Supporting groups in sorting decisions: Methodology and use of a multi-criteria aggregation/disaggregation DSS

Sébastien Damart [a,*], Luis C. Dias [b], Vincent Mousseau [c]

[a] M-Lab — Ecole Normale Supérieure de Cachan, 61 avenue du Président Wilson, 94230 Cachan, France
[b] INESC Coimbra / 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

This paper addresses the situation where a group wishes to cooperatively develop a common multicriteria evaluation model to sort actions (projects, candidates) into classes. It is based on an aggregation/disaggregation approach for the ELECTRE TRI method, implemented on the Decision Support System IRIS. We provide a methodology in which the group discusses how to sort some exemplary actions (possibly fictitious ones), instead of discussing what values the model parameters should take. This paper shows how IRIS may be used to help the group to iteratively reach an agreement on how to sort one or a few actions at a time, preserving the consistency of these sorting examples both at the individual level and at the collective level. The computation of information that may guide the discussion among the group members is also suggested. We provide an illustrative example and discuss some paths for future research motivated by this work.
© 2006 Elsevier B.V. All rights reserved.

Keywords: Classification/sorting; Group decision making/GDSS; Multi-criteria decision making; Imprecise information

1. Introduction

In multicriteria sorting problems [22], a set of actions (projects, candidates, alternatives…) is to be classified into different categories. These categories are ordered and the actions are described by a vector evaluating their performance on multiple criteria. For instance, multiple criteria can be aggregated to sort loan applications into categories such as "Reject", "Accept conditionally", or "Accept". Each action is sorted by comparing its performances with the definitions of the categories, based on the preferences of decision makers. There exist several methods and Decision Support Systems (DSS) for this type of problems, such as ELECTRE TRI, IRIS, PREFDIS, PROAFTN, and TOMASO [20]. However, except for [3], most of the research on group multicriteria decision aiding (e.g., [2,13,14]) deals with choice or ranking problems, rather than sorting. One objective of this paper is then to address sorting problems in group decision settings.

Building a multicriteria sorting model requires defining values for its preference-related parameters. However, the decision maker (DM) often finds it is difficult to express his/her preferences as precise numerical values, correctly taking into account the role played by each parameter. In contexts with multiple DMs, besides phenomena such as leadership emergence or minorities' inhibition, what makes the process harder is that an agreement between the DMs may have to be reached in spite of the diversity of judgments and subjective perceptions of reality.

---

* Corresponding author.
E-mail addresses: damart@mlab.ens-cachan.fr (S. Damart),
lmcdias@fe.uc.pt (L.C. Dias), mousseau@lamsade.dauphine.fr
(V. Mousseau).

This paper proposes a methodology to help DMs interact in order to define a common multicriteria aggregation model indirectly, through the assignment of some exemplary actions (possibly fictitious ones) to categories. For the DMs, agreeing on how to sort these actions is only a means to infer a sorting model that may then be used to sort any other actions. The methodology is based on an aggregation/disaggregation approach [7] suitable to those contexts where numerical information about preferences is hard to obtain. Since the sorting examples correspond to constraints on the parameter values, this type of information is a particular case of what is usually called "incomplete", "imprecise", or "partial" information [9,10,12]. The use of imprecise information in group decision making has been addressed (e.g., [3,4,11,21]), but not using aggregation/disaggregation approaches, which is a second objective of this paper.

We will consider the use of a (single-user) DSS called IRIS [5,6] to support the methodology proposed in this paper. Therefore, we will focus on the case where the aggregation method is ELECTRE TRI, and both the performances of the actions to be sorted and the category limits have been defined a priori: only the criteria weights and a cutting level remain to be set. A brief revision of ELECTRE TRI and the analyses implemented in IRIS are provided in Section 2.

Section 3 introduces the group sorting methodology and the way the current version of IRIS may be used to support the group members and the analyst. In this group setting, each DM may use an instance of the DSS to analyze the problem individually and propose a consistent set of sorting examples, while an analyst may use another instance of the DSS to update a consistent collective model and answer "what-if" questions for the group. By consistency, we mean that there exist parameter values that restitute all the sorting examples. The collective model is (imprecisely) defined by the consistent examples agreed by the group, added in successive discussion rounds. We will suggest some indicators that may be computed to inform the discussion among the DMs, showing where disagreement comes from. It is therefore a methodology that includes elements of sharing, aggregating, and comparing individual information [1].

To illustrate the methodology, an example is provided in Section 4. A concluding section discusses the proposed methodology and the use of IRIS, and presents future research streams motivated by this research.

2. The ELECTRE TRI method and the IRIS decision support system

2.1. A brief presentation of the ELECTRE TRI method

This section outlines the ELECTRE TRI method for sorting actions according to multiple criteria. Readers interested in the method's details can consult, e.g., [5,8,16,18].

Let us denote $A = \{a_1,..., a_m\}$ the set of actions, evaluated on the n criteria $g_1, g_2,..., g_n$, to be sorted into a predefined set of ordered categories $\{C_l,..., C_h\}$ (let C_l denote the worst or lowest category, whereas C_h denotes the best or highest category). Let us denote $b_1, b_2,..., b_{h-1}$ the limit profiles separating the h categories (Fig. 1), b_x being both the lower limit for C_{x+1} and the upper limit of C_x. These limit profiles indicate the performance levels separating the categories on each criterion. ELECTRE TRI sorts each action individually comparing it with the categories profiles.

For each action to sort a and profile b_j, ELECTRE TRI establishes whether a outranks b_j (a is at least as good as b_j), denoted as aSb_j. To establish this, ELECTRE TRI computes n single-criterion concordance indices, which evaluate on a [0,1] scale the agreement of each criterion with the assertion aSb_j, taking into account
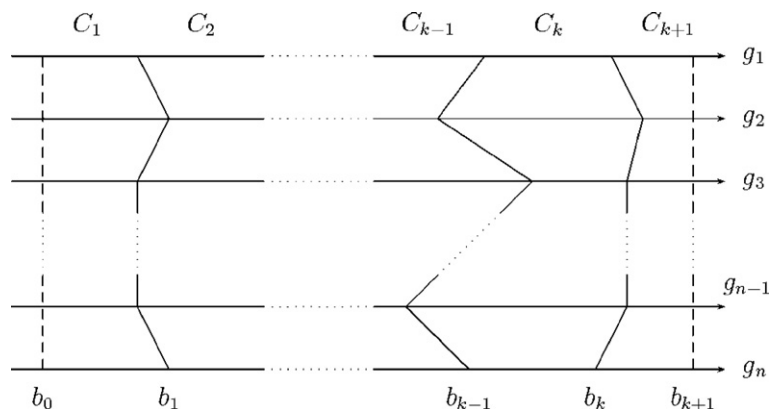

Fig. 1. Definition of categories through limit profiles.

indifference and preference thresholds [19] associated to the criteria. A vector of weights $w_1,..., w_n$ is then used to aggregate these single-criterion concordance indices into a global concordance index. These weights represent the "voting power" of each criterion, not trade-offs as in compensatory aggregation methods. A vector of veto thresholds is also used, in order to compute discordance indices for each criterion against the assertion aSb_j, i.e., the criteria where a is worse than b_j, on a [0, 1] scale. The global index and the discordance indices are then aggregated into a credibility index $s(a, b_j)$, also expressed on a [0, 1] scale. The assertion aSb_j is established if and only if $s(a, b_j) \geq \lambda$, where λ is a cutting level $\lambda \in [0.5, 1]$ (considering the voting power analogy for weights, λ can be interpreted as the required majority).

To assign each action to a category we will consider the pessimistic variant of ELECTRE TRI, which sorts each action a_i into a category C_x such that the action outranks its lower-bound profile $(s(a, b_{x-1}) \geq \lambda$, except $C_1)$ and does not outrank its upper-bound profile $(s(a, b_x) < \lambda$, except $C_h)$.

2.2. Concepts of robust assignment and consistency in IRIS

The standard use of ELECTRE TRI consists in the elicitation of all of the method's parameters, thereby defining a sorting model that assigns each action to a single category. Alternatively, one can follow an aggregation/disaggregation approach [7,15], which consists in inferring part of the preference-related information on the basis of sorting examples provided by the DM, according to his/her holistic appreciation of some actions. The IRIS DSS implements such an approach to avoid the direct elicitation of the weights and cutting level of ELECTRE TRI, considering the remaining aspects of the model (the profiles, the veto, the preference, and the indifference thresholds) have already been elicited. Hereafter, we present only the main features implemented by IRIS. Readers interested in the details may refer to [5,6].

The IRIS DSS is able to infer weights and a cutting level from a set E of sorting examples (which can be provided as intervals, e.g., in left part of Fig. 2, the "ELow" and "EHigh" columns indicate the minimum and maximum categories to which an action can be assigned, according to one DM), i.e., to find a vector of criteria weights $w=(w_1, w_2,..., w_n)$ and a value for λ that restores E with the aggregation rule of ELECTRE TRI (IRIS indicates these values in the right bottom of the screen, see Fig. 2). The right part of Fig. 2 depicts the robust assignment (robust sorting) intervals [3,7]. A sorting interval $[C_x, C_y]$ is said to be robust for an action a_i if the action cannot be assigned to a category lower than C_x, or higher than C_y, for all $(\lambda, w_1,..., w_n) \in \Omega$ (Ω
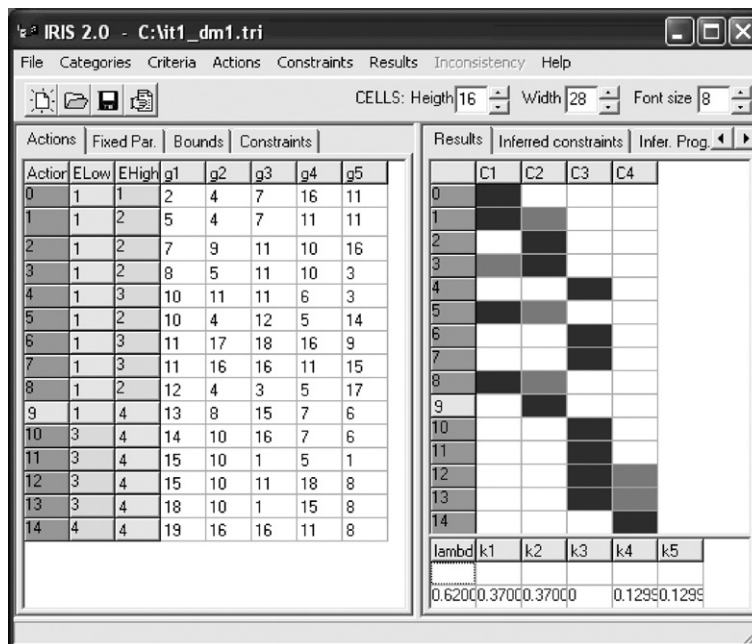

Fig. 2. IRIS screen: examples are given on the left, the inferred values and corresponding assignments, as well as the robust sorting intervals, are given on the right (initial inputs of DM1).

Fig. 3. Inputs of DM1 at a second iteration (after accepting to sort a_{12} into C_3). Note that a_{13} can no longer be assigned into C_4.

represents the set of admissible parameter vectors given E). Adding examples or making them more precise leads to narrowing the robust assignment intervals, possibly until all actions are sorted into single categories. For instance, Fig. 3 shows that if the assignment example C $(a_{12}) \in [C_3, C_4]$ becomes $C(a_{12}) = C_3$, then this results in narrowing the interval for action a_{13} to a single category (in Fig. 2 the interval was two categories wide).

Sometimes, there does not exist any vector $(\lambda, w_1, \dots, w_n) \in \Omega$ that reproduces all the examples through the aggregation rule. For instance, in the situation depicted in Fig. 2, stating that a_{14} is to be sorted into C_3 originates an inconsistency with the examples already introduced. The IRIS DSS includes an inconsistency analysis [17] module that suggests alternatives to remove the inconsistency. Fig. 4 shows that in this situation either the 15th constraint is removed (i.e., stating that a_{14} is to be sorted into C_3), or the example concerning a_{14} can be kept but the examples concerning a_{10} and a_{11} have to be changed (10th and 11th constraints). Hence, IRIS guides the DM to stay consistent by successively choosing one example at a time within the robust assignment ranges, but also allows the DM to test new examples that contradict previous ones, showing what previous judgments must then be revised.

Fig. 4. IRIS inconsistency analysis: constraints implied by the assignment examples are given on the left, possibilities for solving the inconsistency are given on the right.

3. A group decision methodology based on an aggregation/disaggregation approach

3.1. Outline of the process

The methodology we propose is based on the ideas of aggregation/disaggregation methods and methods that deal with imprecise information on parameter values. Although we do not exclude that DMs may agree on other constraints on the parameters (e.g., stating that one criterion weighs more than some other), we here suppose they provide information only about assignment examples. DMs will individually assign some actions (eventually all) to a single category or an interval of categories. Although the proposed methodology can be applied to any multicriteria sorting aggregation method and does not imply the use of IRIS (nor even ELECTRE TRI), we will also introduce the role IRIS can play.

In this context we identify two main difficulties. The first one stems from the possible disagreement among DMs on the different sorting examples they are asked to give. Indeed, DMs do not always have in mind the same relative importance of each criterion, and this influences the way each of them assigns actions to different categories. The second difficulty arises when looking for an agreement on a set of sorting examples that is consistent, i.e., that can be reproduced by the method using a vector of suitable parameter values. This second difficulty leads us to separate the individual consistency from the collective consistency. The first one is concerning the information each DM is providing. The second type of consistency is concerning the information all the DMs agree on. Simple examples show that reducing disagreement on sorting examples without being collectively consistent is possible when the DMs are staying individually consistent through the whole process. For this reason, we propose a procedure that consists in maintaining both collective and individual consistency throughout the process. The methodology that we suggest takes into account these two following issues:

- the necessity to make the whole group converge toward a collective set of robust assignments and finally a common vector of inferred weights;
- the necessity to make DMs being and staying collectively as well as individually consistent.

To cope with the last requirement, we suggest that each group member should build his/her model using IRIS (each one using his own instance of the program), whereas an analyst uses another instance of IRIS to build a group model, based on the agreements reached by the group.

We propose a process based on progressively agreeing on the assignment of one or a few actions at a time. The agreed assignment may not be a precise one, since DMs are allowed to agree on conclusions such as "a_{10} can be assigned only to C_2 or to C_3". Let us define:

- $L(a_i)_t$ and $U(a_i)_t$ are the lower and upper bound (respectively) for assigning action a_i according to the group's agreement, at iteration t.

At the outset, $L(a_i)_1 = C_l$ and $U(a_i)_1 = C_h$, $\forall a_i \in A$, i.e., there are no agreed examples. At each iteration t ($t \geq 1$), the outcome of each discussion round is to narrow the interval of categories $[L(a_i)_t, U(a_i)_t]$ for at least one of the actions a_i, if possible making $L(a_i)_t = U(a_i)_t$. An agreement on a precise (or imprecise) assignment for some action a_i introduces constraints on the parameter values, which in turn will constrain the interval of possible assignments for the remaining actions, in order to maintain consistency. Hence, by agreeing on an assignment the DMs should be aware of the implications that the assignment has concerning other actions, and this can be easily observed by each DM when inserting a potential assignment in IRIS. The iterative process is the following:

Step 1: Each DM gives his/her individual set of consistent sorting examples and determines the corresponding robust sorting intervals. IRIS can be used by each DM to solve possible inconsistencies and to compute the robust assignment intervals.

Step 2: Aided by an analyst who gathers the robust sorting intervals from all the DMs, the group discusses in order to agree on at least one assignment. If the group is not able to agree on any example among the multiple possibilities available, then this iterative process ends.

Step 3: The agreed sorting example or examples are incorporated in the collective model and all the individual models. If the group feels the collective model is satisfactory, then the procedure stops. Otherwise, return to Step 2.

At any stage, each DM may privately revise his/her individual model by adding, deleting, or modifying examples that have not yet been settled by the group, again using IRIS. Moreover, an important aspect is that the methodology does not preclude the DMs from collectively reneging on a previous agreement, if they conclude they prefer to agree on an assignment that contradicts a previously agreed one.

This iterative sequence of discussion rounds may end in different ways. Ideally, it will end because the DMs managed to agree on how to sort all the actions into precise categories. It may also end because the DMs find

they cannot progress any further, although an agreement was not reached regarding all the actions. In this case, the analyst may suggest the DMs to accept an inferred vector of parameter values able to reproduce all the assignments they agreed on, and chosen according to the stability criterion used by IRIS [6]. However, assisting groups that are not cooperative enough to follow this procedure is outside the scope of this paper.

The goal of the process may be not only to sort all the actions in A, but also, and perhaps more importantly, to find an inferred vector of parameter values able to reproduce the assignments that were agreed. Indeed, the set of example actions will usually be a subset of a much larger set of actions to which the method will be applied. For instance, DMs working for a bank may discuss a sample of past loan applications and try to agree on how they would be sorted, aiming at using the resulting sorting model as a standard for sorting applications that arrive in the future.

3.2. Implications of agreements for the individual models

A key issue in the methodology outlined above is the coexistence of internally consistent individual and collective models, each defined by possibly different sets of sorting examples. The collective model is defined by all the assignments that have been agreed on by the group. The individual models reflect the sorting examples of each group member and should not contradict the collective model. This means that some group members may have to change their individual models as a result of the assignments that the group agreed to change in the collective model. Let us define:

- Let $L_k(a_i)_t$ and $U_k(a_i)_t$ denote the lower bound and upper bound (respectively) for assigning action a_i according to the group member DM_k at iteration t (obviously, $L_k(a_i)_t \leq U_k(a_i)_t$).

The implications of the agreements for the individual models are not restricted to the actions that were subject of the group's agreement. For instance, let us imagine that at iteration t, the group agreed to sort action a_i into category C_3. Three situations may occur for DM_1:

- He/she might have already have $L_1(a_i)_t = U_1(a_i)_t = C_3$, hence nothing would change in his/her individual model;
- He/she might have $L_1(a_i)_t = C_3$, $U_1(a_i)_t = C_4$, and to place $L_1(a_i)_{t+1} = U_1(a_i)_{t+1} = C_3$ does not contradict his/her individual model, but may imply reducing the sorting possibilities of other actions (this was exemplified with IRIS in Section 2.2, cf. Figs. 2 and 3);

- He/she might have $L_1(a_i)_t = U_1(a_i)_t = C_4$, and to place $L_1(a_i)_{t+1} = U_1(a_i)_{t+1} = C_3$ implies revising some of his individual sorting examples to restore consistency, possibly with implications in other actions besides a_i (in Section 2.2 — Figs. 2 and 4 — we exemplified with IRIS that accepting $C(a_{14}) = C_3$ also would imply revising the examples concerning a_{10} and a_{11}).

3.3. Computations to inform the discussion in Step 2

At the second step of each round of the methodology, the DMs will try to reach agreements on how to assign at least one of the exemplary actions. We will now suggest two simple indicators that may be computed in order to inform the DMs and structure this discussion, allowing the group to know what actions are more (or less) likely to generate agreements. These indicators, hence, have informative rather than normative value.

The first idea we propose is to compute the proportion of DMs that accept each possible assignment. A straightforward measure of the support for assigning a_i to C_x is:

$$E(a_i, C_x)_t = \frac{\sum_{k=1}^{K} E_k(a_i, C_x)_t}{K} \times 100\%,$$
$$\text{with } E_k(a_i, C_x)_t$$
$$= \begin{cases} 1, & \text{if } C_x \in [L_k(a_i)_t, U_k(a_i)_t] \\ 0, & \text{if } C_x \notin [L_k(a_i)_t, U_k(a_i)_t] \end{cases}.$$

This natural definition, however, places an interesting question. Let us for instance imagine that (in a sorting problem involving 3 categories) the DMs provide the following support for an action a_1: $E(a_1, C_1)_t = 50\%$, $E(a_1, C_2)_t = 0\%$, and $E(a_1, C_3)_t = 50\%$. This might be interpreted as if the DMs unanimously agree that a_1 will not be assigned to C_2. However, assigning a_1 to C_2 seems to be a natural proposal for consensus reaching, in a "split the difference" type of agreement.

For this reason we suggest a modified version for the support computation that yields a "unimodal" distribution:

$$E'(a_i, C_x)_t$$
$$= \begin{cases} E(a_i, C_x)_t, & x = 1 \vee x = h \\ \max\{E(a_i, C_x)_t, & 1 < x < h \\ \min\{\max_{y<x} E(a_i, C_y)_t, & \\ \max_{y>x} E(a_i, C_y)_t\}\}, & \end{cases} \times 100\%$$

Fig. 5. Fixed values for the profiles and criteria thresholds.

In this case, we would have $E'(a_1, C_1)_t = E'(a_1, C_2)_t = E'(a_1, C_3)_t = 50\%$, highlighting the fact that the assignment of a_1 to C_2 should not be less considered than the assignment of a_1 to C_1 or to C_3. The sum of the support for the possible assignments of an action may be greater than 100% for this reason, and also because each DM may provide intervals as assignment examples.

Given the support computed for each assignment possibility, the analyst may invite the group to focus on the assignments with highest support values. Alternatively, the analyst may focus on the lowest support

values and suggest the group to agree on the exclusion of such assignments. The current version of IRIS may help by generating a report file indicating its users' robust assignment intervals, which can be imported into a spreadsheet to automatically calculate the $E'(a_i, C_k)_t$ values for all possible assignments.

A second idea we suggest is try to measure the "cost" or "effort" incurred by each DM when accepting a collective assignment, through what we have called the number of "shifts". One shift corresponds to change an action's assignment to one category above or below the interval accepted by the DM. To exemplify this concept, consider that DM_1's examples were those depicted in Fig. 2. Then, agreeing for instance that a_{12} is assigned to C_3 would cost no shift, and the same would happen if the assignment was to C_4. Assigning a_{14} to C_3 would cost one shift, assigning it to C_2 would cost 2 shifts, etc. Besides these shifts, the ones implied in the remaining actions by the need to remain consistent also have to be taken into account. Hence, assigning a_{14} to C_3 would cost not one shift, but a total of 3 shifts, taking into account that a_{10} and a_{11} would have to be sorted into category C_2, as IRIS would show (Fig. 4).

This measure accounts for the number and "extent" of the changes that a DM has to make in his inputs to accept an agreement and remain consistent. This is a rough measure; for instance, it can be criticized for assuming that two shifts in one action are equivalent to one shift for two actions, and

Fig. 6. Initial inputs of DM2.

for disregarding the information contained in ELECTRE TRI's credibility indices. However, it is simple to compute and to explain. More sophisticated measures may be sought, but to accurately model the DM's preferences concerning concessions may well be an elusive goal. It would be possible to compute other indicators, such as the sum of the "lost support" or the maximum of "lost support", where "lost support" is the support for the assignments that become infeasible when agreeing on a potential assignment.

4. A short illustrative example

4.1. The decision problem

For the sake of illustrating the concepts and to show how a group might proceed, let us consider a hypothetical problem of a bank that decided to use the ELECTRE TRI method to sort loan applications. A team of four experts (the DMs) met to build a common sorting model to be used by all the bank branches. Let us suppose that 15 past loan applications (the actions), which have been evaluated on 5 criteria (g_1,..., g_5) using a 0–20 scale, were taken as examples. The four experts should agree on how to sort each action. We denote C_1, C_2, C_3 and C_4 the 4 categories and a_i ($i=0$,..., 14) the actions to sort (Fig. 2). In this example the thresholds are constant for all the profiles and no veto thresholds were

used (Fig. 5). DMs have the possibility to give for each action only one assignment (one category) or a range of possible assignments (an interval of categories). The four DMs have different ways to judge the actions, as depicted in Fig. 2 (DM1) and Figs. 6–8 (DM2, DM3, and DM4).

4.2. Application of the methodology

4.2.1. Iteration 1

Let us suppose that the results for the acceptability of the different assignments were those presented in Fig. 9. For example, the possibility $C(a_4)=C_3$ yields 75% of agreement among DMs, i.e., 3 out of 4 DMs agree that action a_4 can be assigned to category C_3. If DMs agree on this assignment it means that 25% of them (here it represents 1 DM) will have to make a concession, that is to say at least one "shift" from one category to another category. The assignment $C(a_{12})=C_3$ does not require any shift from any DM, i.e., all have placed C_3 in the interval assignments of a_{12}. Its acceptability is 100%. Observing this, the DMs agreed to add $C(a_{12})=C_3$ to their individual models (Fig. 3 illustrates this for DM1), and the analyst placed the same example in the collective model. They also noticed that it was unanimous that a_0 and a_1 cannot reach C_4, a_2 can be assigned only to C_2 or C_3, etc.
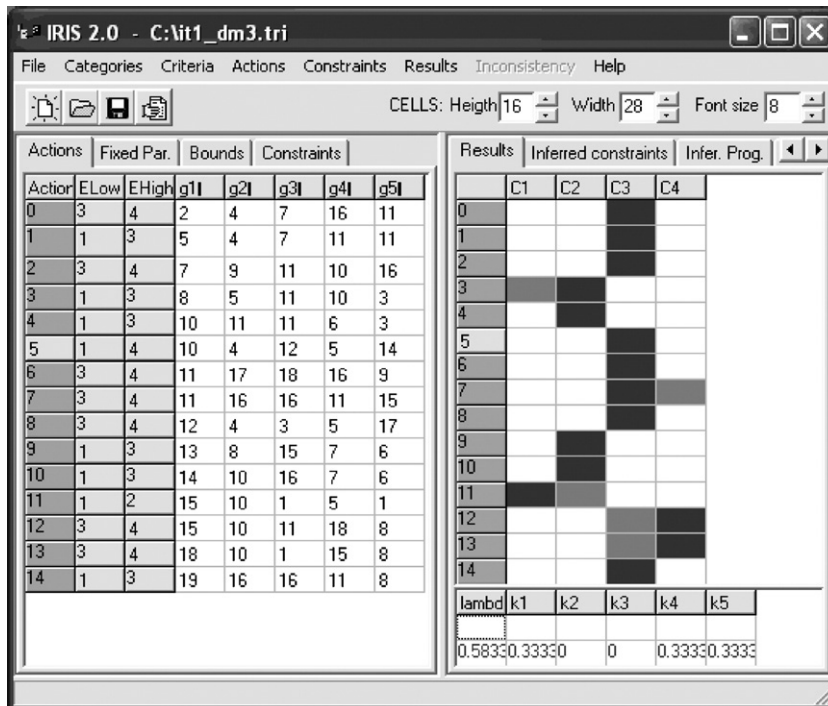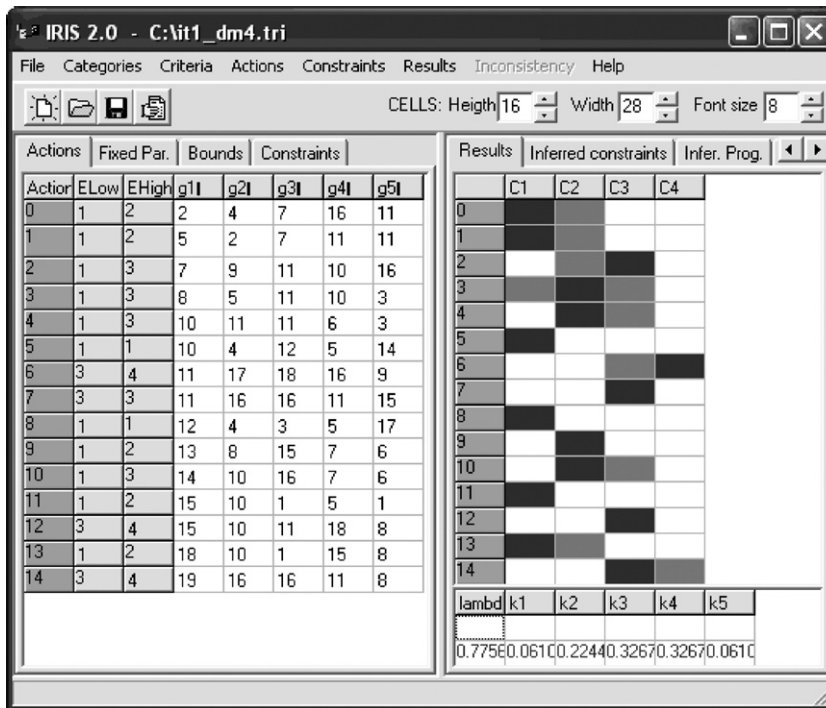

Fig. 7. Initial inputs of DM3.

Fig. 8. Initial inputs of DM4.

4.2.2. Iteration 2

The analyst studied the implications of fixing each of the 11 assignments that were acceptable by 3 out of 4 DMs (75%). By using IRIS, he computed the number of shifts that each DM would need to perform to accept the assignment, considering the implied changes to other assignments resulting from the need of staying consistent (see Table 1). For instance, if a_3 is assigned to C_1 then DM2 (Fig. 6) has to make two shifts concerning that action (since he had placed $C(a_3)=C_3$). Furthermore, the fact that $C(a_3)=C_1$ implies $C(a_9) \leq C_2$ requires two additional shifts (since DM2 had placed $C(a_9)=C_4$), and the fact that $C(a_3)=C_1$

Fig. 9. Acceptability of the different assignments.

Table 1
Needed shifts for each possible assignment with 75% of acceptability (iteration 2)

Assignment	Implied constraints	DM1	DM2	DM3	DM4	Total
$C(a_1)=C_2$	$C(a_0) \leq C_2$	0	0	2	0	2
$C(a_2)=C_3$	None	1	0	0	0	1
$C(a_3)=C_1$	$C(a_9) \leq C_2$, $C(a_{10}) \leq C_3$	0	5	0	0	5
$C(a_3)=C_2$	None	0	1	0	0	1
$C(a_4)=C_3$	$C(a_0) \leq C_2$, $C(a_1) \leq C_2$, $C(a_{10}) \geq C_3$, $C(a_{14})=C_4$	0	0	5	0	5
$C(a_6)=C_3$	$C(a_3) \leq C_2$	0	2	0	0	2
$C(a_7)=C_3$	None	0	1	0	0	1
$C(a_8)=C_1$	None	0	0	2	0	2
$C(a_9)=C_2$	$C(a_{10}) \leq C_3$	0	3	0	0	3
$C(a_{11})=C_1$	None	2	0	0	0	2
$C(a_{14})=C_4$	$C(a_0) \leq C_2$, $C(a_1) \leq C_2$, $C(a_4)= C_3$, $C(a_{10}) \geq C_3$	0	0	5	0	5

implies $C(a_{10}) \leq C_3$ requires one more shift (since DM2 had placed $C(a_{10})=C_4$), totaling 5 shifts. The numbers of shifts for each DM seem to indicate that DM2 and DM3 are the ones further away from a group consensus.

This brief analysis shows that the easiest concessions could come from DM1 (accepting that $C(a_2)=C_3$), or DM2 (either $C(a_3)=C_2$ or $C(a_7)=C_3$). DM4 does not need to make any concession. Given these results, we can imagine DM2 would offer to accept $C(a_3)=C_2$, because he prefers to offer a concession now and avoid the prospect of $C(a_3)=C_1$, which also has an acceptability of 75%. Furthermore, he noticed the poor performance of a_3 on all the criteria except the third (the one he cared most about and that justified his initial position of placing a_3 in a higher category). All DMs incorporated the example $C(a_3)=C_2$ in their individual models. The same example was added to the collective model.

Table 2
Needed shifts for each possible assignment with 75% of acceptability (iteration 5)

Assignment	Implied constraints	DM1	DM2	DM3	DM4	Total
$C(a_0)=C_2$	$C(a_{11}) \leq C_2$	3	0	0	0	3
$C(a_4)= C_3 \Leftrightarrow C(a_{14})=C_4$	$C(a_0) \leq C_2$, $C(a_1) \leq C_2$, $C(a_{10}) \geq C_3$	0	0	5	0	5
$C(a_6)=C_3$	None	0	1	0	0	1
$C(a_7)=C_3$	None	0	1	0	0	1
$C(a_8)=C_1$	None	0	0	2	0	2
$C(a_9)=C_2$	$C(a_{10}) \leq C_3$	0	3	0	0	3
$C(a_{11})=C_1$	None	2	0	0	0	2

4.2.3. Iteration 3

The agreed example does not have much impact on the other actions. The easiest concessions are the same as before, which could come from DM1 (accepting $C(a_2)= C_3$) or DM2 (accepting $C(a_7)=C_3$). This time DM1 was the one to concede, hence the collective model, as well as the individual models, were updated by placing the example $C(a_2)=C_3$.

4.2.4. Iterations 4 and 5

At iteration 4, of the remaining assignments acceptable by 75% of the DMs, only DM2 can make concessions costing only one shift. Due to the former concessions each DM has made, the discussion led to an agreement to make $C(a_1)=C_2$ (a concession by DM3) at iteration 4 and $C(a_7)=C_3$ (a concession by DM2), at iteration 5.

4.2.5. Iteration 6

The shifts in Table 2 for the remaining 7 possible assignments with 75% acceptability did not change. Now, the easiest concession is again from DM2 ($C(a_6)=C_3$). However, this DM objects that he has just made a concession. DM1 was then persuaded to place $C(a_{11})=C_1$, given its poor performance (below level 6) on the 3rd, 4th, and 5th criteria. DM1 was receptive to that argument, provided that the same argument was accepted to place $C(a_8)= C_1$, given its poor performance (below level 6) on the 2nd,

Table 3
Needed shifts for each possible assignment with 50% of acceptability or more (iteration 7)

Assignment	Implied constraints	DM1	DM2	DM3	DM4	Total
$C(a_0)=C_1^*$	None	0	1	1	0	2
$C(a_0)=C_2^{**}$	None	1	0	0	0	1
$C(a_4)=C_2^* \Leftrightarrow C(a_{10})=C_2^* \Leftrightarrow C(a_{14})=C_3^*$	$C(a_9)=C_2$	2	5	1	0	8
$C(a_4)=C_3^{**} \Leftrightarrow C(a_{14})=C_4^{**}$	$C(a_{10}) \geq C_3$	0	0	3	0	3
$C(a_5)=C_1^*$	$C(a_9)=C_2$, $C(a_{10}) \leq C_3$	1	5	0	0	6
$C(a_5)=C_2^*$	$C(a_9)=C_2$, $C(a_{10}) \leq C_3$	0	4	0	1	5
$C(a_5)=C_3^*$	None	1	0	0	2	3
$C(a_6)=C_3^{**}$	None	0	1	0	0	1
$C(a_6)=C_4^{**}$	None	1	0	0	0	1
$C(a_9)=C_2^{**}$	$C(a_{10}) \leq C_3$	0	3	0	0	3
$C(a_{10})=C_3^*$	$C(a_4)=C_3$, $C(a_9)=C_2$, $C(a_{14})=C_4$	0	3	3	0	5
$C(a_{13})=C_1^*$	None	2	0	2	0	4
$C(a_{13})=C_2^*$	None	1	1	1	0	3
$C(a_{13})=C_3^*$	None	0	2	0	1	3

*50% acceptability; **75% acceptability.

3rd, and 4th criteria, requiring the agreement of DM3. The group agreed to add the examples $C(a_{11})=C(a_8)=C_1$ in their individual models and in the collective model.

We will end the example here, but the procedure would continue until either the group feels the collective model is satisfactory or the group is not able to agree on any further example. The situation at this point is presented in Table 3, where it can be seen that that the assignments of a_0, a_6 and a_{13} do not have any impact on the remaining actions, whereas the assignment of the pair a_4, a_{14} is crucial and has an impact on a_9 and a_{10}.

5. Conclusions

The methodology we suggest extends an aggregation/disaggregation approach based on the use of an ELECTRE TRI method to a context with multiple decision makers. It aims at supporting interaction among DMs, helping them to reach a common and accepted unique multicriteria aggregation model in a context where numerical information is hard to obtain. Hence, it extends current literature on group decision aiding by addressing sorting problems and by using the ideas of aggregation/disaggregation.

We believe that the use of imprecise information (assignment of some actions as examples that are translated to constraints on the weights) rather than precise numbers for the weights brings two important benefits. First, it contributes to avoid the cognitively difficult task of eliciting some of the model's parameters. Second, it contributes to ground the discussion on assignment examples (that is a rather natural form of expression for DMs), rather than on weight values, whose understanding may be problematic. Moreover, we can note that there is a many-to-one correspondence from parameter values to results (i.e., different input values may yield the same output).

The proposed process should not be seen as a precise "recipe" or method to be rigidly followed. Rather, it may be seen as a methodology that may be flexibly adjusted by a facilitator–analyst, where the computed outputs serve as an orientation for the discussion among the group members. As a matter of fact, the methodology is general enough to be applied to other multicriteria sorting methods based on aggregation/disaggregation, namely UTADIS [22].

The example we have built illustrates how the group may progress towards a common and unique ELECTRE TRI model while maintaining consistency along the whole process both at the individual and at the collective levels. This is the main contribution brought by the use of the IRIS DSS. By using an individual instance of the software to privately build his/her model, each DM is helped to keep a consistent set of examples. Moreover, the DM can check the impact on his/her model that results from agreeing to an assignment example being discussed by the group. At the same time, the analyst uses an instance of IRIS to maintain a consistent set of examples agreed by the group, and may use the software to answer "what-if" questions or to verify whether a "package agreement" involving the assignment of several actions simultaneously is consistent or not.

We deem that the a Delphi-like procedure, where DMs can present arguments for or against some assignments and then each one is invited to change his/her inputs, given the arguments presented, is the most desirable interaction mode for the discussion stage. However, it is not impossible that, as suggested in the example, DMs may bargain or exchange concessions.

The discussion-focusing suggestions of maximizing the support for the assignment examples and minimizing the number of shifts can be joined by other measures. The main issue here is that these measures are meant to focus the discussion and suggest agreements, rather than as voting schemes to aggregate individual examples, which are always hard to justify. Indeed, the latter perspective, that of an automatic arbitration, should be used only as a last resort and when there remain few actions to be sorted. If this were not the case, then the possibility of manipulation would be a true concern, for instance. By considering the criteria's role to be more suggestive rather than normative, strategic misrepresentation (which may always exist) is less prone to be a problem to the group. Moreover, the requirement of consistency in the set of assignment examples makes it more difficult to misrepresent the assignment of an action being discussed without being contradictory with the way the remaining actions are sorted.

Future research is still needed for the cases where the group is not cooperative enough to proceed as suggested in this paper. One idea is to study what are the best computed measures/rules for choosing or imposing assignment examples. In such cases the computed measures would become normative, with the associated risks indicated above. Another idea could consist in identifying different coalitions of DMs (i.e., subgroups of DMs with similar preferences), and inferring a different model for each coalition. At the end of such a process, it could be interesting to look for robust conclusions [3] acceptable by all the coalitions.

Another future development envisaged by us is to develop a GDSS based on IRIS. Firstly, this means adding networking capabilities to the software, namely to facilitate sending the assignment examples information from the DMs to the analyst. Secondly, this means adding an additional module for the analyst to perform the computations for which we used a spreadsheet. We feel that the level of interaction among group members recommends same-time/same-place meetings, but the GDSS can be built without this assumption.

Acknowledgements

References

[1] V. Belton, J. Pictet, A framework for group decision using a MCDA model: sharing, aggregating, or comparing individual information? Journal of Decision Systems 6 (1997) 283–303.

[2] A. Davey, D. Olson, Multiple criteria decision making models in group decision support, Group Decision and Negotiation 7 (1998) 55–75.

[3] L.C. Dias, J.N. Clímaco, ELECTRE TRI for groups with imprecise information on parameter values, Group Decision and Negotiation 9 (5) (2000) 355–377.

[4] L.C. Dias, J.N. Clímaco, Dealing with imprecise information in group multicriteria decisions: a methodology and a GDSS architecture, European Journal of Operational Research 160 (2) (2005) 291–307.

[5] L.C. Dias, V. Mousseau, IRIS — interactive robustness analysis and parameters' inference for multicriteria sorting problems (Version 2.0) — user manual, documents of INESC Coimbra, No. 1/2003, April 2003 (Available online http://www4.fe.uc.pt/lmcdias/IrisMan2.pdf).

[6] L.C. Dias, V. Mousseau, IRIS: a DSS for multiple criteria sorting problems, Journal of Multi-Criteria Decision Analysis 12 (2003) 285–298.

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

[8] J. Figueira, V. Mousseau, B. Roy, ELECTRE methods, in: J. Figueira, S. Greco, M. Ehrgott (Eds.), Multiple Criteria Decision Analysis: State of the Art Surveys, Springer Verlag, Boston, 2005, pp. 133–162.

[9] P.C. Fishburn, Analysis of decisions with incomplete knowledge of probabilities, Operations Research 13 (1965) 217–237.

[10] G.B. Hazen, Partial information, dominance, and potential optimality in multiattribute utility theory, Operations Research 34 (1986) 297–310.

[11] S.H. Kim, B.S. Ahn, Interactive group decision making procedure under incomplete information, European Journal of Operational Research 116 (1999) 498–508.

[12] C.W. Kirkwood, R.K. Sarin, Ranking with partial information: a method and an application, Operations Research 33 (1985) 38–48.

[13] C. Macharis, J.P. Brans, B. Mareschal, The GDSS PROMETHEE procedure, Journal of Decision Systems 7 (1998) 283–307.

[14] N.F. Matsatsinis, A.P. Samarras, MCDA and preference disaggregation in group decision support systems, European Journal of Operational Research 130 (2001) 414–429.

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

[16] V. Mousseau, R. Slowinski, P. Zielniewicz, A user-oriented implementation of the ELECTRE TRI method integrating preference elicitation support, Computers & Operations Research 27 (7–8) (2000) 757–777.

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

[18] B. Roy, D. Bouyssou, Aide Multicritère à la Décision: Méthodes et Cas, Economica, Paris, 1993.

[19] B. Roy, Ph. Vincke, Relational systems of preference with one or more pseudo-criteria: some new concepts and results, Management Science 30 (1984) 1323–1335.

[20] H. Roland Weistroffer, Charles H. Smith, Subhash C. Narula, Multiple criteria decision support software, in: J. Figueira, S. Greco, M. Ehrgott (Eds.), Multiple Criteria Decision Analysis: State-of-the-Art Survey, vol. 2, Springer, 2004.

[21] A.A. Salo, Interactive decision aiding for group decision support, European Journal of Operational Research 84 (1995) 134–149.

[22] C. Zopounidis, D. Doumpos, Multicriteria classification and sorting problems: a literature review, European Journal of Operational Research 138 (2002) 229–246.

Sébastien Damart received his PhD degree in Management Sciences from the University of Paris Dauphine. He is now associate professor at the Institute of Technology of the University 'Evry Val d'Essonne' where he teaches management sciences, organization theory and logistics. He is also a member of the M-Lab research unit of 'Dauphine Recherche en Management' where he develops his research interests which are implementation of operational research models in organizational specific contexts, multicriteria decision aiding models and participation-based decision aiding tools.

Luis Cândido Dias has a first degree in Informatics Engineering and a PhD in Management/Systems Sciences from the University of Coimbra. He is with the Faculty of Economics of the University of Coimbra, where he teaches Informatics, Decision Analysis and Operations Research courses. He is also a researcher and one of the directors of the INESC Coimbra R&D institute. His interests include DSS, multicriteria decision support, supporting group decision and negotiations, and robustness analysis. He has published numerous papers in journals such as Computers and Operations Research, European J. of Operational Research, Group Decision and Negotiation, J. of Multi-Criteria Decision Analysis, and J. of the Operational Research Society.

Vincent Mousseau received his MS degree, PhD degree and Habilitation in computer Science from the University of Paris Dauphine (UPD). He is currently associate professor at UPD and teaches operations research and decision sciences and heads a master "decision and computer science". He is member of the Lamsade Research Laboratory. His research interests deal with Multiple Criteria Decision Aid, Preference Modeling and Elicitation, the implementation of MCDA methodologies in real world applications. He published numerous papers in journals such as EJOR, Annals of OR, Computers and OR, DSS, Journal of global optimization, 4OR, ….