq 4 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
19. $C(a_1) \geq 3 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
20. $C(a_1) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
21. $C(a_{18}) \geq 3 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
22. $C(a_{18}) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$

23. $C(a_{23}) \leq 4 \Leftrightarrow \lambda - w_5 \geq \varepsilon$
24. $C(a_{24}) \leq 4 \Leftrightarrow \lambda - w_5 \geq \varepsilon$
25. $C(a_{26}) \geq 4 \Leftrightarrow -\lambda + w_1 + w_2 + w_4 + w_5 \geq 0$
26. $C(a_{26}) \geq 3 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
27. $C(a_{26}) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
28. $C(a_{30}) \leq 2 \Leftrightarrow \lambda - w_1 - w_2 - w_3 - w_4 - w_5 - w_6 - w_7 \geq \varepsilon$
29. $C(a_{30}) \leq 3 \Leftrightarrow \lambda - w_1 - w_2 - w_3 - w_4 - w_5 - w_7 \geq \varepsilon$
30. $C(a_{30}) \leq 4 \Leftrightarrow \lambda - w_1 - w_4 - w_5 \geq \varepsilon$
31. $C(a_{31}) \geq 4 \Leftrightarrow -\lambda + w_1 + w_5 + w_7 \geq 0$
32. $C(a_{31}) \geq 3 \Leftrightarrow -\lambda + w_1 + w_3 + w_5 + w_6 + w_7 \geq 0$
33. $C(a_{31}) \geq 2 \Leftrightarrow -\lambda + w_1 + 10.75w_2 + w_3 + w_5 + w_6 + w_7 \geq 0$
34. $C(a_{35}) \leq 3 \Leftrightarrow \lambda \geq \varepsilon$
35. $C(a_{35}) \leq 4 \Leftrightarrow \lambda \geq \varepsilon$
36. $C(a_{36}) \geq 3 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 \geq 0$
37. $C(a_{36}) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
38. $C(a_{38}) \geq 3 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 \geq 0$
39. $C(a_{38}) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
40. $C(a_{39}) \geq 2 \Leftrightarrow -\lambda + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \geq 0$
41. $C(a_{39}) \leq 4 \Leftrightarrow \lambda - w_5 \geq \varepsilon$

References

Bana, Costa C (1996) Les problématiques de l'aide à la décision: vers l'enrichissement de la trilogie choix-tri-rangement, *RAIRO/ Recherche Operationnelle* 30(2): 191–216

Bana e Costa C, De Corte J, Vansnick J (2005) On the mathematical foundation of MACBETH. In: Figueira J, Greco S, Ehrgott M (eds) *Multiple criteria decision analysis: State of the art surveys*. Springer, Boston Dordrecht London, pp 409–443

Belacel N (2000) Multicriteria assignment method PROAFTN: methodology and medical application. *European Journal of Operational Research* 125(1): 175–183

Dias L, Mousseau V, Figueira J, Clímaco J (2002) An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI. *European Journal of Operational Research* 138(2): 332–348

Jacquet-Lagrèze E, Siskos Y (1982) Assessing a set of additive utility functions for multicriteria decision making: the UTA method. *European Journal of Operational Research* 10: 151–164

Larichev O, Moskovich H (1994) An approach to ordinal classification problems. *International Transactions in Operational Research* 1(3): 375–385

León T, Liern V (2001) A fuzzy method to repair infeasibility in linearly constrained problems. *Fuzzy Sets and Systems* 122(2): 237–243

Massaglia R, Ostanello A (1991) N-tomic: a support system for multicriteria segmentation problems. In: Korhonen P, Lewandowski A, Wallenius J (eds) *Multiple criteria decision support*. Springer, Berlin, pp. 167–174

Moscarola J, Roy B (1977) Procédure automatique d'examen de dossiers fondée sur une segmentation trichotomique en présence de critères multiples. *RAIRO Recherche Opérationnelle* 11(2): 145–173

Mousseau V, Dias L, Figueira J (2004) Dealing with inconsistent judgments in multiple criteria sorting models. *Research report no.17/2004*, INESC Coimbra, http://www.inescc.pt/ingles/pubinter.php

Mousseau V, Dias L, Figueira J, Gomes C, Clímaco J (2003) Resolving inconsistencies among constraints on the parameters of an MCDA model. *European Journal of Operational Research* 147(1): 72–93

Mousseau V, Slowinski R (1998) Inferring an ELECTRE TRI model from assignment examples. Journal of Global Optimization 12(2): 157–174

Perny P (1998) Multicriteria filtering methods based on concordance/non-discordance principles. *Annals of Operations Research* 80: 137–167
Roodman G (1979) Post-infeasibility analysis in linear programming. *Management Science* 25(9): 916–922
Roy B (1996) *Multicriteria methodology for decision aiding*. Kluwer Academic, Dordrecht
Roy B, Bouyssou D (1993) *Aide Multicritère à la Décision: Méthodes et Cas*. Economica, Paris
Zopounidis C, Zanakis S, Doumpos M (2001) Multicriteria preference disaggregation for classification problems with an application to global investing risk. *Decision Sciences* 32(2): 333–385