CAHIER DU LAMSADE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision
(Université Paris-Dauphine)
Unité de Recherche Associée au CNRS ESA 7024

USING ASSIGNMENT EXAMPLES TO INFER CATEGORY LIMITS FOR THE ELECTRE TRI METHOD

CAHIER N° 168
mai 2000

An NGO THE [1]
Vincent MOUSSEAU [1]

reçu : juillet 1999.

---

[1] *LAMSADE, Université Paris-Dauphine, Place du Maréchal De Lattre de Tassigny, 75775 Paris Cedex 16, France ({ngothe,mousseau}@lamsade.dauphine.fr).*

Contents

UTILISATION D'EXEMPLES D'AFFECTATION POUR INFERER LIMITES DES CATEGORIES DANS LA METHODE ELECTRE TRI

Résumé

Le problème du tri d'un ensemble d'actions consiste à affecter chaque action à une des catégories pré-définies. Dans cet article, nous nous intéressons au problème de tri multicritère et, plus précisément, à la méthode ELECTRE TRI. Cette méthode nécessite l'évaluation de paramètres (coefficients d'importance, seuils de discrimination et de veto, limites des catégories, ...) pour construire le modèle des préférences du décideur. L'évaluation directe de ces paramètres étant difficile, [MS98] a proposé une approche interactive d'agrégration-désagrégation qui infère les paramètres d'ELECTRE TRI indirectement à partir d'informations hollistiques, i.e., d'exemples d'affectation. Dans cette approche, la détermination des paramètres d'ELECTRE TRI qui restituent au mieux les exemples d'affectation est formulée à travers un programme d'optimisation non-linéaire. Dans cette même direction, [MFN97] a considéré le sous-problème visant à déterminer uniquement les coefficients d'importance (les seuils et les limites des catégories étant fixés). Ce sous-problème se traduit par un programme linéaire.

Nous continuons l'idée d'un modèle d'inférence partiel en considérant le sous-problème complémentaire qui détermine les seuils et les limites des catégories (les coefficients d'importance étant fixés). Avec une certaine simplification, il se traduit aussi par un programme linéaire. Avec le résultat de [MFN97], nous avons un couple de modèles complémentaires qui peuvent être combinés dans une approche interactive qui infère les paramètres d'un modèle ELECTRE TRI à partir des exemples d'affectation. Dans chaque interaction, le décideur peut réviser ses exemples d'affectation, donner une information supplémentaire et choisir les paramètres à fixer avant de redémarer la phase d'optimisation.

Mots clés : Problématique du tri, Méthode ELECTRE TRI, Evaluation des limites des catégories, Procédure d'inférence.

USING ASSIGNMENT EXAMPLES TO INFER CATEGORY LIMITS FOR THE ELECTRE TRI METHOD

Abstract

Given a finite set of alternatives, the sorting (or assignment) problem consists in the assignment of each alternative to one of the pre-defined categories. In this paper, we are interested in multiple criteria sorting problems and, more precisely, in the existing method ELECTRE TRI. This method requires the elicitation of preferential parameters (importance coefficients, thresholds, profiles,...) in order to construct the DM's preference model. A direct elicitation of these parameters being rather difficult, [MS98] proposed an interactive aggregation-disaggregation approach that infer ELECTRE TRI parameters indirectly from hollistic information, i.e., assignment examples. In this approach, the determination of ELECTRE TRI parameters that best restitute the assignment examples is formulated through a non-linear optimization program. Also in this direction, [MFN97] considered the subproblem of the determination of the importance coefficients only (the thresholds and category limits being fixed). This subproblem leads to solve a linear program.

We continue the idea of partial inference model by considering the complementary subproblem which determines the category limits (the importance coefficients being fixed). With some simplification, it also leads to solve a linear program. Together with the result of [MFN97], we have a couple of complementary models which can be combined in an interactive approach infering the parameters of an ELECTRE TRI model from assignment examples. In each interaction, the DM can revise his/her assignment examples, to give additional information and to choose which parameters to fix before the optimization phase restarts.

Keywords : Sorting problem, ELECTRE TRI method , Category limits' elicitation, Inference procedure.

1 Introduction

According to [Roy85], real world decision problems can be classified in three basic problematics: choice ($P.\alpha$), sorting ($P.\beta$) and ranking ($P.\gamma$). The sorting problematic consists in formulating the decision problem in terms of the assignment of a set of potential alternatives $A = \{a_1, a_2, ..., a_N\}$ to one of the pre-defined ordered categories $C_1, C_2, ..., C_p, C_{p+1}$. The assignment of an alternative a to the appropriate category relies on the intrinsic value of a, and not on the comparison of a with other alternatives.

In this paper, we are interested in the multiple criteria sorting problematic and, more precisely, in the ELECTRE TRI method (see [Yu92a], [MSZ00] and [RB93]). The use of this method requires the determination of several parameters such as: limit profiles between consecutive categories, importance coefficients of criteria, discrimination thresholds, ... The set of these parameters (that we will call an ELECTRE TRI model in this paper) is used to construct a preference model that the Decision Maker (DM) accept as a working hypothesis. In many situations, it is difficult for the DM to determine these values; a direct evaluation of these parameters requires an important cognitive effort. To overcome this difficulty, [MS98] proposed an indirect approach in order to infer these parameters from assignment examples through a certain form of regression on assignment examples. The proposal of [MS98] consists in a global inference model which infers all ELECTRE TRI parameters simultaneously starting from assignment examples. In this approach, the determination of the parameters' values that best fit the assignment examples results from the resolution of a non-linear mathematical program. This optimization procedure is integrated into an interactive tool that enables the DM to react on the set of obtained parameters and to get insights into his/her preferences. In the continuation of this idea, [MFN97] proposed a partial inference approach consisting in the introduction of a subproblem that infers the importance coefficients and the cutting level only. In this case, the mathematical program to be solved becomes linear.

Our work also accounts for the idea of inferring a subset of ELECTRE TRI parameters from assignment examples. We consider the problem of determining the definition of categories (limit profiles and discrimination thresholds), the importance coefficients being fixed. Our paper presents a new inference procedure that determines the category limits from assignment examples. This procedure is validated by numerical results obtained in a laboratory experiment aiming at testing the operational usefulness of the category limits' inference procedure in an interactive process. Hence, together with the former results of [MFN97] and [MS98], the present paper tends to enrich the different approaches to determine parameters of an ELECTRE TRI model (see Figure 1).

Figure 1 reveals the fact that the use of our inference model can be considered in a broader scheme where all ELECTRE TRI parameters are to be inferred. In such a situation, the two partial inference models, the inference of importance coefficients ([MFN97]) and the inference of category limits can be used iteratively. At each iteration, the DM can revise his/her assignment examples, give additional information and choose which parameters to fix before the optimization phase restarts.

The paper is organized as follows. In section 2, we recall briefly the ELECTRE TRI method and the general inference scheme. In the next two sections, we present the two phases of the category limit inference model. In section 5, we consider some variations of the model when more information is available or when a strong consistency is required. Section 6 is dedicated to the experimental design and the empirical results. A final section groups conclusions.

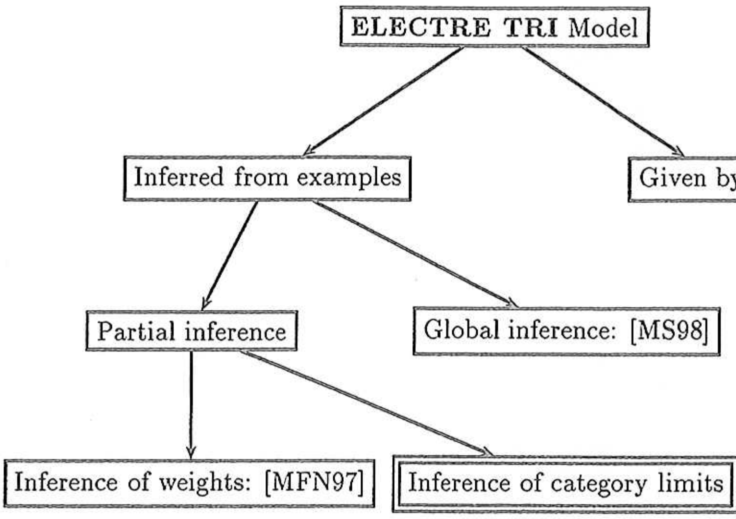

Figure 1: Different approaches to determine ELECTRE TRI parameters

2 The inference procedure

2.1 Brief presentation of the ELECTRE TRI method

We give here a very brief overview of the ELECTRE TRI method and define some notations that will be used. A complete description can be found in [RB93]. The corresponding software is described in [MSZ00]

ELECTRE TRI is a multiple criteria sorting method used to assign alternatives to predefined ordered categories. The assignment of an alternative a results from the comparison of a with the profiles defining the limits of the categories. Let A denote the set of alternatives to be assigned and $A^* \subset A, A^* = \{a_1, a_2, ...a_n\}$ denote a subset of alternatives that the DM intuitively assigns to a category or a range of categories (A^* contains the assignment examples given by the DM) and let $K = \{1, 2, ...n\}$ be the set of indices of the alternatives from A^*. Let F denote the set of the indices of the criteria $g_1, g_2, ..., g_m$ (F=$\{1, 2, ..., m\}$), k_j the importance coefficient of the criterion g_j, B the set of indices of the profiles defining p+1 categories (B=$\{1, 2, ..., p\}$), b_h being the upper limit of category C_h and the lower limit of category C_{h+1}, h=1, 2, ...,p. Each profile b_h is characterized by its performances $g_j(b_h)$ and its thresholds $p_j(b_h)$ (preference thresholds), $q_j(b_h)$ (indifference thresholds) and $v_j(b_h)$ (veto thresholds). In what follows, we will assume, without any loss of generality, that preferences increase with the value on each criterion and that $\sum_{j \in F} k_j = 1$.

Further on, we use $a \to C_h$ to denote that the alternative a is assigned to the category C_h, when necessary, $a \to_{DM} C_h$ is used to highlight the fact that the assignment is given by the DM.

ELECTRE TRI builds a valued 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 [RV84] for details on this double thresholds preference representation). Beside the intra-criterion preferential information, represented by the indifference and preference thresholds $q_j(b_h)$ and $p_j(b_h)$, the construction of S also makes use of two types of inter-criterion preferential information:

- the set of weight-importance coefficients ($\{k_j, j \in F\}$) is used in the concordance test when computing the relative importance of the coalitions of criteria being in favour of the assertion aSb_h (and b_hSa);
- the set of veto thresholds ($\{v_j(b_h), j \in F, h \in B\}$) is used in the discordance test; $v_j(b_h)$ represents the smallest difference $g_j(b_h) - g_j(a)$ 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 "defuzzyfied" using a so-called λ-cut: the assertion aSb_h (b_hSa, respectively) is considered to be valid if the credibility index of the fuzzy outranking relation is greater than a "cutting level" λ such that $\lambda \in (0.5, 1]$. This λ-cut determines the preference situation between a and b_h.

Two assignment procedures (optimistic and pessimistic) are available, their role being to analyse the way in which an alternative a compares to the profiles so as to determine the category to which a should be assigned. The result of these two assignment procedures differs when the alternative a is incomparable with at least one profile b_h.

2.2 Scheme of the general inference procedure

The general scheme of the inference procedure proposed in [MS98] and [MSZ00] (see Figure 2) is to find an ELECTRE TRI model as compatible as possible with the assignment examples (A^*) given by the DM. 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 from A^* in the same way as the DM did.

In order to minimize the differences between the assignments made by ELECTRE TRI and the assignments made by the DM, an optimization procedure is used. The DM can tune up the model in the course of an interactive procedure. He/she may either revise the assignment examples or fix values (or intervals of variation) for some model parameters. The DM may modify (A^*) as well as introduce some more constraints concerning the profiles.

When the model is not perfectly compatible with the assignement examples, the procedure should be able to detect all "hard cases", i.e., the alternatives for which the assignment computed by the model strongly differs from the DM's assignment. The DM could then be asked to reconsider his/her judgment. This general scheme is applicable for the global inference ([MS98]) as well as partial inference procedures ([MFN97] and this paper). For more discussion on the procedure and its interest, see [MS98], [MSZ00] and [MFN97].

2.3 Formulation of the problem

In what follows, we will confine our analysis to the case were the pessimistic assignment procedure is used and no veto phenomenon occurs ($v_j(b_h) = \infty, \forall j \in F, \forall h \in B$). As the importance coefficients are fixed, the inferred parameters are the category limits (i.e., the limit profiles: $g_j(b_h)$ as well as the thresholds

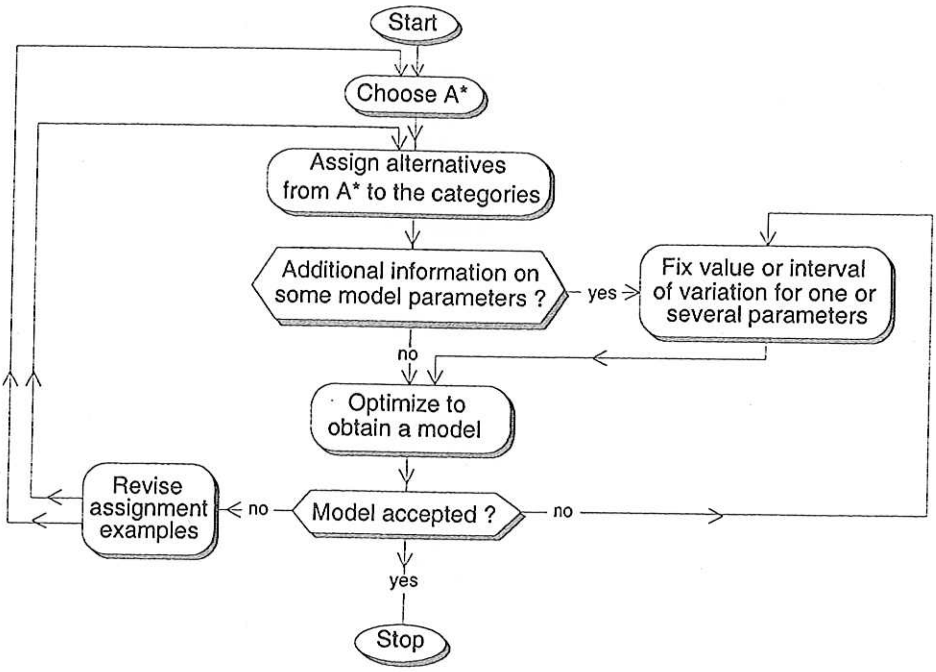

Figure 2: General scheme of the inference procedure ([MS98], [MSZ00])

$q_j(b_h)$ and $p_j(b_h), j \in F, h \in B$) and the cutting level λ.

Because of the complexity of the computation of the valued outranking relation in the ELECTRE TRI method, it is difficult to infer the category limits directly. Therefore, we propose to decompose the computation in two phases:

- Phase 1: local concordance indices $c_j(a, b_h), c_j(b_h, a), j \in F, h \in B$ are determined by means of a linear program.
- Phase 2: $g_j(b_h), p_j(b_h), q_j(b_h), \forall j \in F, \forall h \in B$, are assessed so as to be compatible with the indices computed in phase 1.

3 Phase 1: determination of local concordance indices $c_j(a_k, b_h)$ and $c_j(b_h, a_k)$

3.1 Basic notations and hypothesis

In the ELECTRE TRI method, the construction of the outranking relation S is based on the agregation-disaggregation paradigm which is materialized by the local concordance indices $c_j(a_k, b_h), c_j(b_h, a_k), j \in F, k \in K, h \in B$, and then by the global concordance indices $\sigma(a_k, b_h), \sigma(b_h, a_k), k \in K, h \in B$. We recall that the hypothesis of no veto is assumed in our approach. The following observations are straightforward from ELECTRE TRI:

$$\left.\begin{array}{l}\sigma(a_k,b_h)=c(a_k,b_h)=\sum_{j\in F}k_jc_j(a_k,b_h)\\ \sigma(b_h,a_k)=c(b_h,a_k)=\sum_{j\in F}k_jc_j(b_h,a_k)\end{array}\right\}\forall k\in K,\forall h\in B \tag{1}$$

$$c_j(a_k,b_h),c_j(b_h,a_k)\in[0,1],\forall j\in F,\forall k\in K,\forall h\in B \tag{2}$$

When g_j is a quasi-criterion (i.e., $p_j = q_j$), (2) becomes

$$c_j(a_k,b_h),c_j(b_h,a_k)\in\{0,1\},\forall j\in F,\forall k\in K,\forall h\in B \tag{3}$$

$$\left.\begin{array}{l}g_j(a_k)<g_j(a_l)\Rightarrow\left\{\begin{array}{l}c_j(a_k,b_h)\le c_j(a_l,b_h)\\ c_j(b_h,a_k)\ge c_j(b_h,a_l)\end{array}\right.\\ g_j(a_k)=g_j(a_l)\Rightarrow\left\{\begin{array}{l}c_j(a_k,b_h)=c_j(a_l,b_h)\\ c_j(b_h,a_k)=c_j(b_h,a_l)\end{array}\right.\end{array}\right\}\forall j\in F,\forall k,l\in K,\forall h\in B \tag{4}$$

$$\left.\begin{array}{l}c_j(a_k,b_{h+1})\le c_j(a_k,b_h)\\ c_j(b_h,a_k)\le c_j(b_{h+1},a_k)\end{array}\right\}\forall j\in F,\forall k\in K,h=1,2,...,p-1 \tag{5}$$

$$\left.\begin{array}{l}c(a_k,b_{h+1})=\sum_{j\in F}c_j(a_k,b_{h+1})\le\sum_{j\in F}c_j(a_k,b_h)=c(a_k,b_h)\\ c(b_h,a_k)=\sum_{j\in F}c_j(b_h,a_k)\le\sum_{j\in F}c_j(b_{h+1},a_k)=c(b_{h+1},a_k)\end{array}\right\}\forall k\in K,h=1,2,...,p-1 \tag{6}$$

In the ELECTRE TRI method, the pessimistic procedure assigns the alternative a_k to the category C_{h_k} ($a_k \to C_{h_k}$) iff

$$\left\{\begin{array}{l}c(a_k,b_{h_k-1})\ge\lambda\\ c(a_k,b_{h_k})<\lambda\end{array}\right. \tag{7}$$

3.2 Some results justifying the inference model

Definition 3.1 *For each criterion $g_j, j \in F$ and each profile $b_h, h \in B$, the function $\phi_{jh}(x)$ is called the category limit characterization function.*

$$\phi_{jh}(x)=\left\{\begin{array}{ll}0 & \text{if } x\le g_j(b_h)-p_j(b_h)\\ \frac{x-g_j(b_h)+p_j(b_h)}{p_j(b_h)-q_j(b_h)} & \text{if } g_j(b_h)-p_j(b_h)<x<g_j(b_h)-q_j(b_h)\\ 1 & \text{if } g_j(b_h)-q_j(b_h)\le x\le g_j(b_h)+q_j(b_h)\\ \frac{g_j(b_h)+p_j(b_h)-x}{p_j(b_h)-q_j(b_h)} & \text{if } g_j(b_h)+q_j(b_h)<x<g_j(b_h)+p_j(b_h)\\ 0 & \text{otherwise}\end{array}\right. \tag{8}$$

It is obvious from the definition 3.1 that

$$\phi_{jh}(a_k)=\min\{c_j(a_k,b_h),c_j(b_h,a_k)\}^{1} \tag{9}$$

This function plays a central role in our approach. As illustrated in Figure 3, it represents a fuzzy membership of the relation $(a_k I_j b_h)$. It also represents all the local concordance indices that can be used to reconstruct the category limits.

[1]Formally, we have to use $\phi_{jh}(g_j(a_k))$ instead of $\phi_{jh}(a_k)$. But here, we use $\phi_{jh}(a_k)$ in order to simplify the notation.

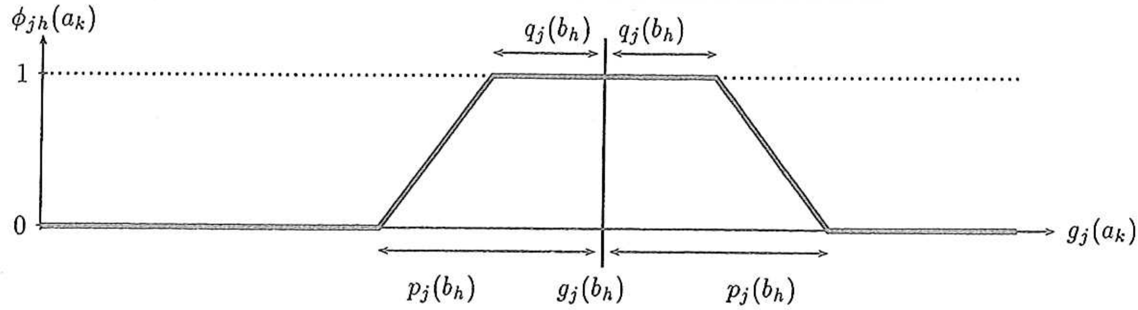

Figure 3: Local concordance indices and category limits characterized by $\phi_{jh}(a_k)$

Proposition 3.1 *The symmetrical condition: the category limit characterization function $\phi_{jh}(a_k)$ is symmetrical through the vertical line $x = g_j(b_h)$.*

The demonstration is straightforward from the definition.

To ensure the consistency of the categories, we base on the condition "no alternative should be indifferent to more than one profile" (see [Yu92b] and [Yu92a]). We have to express this condition by constraints concerning local concordance indices in order to introduce it into our program. We have the following results:

Proposition 3.2 *When using the ELECTRE TRI pessimistic procedure, if $a_k \to C_{h_k}$ then:*
(i) - $\forall h \geq h_k, \neg a_k I b_h$
(ii) - $\forall h < h_k - 1, b_h S a_k \Rightarrow b_h I a_k$
i.e., The indifference between a_k and b_h appears only for $h \leq h_k - 1$ when $b_h S a_k$ takes place.

Demonstration
$a_k \to C_{h_k} \Leftrightarrow [c(a_k, b_{h_k}) < \lambda$ and $c(a_k, b_{h_k-1}) \geq \lambda]$
(i) $c(a_k, b_{h_k}) < \lambda \Rightarrow \neg a_k S b_{h_k}$
$\Rightarrow \forall h > h_k, c(a_k, b_h) \leq c(a_k, b_{h_k}) < \lambda \Rightarrow \neg a_k S b_h$
$\Rightarrow \forall h > h_k, \neg a_k I b_h$
(ii) $c(a_k, b_{h_k-1}) \geq \lambda \Rightarrow a_k S b_{h_k-1}$
$\Rightarrow \forall h \leq h_k - 1, c(a_k, b_h) \geq c(a_k, b_{h_k-1}) \geq \lambda \Rightarrow a_k S b_h$
$\Rightarrow \forall h \leq h_k - 1, b_h S a_k \Rightarrow b_h I a_k$

■

Proposition 3.3 *If $\exists h_0$ s.t. $\neg b_{h_0} S a_k$, then $\forall h \leq h_0, \neg b_h S a_k$*

Demonstration $\neg b_{h_0} S a_k \Rightarrow c(b_{h_0}, a_k) < \lambda$
$\Rightarrow \forall h \leq h_0, c(b_h, a_k) \leq c(b_{h_0}, a_k) < \lambda \Rightarrow \neg b_h S a_k$

■

Proposition 3.4 *The two following conditions are equivalent when using the pessimistic assignment procedure :*
(i) No alternative in A^* is indifferent to more than one profile.
(ii) $\forall a_k \in A^*, a_k \to C_{h_k},$ *then* $\neg b_{h_k-2} S a_k$

Demonstration
$(i) \Rightarrow (ii)$
Let b_{h_1} the profile (unique if it exists) s.t. $a_k I b_{h_1}$
By 3.2, $h \geq h_k \Rightarrow \neg a_k I b_h$, therefore $a_k I b_{h_1} \Rightarrow h_1 \leq h_k - 1$
By definition, $a_k I b_{h_1} \Rightarrow a_k S b_{h_1}$ and $b_{h_1} S a_k$
If $h_1 < h_k - 1$, then $c(b_{h_k-1}, a_k) \geq c(b_{h_1}, a_k) \geq \lambda \Rightarrow b_{h_k-1} S a_k \Rightarrow a_k I b_{h_k-1}$, contradictory to the unicity of h_1.
Therefore, $h_1 = h_k - 1$. As $h_k - 2 < h_k - 1 \Rightarrow [\neg a_k I b_{h_k-2}$ and $a_k S b_{h_k-2}] \Rightarrow \neg b_{h_k-2} S a_k$.
$(ii) \Rightarrow (i)$
$\neg b_{h_k-2} S a_k \Rightarrow \forall h \leq h_k - 2, \neg b_h S a_k$
Let h_1 be the profile s.t. $a_k I b_{h_1}$. Then, $h_1 > h_k - 2$ and $h_1 \leq h_k - 1 \Rightarrow h_1 = h_k - 1$

$\square$

Remark: This condition ensures only the consistency in the set A^*, not in A. It is weaker than the following condition which is too strong (sufficient but not necessary) to ensure the consistency in A.

$$g_j(b_{h+1}) \geq g_j(b_h) + p_j(b_h) + p_j(b_{h+1}) \tag{10}$$

3.3 Variables of the problem

In ELECTRE TRI pessimistic assignment procedure, an alternative a_k is assigned to category C_h ($a_k \to C_h$) iff $c(a_k, b_{h-1}) \geq \lambda$ and $c(a_k, b_h) < \lambda$. To ensure the consistency of the profiles, we need the condition $c(b_{h_k-2}, a_k) < \lambda$.

Let us suppose that the DM has assigned the alternative $a_k \in A^*$ to category C_{h_k} ($a_k \to_{DM} C_{h_k}$). Let us define the slack variables x_k, y_k and z_k unrestricted in sign such that $c(a_k, b_{h_k-1}) - x_k = \lambda, c(a_k, b_{h_k}) + y_k = \lambda$ and $c(b_{h_k-2}, a_k) + z_k = \lambda$.

These slack variables are used only as auxiliary variables to construct the objective function. Then, we can eliminate them easily by using $\beta = \min_{a_k \in A^*} x_k, y_k, z_k$. So they are not introduced explicitely in the program. Therefore, the optimization problem will include the following variables (2mnp+2):

$c_j(a_k, b_h), c_j(b_h, a_k), \forall j \in F, \forall k \in K, \forall h \in B$	local concordance indices (2mnp)
λ	cutting level (1)
β	(1)

For technical reasons (see 3.6), we propose the introduction of two fictitious alternatives into A^*: an ideal alternative a^* which is obviously assigned to the best category C_{p+1} and an anti-ideal one a_* to the worst category C_1.
$a_* : \forall j \in F, g_j(a_*) < \min_{a_k \in A^*} g_j(a_k)$
$a^* : \forall j \in F, g_j(a^*) < \min_{a_k \in A^*} g_j(a_k)$
These two alternatives ensure that there is always a transition between 0 and 1 in the set of the local

concordance indices within each criterion. Indeed, from the definition of a^* and a_*, it is obvious that:

$$\left.\begin{array}{l}(c_j(a_*,b_h)=0) \text{ and } (c_j(b_h,a_*)=1)\\ (c_j(a^*,b_h)=1) \text{ and } (c_j(b_h,a^*)=0)\end{array}\right\}\forall j\in F,\forall h\in B \tag{11}$$

3.4 Accuracy criterion

If the values of the slack variables x_k, y_k and z_k are all positive, then ELECTRE TRI pessimistic assignment procedure will assign alternative a_k to the "correct" category and the consistency of categories is respected. If, however, x_k or y_k is negative, the ELECTRE TRI pessimistic assignment procedure will assign alternative a_k to a "wrong" category. If z_k is negative, the consistency of categories is not respected (a_k can be indifferent to two consecutive profiles). The lower the minimum of these values, the less adapted is the ELECTRE TRI model to give an account of the assignment of a_k made by the DM. Moreover, if x_k, y_k and z_k are all positive, then a_k is assigned consistently with the DM's statement, and the consistency is respected for all $\lambda' \in [\lambda - \min\{y_k, z_k\}, \lambda + x_k]$.

Let us now consider the set of alternatives $A^* = \{a_1, a_2, ..., a_k, ..., a_n\}$ and suppose that the DM has assigned the alternative a_k to the category C_{h_k}, $\forall a_k \in A^*$. The ELECTRE TRI model will be consistent with the DM's assignments iff $x_k \geq 0, y_k \geq 0$ and $z_k \geq 0$, $\forall a_k \in A^*$.

Consistently with the preceding argument, an accuracy criterion to be maximized can be defined as: $\beta = \min_{a_k \in A^*} \{x_k, y_k, z_k\}$.

3.5 Optimization problem to be solved

In order to replace the strict inequalities ($<$ or $>$) by ($\leq$ or $\geq$), we introduce an arbitrary small constant ϵ. From the results invoked in sections 3.1 and 3.2, we obtain the following constraints:

Bounds of variables

$0.5 \leq \lambda \leq 1$

$0 \leq c_j(a_k, b_h) \leq 1, \forall j \in F, \forall k \in K, h \in B$

$0 \leq c_j(b_h, a_k) \leq 1, \forall j \in F, \forall k \in K, h \in B$

Other constraints

$$\begin{array}{ll}
\max\{c_j(a_k,b_h), c_j(a_h,b_k)\} = 1, \forall j\in F, \forall k \in K, h\in B & \text{(mnp)}\\
\left.\begin{array}{l} c_j(a_k,b_h) \leq c_j(a_l,b_h) \text{ if } g_j(a_k) < g_j(a_l)\\ c_j(a_k,b_h) = c_j(a_l,b_h) \text{ if } g_j(a_k) = g_j(a_l)\end{array}\right\}\forall j\in F,\forall k,l\in K, h\in B & \text{(m(n-1)p)}\\
\left.\begin{array}{l} c_j(b_h,a_k) \geq c_j(b_h,a_l) \text{ if } g_j(a_k) < g_j(a_l)\\ c_j(b_h,a_k) = c_j(b_h,a_l) \text{ if } g_j(a_k) = g_j(a_l)\end{array}\right\}\forall j\in F,\forall k,l\in K, h\in B & \text{(m(n-1)p)}\\
c_j(a_k,b_{h+1}) \leq c_j(a_k,b_h), \forall j \in F, \forall k\in K, h=1,2,...,p-1 & \text{(mn(p-1))}\\
c_j(b_h,a_k) \leq c_j(b_{h+1},a_k), \forall j \in F, \forall k\in K, h=1,2,...,p-1 & \text{(mn(p-1))}\\
\beta \leq \sum_{j\in F} k_j c_j(a_k, b_{h_k-1}) - \lambda, \forall k\in K & \text{(n-1)}\\
\beta + \epsilon \leq \lambda - \sum_{j\in F} k_j c_j(a_k, b_{h_k}), \forall k\in K & \text{(n)}\\
\beta + \epsilon \leq \lambda - \sum_{j\in F} k_j c_j(b_{h_k-2}, a_k), \forall k\in K & \text{(n-2)}
\end{array}$$

These constraints suffer from two weaknesses:

- They do not take into account the symmetrical condition (see proposition 3.1).
- The condition $\max\{c_j(a_k,b_h), c_j(a_h,b_k)\} = 1$ is non-linear.

By observing that almost all variables $c_j(a_k, b_h), c_j(b_h, a_k)$ are 0 or 1 (see [Nau96]), we accept the integrity hypothesis to obtain a approximate preliminary solution of the problem. Under this hypothesis, we replace $c_j(a_k, b_h), c_j(b_h, a_k) \in [0,1]$ by $c_j(a_k, b_h), c_j(b_h, a_k) \in \{0,1\}$. Hence, the constraint $max\{c_j(a_k, b_h), c_j(b_h, a_k)\} = 1$ becomes $c_j(a_k, b_h) + c_j(b_h, a_k) \geq 1$. This hypothesis helps to overcome these two weaknesses as the program turns out to be a linear one and the verification of the symmetry condition can be postponed to the next phase which determinines the profiles and the thresholds. However, making this hypothesis implies that the solution will formally not be optimal.

The basic form of the optimization problem to be solved is the following:
Program $P1$

$$\max \beta \tag{12}$$

s.t.

$$\begin{aligned}
\beta &\leq \sum_{j \in F} k_j c_j(a_k, b_{h_k - 1}) - \lambda, \forall k \in K & (13)\\
\beta + \epsilon &\leq \lambda - \sum_{j \in F} k_j c_j(a_k, b_{h_k}), \forall k \in K & (14)\\
\beta + \epsilon &\leq \lambda - \sum_{j \in F} k_j c_j(b_{h_k - 2}, a_k), \forall k \in K & (15)\\
1 &\leq c_j(a_k, b_h) + c_j(b_h, a_k), \forall j \in F, \forall k \in K, h \in B & (16)\\
c_j(a_k, b_{h+1}) &\leq c_j(a_k, b_h), \forall j \in F, \forall k \in K, h = 1, 2, ..., p-1 & (17)\\
c_j(b_{h+1}, a_k) &\geq c_j(b_h, a_k), \forall j \in F, \forall k \in K, h = 1, 2, ..., p-1 & (18)\\
c_j(a_k, b_h) &\leq c_j(a_l, b_h), \forall j \in F, \forall k, l \in K, h \in B, \text{ if } g_j(a_k) < g_j(a_l) & (19)\\
c_j(a_k, b_h) &= c_j(a_l, b_h), \forall j \in F, \forall k, l \in K, h \in B, \text{ if } g_j(a_k) = g_j(a_l) & (20)\\
c_j(b_h, a_k) &\geq c_j(b_h, a_l), \forall j \in F, \forall k, l \in K, h \in B, \text{ if } g_j(a_k) < g_j(a_l) & (21)\\
c_j(b_h, a_k) &= c_j(b_h, a_l), \forall j \in F, \forall k, l \in K, h \in B, \text{ if } g_j(a_k) = g_j(a_l) & (22)\\
0.5 &\leq \lambda \leq 1 & (23)\\
c_j(a_k, b_h) &\in \{0,1\}, \forall j \in F, \forall k \in K, h \in B & (24)\\
c_j(b_h, a_k) &\in \{0,1\}, \forall j \in F, \forall k \in K, h \in B & (25)
\end{aligned}$$

The program obtained is a MIP (Mixed Integer Program) which contains 2mnp+2 variables and 4n+3mp+2 constraints. As we mentioned previously, the slack variables x_k, y_k, z_k can be eliminated from the problem formulation since they are defined by the constraints (13), (14) and (15).

3.6 Refinement of the result

In the previous paragraph, we introduced the integrity hypothesis to simplify the problem. It is important to check whether it is possible to improve the result by relaxing the integrity condition for some values $c_j(a_k, b_h)$ or $c_j(b_h, a_k)$.

We relax, in the previous problem, the integrity constraint ($c_j(a_k, b_h), c_j(b_h, a_k) \in \{0, 1\}$ is replaced by $c_j(a_k, b_h), c_j(b_h, a_k) \in [0, 1]$), other values becoming constants (already determined by the program $(P1)$). It is quite natural to consider the points to relax in the neighborhood of the transition between 0 and 1. For each criterion g_j and each profile b_h, we define the following values:

$$\begin{cases} z_{2jh} = \max\{g_j(a_k | a_k \in A^*, c_j(a_k, b_h) = 0\} \\ z_{1jh} = \max\{g_j(a_k | a_k \in A^*, g_j(a_k) < z_{2jh}\} \text{ or } -\infty \text{ if the set is empty} \\ z_{3jh} = \min\{g_j(a_k | a_k \in A^*, c_j(a_k, b_h) = 1\} \\ t_{2jh} = \min\{g_j(a_k | a_k \in A^*, c_j(b_h, a_k) = 0\} \\ z_{1jh} = \max\{g_j(a_k | a_k \in A^*, c_j(b_h, a_k) = 1\} \\ z_{3jh} = \min\{g_j(a_k | a_k \in A^*, g_j(a_k) > t_{2jh}\} \text{ or } +\infty \text{ if the set is empty.} \end{cases} \quad (26)$$

With the insertion of the ideal alternative a^* and the anti-ideal alternative a_*, the existence of the values $z_{2jh}, z_{3jh}, t_{1jh}, t_{2jh}$ is ensured (see Figure 4).

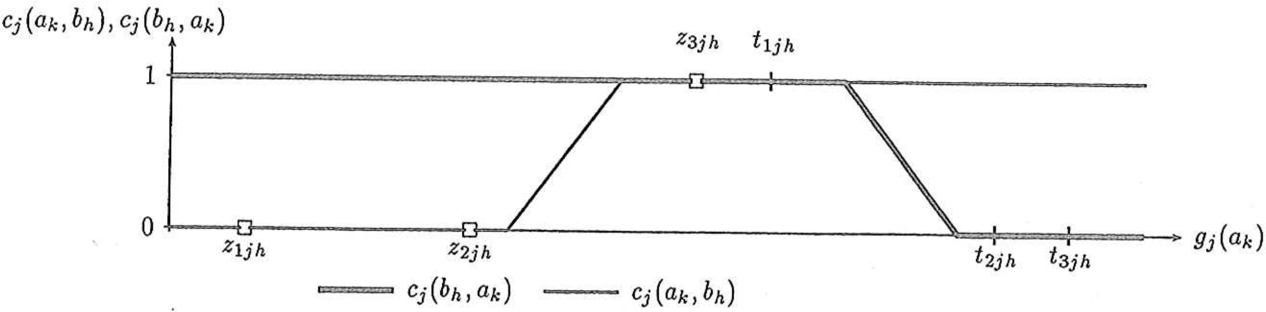

Figure 4: Values around the transition between 0 and 1

The values z_{2jh} and t_{2jh} are to be relaxed according to the following rules:
- if $z_{1jh} > -\infty$ and $z_{3jh} < t_{1jh}$, then z_{2jh} will be relaxed.
- if $t_{3jh} < +\infty$ and $z_{3jh} < t_{1jh}$, then t_{2jh} will be relaxed.

This choice ensures the symmetrical condition (see [Ngo98] for more details).

The indices $c_j(a_k, b_h), c_j(b_h, a_k)$ corresponding to relaxed values z_{2jh} and t_{2jh} are variables of the new program, and will be denoted by cz_{jh}, ct_{jh}. Other values for $c_j(a_k, b_h), c_j(b_h, a_k)$ are now fixed (0 or 1) on the values obtained from solution to the previous program.

The constraints $c_j(a_k, b_h) + c_j(b_h, a_k) \geq 1$ can be eliminated. For example, if a_k corresponds to z_{2jh}, then $c_j(a_k, b_h) = 0$ and $c_j(b_h, a_k) = 1$, while $c_j(a_k, b_h)$ becomes a variable $\in [0, 1]$, $c_j(b_h, a_k)$ is a constant $(= 1)$, the constraints are always satisfied. For other types of contraints, only constraints concerning β, cz_{jh}, ct_{jh} remain in the program.

We obtain the following program:
Program $P2$

$$\max \beta \tag{27}$$

s.t.

$$
\begin{aligned}
\beta &\leq \sum_{j \in F} k_j c_j(a_k, b_{h_k - 1}) - \lambda, \forall k \in K & (28)\\
\beta + \epsilon &\leq \lambda - \sum_{j \in F} k_j c_j(a_k, b_{h_k}), \forall k \in K & (29)\\
\beta + \epsilon &\leq \lambda - \sum_{j \in F} k_j c_j(b_{h_k - 2}, a_k), \forall k \in K & (30)\\
c_j(a_k, b_{h+1}) &\leq c_j(a_k, b_h), \forall j \in F, \forall k \in K, h = 1, 2, ..., p-1 & (31)\\
c_j(b_{h+1}, a_k) &\geq c_j(b_h, a_k), \forall j \in F, \forall k \in K, h = 1, 2, ..., p-1 & (32)\\
c_j(a_k, b_h) &\leq c_j(a_l, b_h), \forall j \in F, \forall k, l \in K, h \in B, \text{ if } g_j(a_k) < g_j(a_l) & (33)\\
c_j(a_k, b_h) &= c_j(a_l, b_h), \forall j \in F, \forall k, l \in K, h \in B, \text{ if } g_j(a_k) = g_j(a_l) & (34)\\
c_j(b_h, a_k) &\geq c_j(b_h, a_l), \forall j \in F, \forall k, l \in K, h \in B, \text{ if } g_j(a_k) < g_j(a_l) & (35)\\
c_j(b_h, a_k) &= c_j(b_h, a_l), \forall j \in F, \forall k \in K, h \in B, \text{ if } g_j(a_k) = g_j(a_l) & (36)\\
0.5 + \epsilon &\leq \lambda \leq 1 & (37)\\
cz_{2jh} &\in [0, 1], \forall j \in F, \forall k \in K, h \in B, \text{ if } \exists z_{2jh} & (38)\\
ct_{2jh} &\in [0, 1], \forall j \in F, \forall k \in K, h \in B, \text{ if } \exists t_{2jh} & (39)
\end{aligned}
$$

Despite its complexity, this refinement fails to show any improvement in the experiments realized (see section 6). The difficulties are due to the satisfaction of the symmetrical condition, which is impossible to represent by means of linear constraints.

4 Phase 2: determination of category limits from local concordance indices

Once all the local concordance indices $c_j(a_k, b_h)$ and $c_j(b_h, a_k)$ are determined, so are x_k, y_k, z_k as well as β. All values of $g_j(b_h), p_j(b_h), q_j(b_h)$ satisfying the following conditions can be accepted.

$$
\begin{aligned}
&c_j(a_k, b_h) = 0 \Rightarrow g_j(b_h) - p_j(b_h) \geq g_j(a_k) && \text{(D1)}\\
&c_j(a_k, b_h) = 1 \Rightarrow g_j(b_h) - q_j(b_h) \leq g_j(a_k) && \text{(D2)}\\
&\text{Otherwise, } c_j(a_k, b_h) = \tfrac{g_j(a_k) + p_j(b_h) - g_j(b_h)}{p_j(b_h) - q_j(b_h)} && \text{(D3)}\\
&c_j(b_h, a_k) = 0 \Rightarrow g_j(b_h) + p_j(b_h) \leq g_j(a_k) && \text{(D4)}\\
&c_j(b_h, a_k) = 1 \Rightarrow g_j(b_h) + q_j(b_h) \geq g_j(a_k) && \text{(D5)}\\
&\text{Otherwise, } c_j(b_h, a_k) = \tfrac{g_j(b_h) + p_j(b_h) - g_j(a_k)}{p_j(b_h) - q_j(b_h)} && \text{(D6)}\\
&g_j(b_{h+1}) \geq g_j(b_h) && \text{(D7)}\\
&p_j(b_h) \geq q_j(b_h) && \text{(D8)}\\
&q_j(b_h) \geq 0 && \text{(D9)}.
\end{aligned}
$$

Under the integrity hypothesis, the conditions (D3) and (D6) are not considered, we have a certain degree of freedom in determining $g_j(b_h), p_j(b_h), q_j(b_h)$. To determine these values, intuitively, we consider

an ideal solution as one that haves the following two characteristics:
- the profile characterization function $\phi_{jh}(a_k)$ has a reasonable form which depends on the ratio $\frac{q_j(b_h)}{p_j(b_h)}$.
- for each criterion g_j, the profiles $g_j(b_h), h \in K$ are well "distributed" along the scale.

These two characteristics can be used as a guideline for a multi-objective optimization program, or least, an optimization program with an objective function which can aggregate these two characteristics. In this paper, we propose a direct computation of these values in which we try to position $g_j(b_h)$ the most possible to the center of the plateau of the profile characterization function $\phi_{jh}(g_j(a_k))$, and then establish $p_j(b_h)$ the largest possible. Finally, $q_j(b_h)$ will be positioned the most approximatively to $\frac{p_j(b_h)}{2}$.

It is obvious that the dertermination of these values also concerns the transitions between 0 and 1 of the local concordance indices. Therefore, we will make use of the values $z_{ijh}, t_{ijh}, i = 1, 2, 3, j \in F, h \in B$ defined in (26).

For each $j \in F$, we proceed by decreasing order of the categories $h = p, p-1, \ldots$ by assuming that $g_j(b_{p+1}) = +\infty$.

Algorithm 4.1
For $j = 1..m$ *do*
  For $h = p..1$ *do*
    $g_j(b_h) = \min\{\frac{t_{2jh}+z_{2jh}}{2}, g_j(b_{h+1})\}$
    $p_j(b_h) = \min\{t_{2jh} - g_j(b_h), g_j(b_h) - z_{2jh}\}$
    $q_j(b_h) = \max\{\frac{p_j(b_h)}{2}, t_{1jh} - g_j(b_h), g_j(b_h) - z_{3jh}\}$
  endfor h
endfor j

Let us now prove that the conditions (D1)-(D9) are satisfied. To simplify the notation, we use $c_j(z_{2jh}, b_h)$ instead of $c_j(a_k, b_h)$ where $g_j(a_k) = z_{2jh}$.

Proposition 4.1 *$\forall j, h$ it holds*
(i) $z_{3jh} > z_{2jh} > z_{1jh}$
(ii) $t_{3jh} > t_{2jh} > t_{1jh}$
(iii) $t_{1jh} \geq z_{2jh}$
(iv) $t_{2jh} \geq z_{3jh}$
(v) $\neg\exists a_k$ *such that* $g_j(a_k)$ *is in the intervals limited by* $z_{1jh}, z_{2jh}, z_{3jh}$ *or* $t_{1jh}, t_{2jh}, t_{3jh}$.
(vi) - $z_{rjh} \leq z_{rj(h+1)}, t_{rjh} \leq t_{rj(h+1)}, r = 1, 2, 3$

Demonstration

(i),(ii) Obvious from definitions.

(iii) By definition, $c_j(z_{2jh}, b_h) = 0 \Rightarrow c_j(b_h, z_{2jh}) = 1 \Rightarrow z_{2jh} \leq t_{1jh} = \max\{g_j(a_k) | a_k \in A^*, c_j(b_h, a_k) = 1\}$.

(iv) Similar to (iii).

(v) Obvious from definition.

(vi) We demonstrate only for z_{2jh}, the other cases are similar.
By definition, $0 = c_j(z_{2jh}, b_h) \geq c_j(z_{2jh}, b_{h+1}) \Rightarrow c_j(z_{2jh}, b_{h+1}) = 0$
$\Rightarrow z_{2jh} \leq z_{2j(h+1)} = \max\{g_j(a_k) | a_k \in A^*, c_j(a_k, b_{h+1}) = 0\}$.

■

Proposition 4.2 $\forall h \in B, z_{2jh} \leq \frac{t_{1jh}+z_{2jh}}{2} < \frac{t_{2jh}+z_{3jh}}{2} \leq t_{2jh}$

Demonstration From proposition 4.1, we have:
$z_{2jh} \leq \frac{t_{1jh}+z_{jh}}{2} < \frac{t_{2jh}+z_{3jh}}{2} \leq t_{2jh}$
And from the procedure: $g_j(b_h) \leq \frac{t_{2jh}+z_{2jh}}{2}$
$z_{2jh} < z_{3jh} \Rightarrow g_j(b_h) < \frac{t_{2jh}+z_{3jh}}{2}$
For the first categorie b_p, we have $g_j(b_{(p+1)}) = +\infty$ then
$g_j(b_p) = \frac{t_{2jp}+z_{2jp}}{2}$.
$t_{2jp} > t_{1jp} \Rightarrow g_j(b_p) > \frac{t_{1jp}+z_{2jp}}{2}$.
Suppose that the inequation $g_j(b_i) > \frac{t_{1ji}+z_{2ji}}{2}$ holds $i = h+1$.
$g_j(b_h) = \min\{\frac{t_{2jh}+z_{2jh}}{2}, g_j(b_{h+1})\}$
$\frac{t_{2jh}+z_{2jh}}{2} > \frac{t_{1jh}+z_{2jh}}{2}$ and $g_j(b_{h+1}) > \frac{t_{1j(h+1)}+z_{2j(h+1)}}{2} \geq \frac{t_{1jh}+z_{2jh}}{2}$
$\Rightarrow g_j(b_h) > \frac{t_{1jh}+z_{2jh}}{2}$. Therefore, the inequation holds for $i = h$.
By induction, the inequation holds $\forall m \in B$.

■

All the conditions (D1)-(D9) can be verified easily from propositions 4.1 and 4.2.

5 How to deal with additional information ?

In the course of the interactive process, the DM may want to add information concerning the category limits (which can take the form of upper and/or lower bounds for $g_j(b_h), q_j(b_h), p_j(b_h)$) as well as the nature of the criteria. While such information can be taken into account directly in the second phase, it is not the case with the first phase as $g_j(b_h), p_j(b_h), q_j(b_h)$ do not intervene explicitly (through variables). We will discuss hereafter how to integrate these constraints in the first phase.

5.1 Constraints on the profiles and the thresholds

In order to integrate constraints on the profiles and the thresholds in the first phase, we construct rules generating constraints on $c_j(a_k, b_h), c_j(b_h, a_k)$ from given constraints on $g_j(b_h), p_j(b_h), q_j(b_h)$. These rules are presented in proposition 5.1.

Proposition 5.1 *We have the following generating rules which hold* $\forall j \in F, \forall k, l \in K, \forall h \in B$.

Original constraints	*Generating rules*	*Rule #*
$b_{jh} \leq g_j(b_h)$	*if* $g_j(a_k) \leq b_{jh}$ *then* $c_j(b_h, a_k) = 1$	*R1*
$g_j(b_h) \leq B_{jh}$	*if* $g_j(a_k) \geq B_{jh}$ *then* $c_j(a_k, b_h) = 1$	*R2*
$q_{jh} \leq q_j(b_h)$	*if* $\lvert g_j(a_l) - g_j(a_k)\rvert < 2q_{jh}$ *then* $c_j(a_k, b_h) + c_j(b_h, a_l) \geq 1$	*R3*
$q_j(b_h) \leq Q_{jh}$	*if* $g_j(a_l) - g_j(a_k) > 2Q_{jh}$ *then* $c_j(a_k, b_h) + c_j(b_h, a_l) \leq 1$	*R4*
$p_j(b_h) \leq P_{jh}$	*if* $g_j(a_l) - g_j(a_k) > 2P_{jh}$ *then* $c_j(a_k, b_h) + c_j(b_h, a_l) \leq 1$	*R5*
$p_{jh} \leq p_j(b_h)$	*if* $g_j(a_l) - g_j(a_k) < 2p_{jh}$ *then* $c_j(a_k, b_h) + c_j(b_h, a_l) \geq 1$	*R6*
$b_{jh} \leq g_j(b_h) \leq B_{jh}$	$g_j(a_k) \leq b_{jh} - P_{jh} \Rightarrow c_j(a_k, b_h) = 0$	*R7.1*
$q_{jh} \leq q_j(b_h) \leq Q_{jh}$	$g_j(a_k) \geq B_{jh} - q_{jh} \Rightarrow c_j(a_k, b_h) = 1$	*R7.2*
$p_{jh} \leq p_j(b_h) \leq P_{jh}$	$g_j(a_k) \leq b_{jh} + q_{jh} \Rightarrow c_j(b_h, a_k) = 1$	*R7.3*
	$g_j(a_k) \geq B_{jh} + P_{jh} \Rightarrow c_j(b_h, a_k) = 0$	*R7.4*

Whenever we have an additional constraint, we add the corresponding generated constraints into the program. However, it should be noticed that the generated constraints are not equivalent (necessary but not sufficient) to the original constraints as we can see in the demonstration.

Demonstration

- (R1) $g_j(a_k) \leq b_{jh} \Rightarrow g_j(a_k) \leq g_j(b_h) \Rightarrow c_j(b_h, a_k) = 1$.

- (R2) $g_j(a_k) \geq B_{jh} \Rightarrow g_j(a_k) \geq g_j(b_h) \Rightarrow c_j(a_k, b_h) = 1$.

- (R3) $[(c_j(a_k, b_h) < 1)$ and $(c_j(b_h, a_l) < 1)] \Rightarrow [(g_j(a_k) < g_j(b_h) - q_j(b_h))$ and $(g_j(a_l) > g_j(b_h) + q_j(b_h))]$
  $\Rightarrow g_j(a_l) - g_j(a_k) \geq 2g_j(b_h) \geq 2q_{jh}$.
  We have
  $\lvert g_j(a_l) - g_j(a_k)\rvert < 2q_{jh} \Rightarrow$ not $[c_j(a_k, b_h) < 1$ and $c_j(b_h, a_l) < 1] \Rightarrow c_j(a_k, b_h) + c_j(b_h, a_l) \geq 1$.

- (R4) $[(c_j(a_k, b_h) = 1)$ and $(c_j(b_h, a_l) = 1)] \Rightarrow [(g_j(a_k) > g_j(b_h) - q_j(b_h))$ and $(g_j(a_l) < g_j(b_h) + q_j(b_h))]$
  $\Rightarrow g_j(a_l) - g_j(a_k) \leq 2q_j(b_h) \leq 2Q_{jh}$
  We have
  $g_j(a_l) - g_j(a_k) > 2Q_{jh} \Rightarrow$ not $[(c_j(a_k, b_h) = 1)$ and $(c_j(b_h, a_l) = 1)] \Rightarrow c_j(a_k, b_h) + c_j(b_h, a_l) \leq 1$ (under integrity hypothesis).

- (R5) $[(c_j(a_k, b_h) > 0)$ and $(c_j(b_h, a_l) > 0)] \Rightarrow [(g_j(a_k) > g_j(b_h) - p_j(b_h))$ and $(g_j(a_l) < g_j(b_h) + p_j(b_h))]$
  $\Rightarrow g_j(a_l) - g_j(a_k) \leq 2p_j(b_h) \leq 2P_{jh}$
  So we have
  $g_j(a_l) - g_j(a_k) > 2P_{jh} \Rightarrow$ not $[(c_j(a_k, b_h) > 0)$ and $(c_j(b_h, a_l) > 0)] \Rightarrow c_j(a_k, b_h) + c_j(b_h, a_l) \leq 1$

- (R6) $[(c_j(a_k, b_h) = 0)$ and $(c_j(b_h, a_l) = 0)] \Rightarrow [(g_j(a_k) < g_j(b_h) - p_j(b_h))$ and $(g_j(a_l) > g_j(b_h) + p_j(b_h))]$
  $\Rightarrow g_j(a_l) - g_j(a_k) \geq 2p_j(b_h) \geq 2p_{jh}$
  So we have
  $g_j(a_l) - g_j(a_k) < 2p_{jh} \Rightarrow$ not $[(c_j(a_k, b_h) = 0)$ and $(c_j(b_h, a_l) = 0)] \Rightarrow c_j(a_k, b_h) + c_j(b_h, a_l) \geq 1$

(under integrity hypothesis).

- (R7)
  1/ $g_j(a_k) \leq b_{jh} - P_{jh} \leq g_j(b_h) - p_j(b_h) \Rightarrow c_j(a_k, b_h) = 0$
  2/ $g_j(a_k) \geq B_{jh} - q_{jh} \geq g_j(b_h) - q_j(b_h) \Rightarrow c_j(a_k, b_h) = 1$
  3/ $g_j(a_k) \leq b_{jh} + q_{jh} \leq g_j(b_h) + q_j(b_h) \Rightarrow c_j(b_h, a_k) = 1$
  4/ $g_j(a_k) \geq B_{jh} + P_{jh} \geq g_j(b_h) + p_j(b_h) \Rightarrow c_j(b_h, a_k) = 0$

5.2 Constraints on the nature of criteria

The DM may want to build a model in which the nature of the criteria is specified, i.e., less general than the pseudo-criterion considered in our general framework. This information leads to some more constraints to add into the program.

- **Quasi-criterion** $p = q$
  We have $c_j(a_k, b_h), c_j(a_h, b_k) \in \{0, 1\}$, i.e., we do not have to introduce the integrity hypothesis as it is already satisfied.

- **Pre-criterion** $q = 0$
  In this case, we can introduce these constraints into the program $P1$:
  $c_j(a_k, b_h) + c_j(b_h, a_k) = 1, \forall j \in F, \forall h \in B, \forall k, l \in K$
  This is a special case of $q_{jh} \leq q_j(b_h) \leq Q_{jh}$ where $q_{jh} = Q_{jh} = 0$.

- **True-criterion** $p = q = 0$
  The same as with pre-criterion and quasi-criterion, i.e.:
  - there is no need of the integrity hypothesis,
  - the constraints $c_j(a_k, b_h) + c_j(b_h, a_k) = 1, \forall j \in F, \forall h \in B, \forall k, l \in K$ will be inserted into the program.

5.3 How to get a strong consistency

If we want to always ensure the consistency of the categories in A, independently of the set A^*, we must base on the following condition (see [Yu92b] and [Yu92a]):
$g_j(b_{h+1}) \geq g_j(b_h) + p_j(b_h) + p_j(b_{h+1}), \forall j \in F, h \in B$
This condition is indeed a sufficient one to ensure the consistency in A (but not necessary). To introduce this condition into the program $P1$, we have to represent it by means of the local concordance indices. As we know, in the reconstruction of a continuous function ($c_j(x, b_h)$ for example, here x is concretized by $g_j(a_k)$) from a set of discrete points ($c_j(a_k, b_h)$), a loss of information is unavoidable. In our case, we do not have an equivalence condition but only either a necessary condition (stated in proposition 5.2.(i)), or a sufficient condition (proposition 5.2.(ii)).

For each $j \in F$, consider a permutation $\sigma_j(k), k \in K$ such that $g_j(a_{\sigma_j(k)}) \leq g_j(a_{\sigma_j(k+1)})$.

Proposition 5.2 *(i) if* $g_j(b_{h+1}) \geq g_j(b_h) + p_j(b_h) + p_j(b_{h+1})$, *then* $\min\{c_j(b_h, a_k), c_j(a_k, b_{h+1})\} = 0$.
(ii) if $\min\{c_j(b_h, a_{\sigma(k)}), c_j(a_{\sigma(k+1)}, b_{h+1})\} = 0$, *then* $g_j(b_{h+1}) \geq h_j(b_h) + p_j(b_h) + p_j(b_{h+1})$

Demonstration
(i) $c_j(b_h, a_k) > 0 \Rightarrow g_j(a_k) < g_j(b_h) + p_j(b_h) \leq g_j(b_{h+1}) - p_j(b_{h+1}) \Rightarrow c_j(a_k, b_{h+1}) = 0$
(ii) Let $a_{\sigma(k*)}$ the first alternative such that $c_j(b_h, a_{\sigma(k^*)}) = 0$, then $c_j(b_h, a_{\sigma(k^*-1)}) > 0$, the existence of these two alternatives is ensured by the two additional fictitious alternatives a_*, a^*.
Replace k by $k^* - 1$ in the condition $\min\{c_j(b_h, a_{\sigma(k)}), c_j(a_{\sigma(k+1)}, b_{h+1})\} = 0$, we obtain $c_j(b_h, a_{\sigma(k^*-1)}) > 0 \Rightarrow c_j(a_{\sigma(k^*}, b_{h+1}) = 0$
We have
$c_j(b_h, a_{\sigma(k^*)}) = 0 \Rightarrow g_j(a_{\sigma(k^*)}) \geq g_j(b_h) + p_j(b_h)$
$c_j(a_{\sigma(k^*)}, b_{h+1}) = 0 \Rightarrow g_j(a_{\sigma(k^*)} < g_j(b_{h+1}) - p_j(b_{h+1})$
Therefore,
$g_j(b_h) + p_j(b_h) \leq g_j(a_{\sigma(k^*)}) < g_j(b_{h+1}) - p_j(b_{h+1}) \Rightarrow g_j(b_{h+1}) \geq g_j(b_h) + p_j(b_h) + p_j(b_{h+1})$

$\blacksquare$

To ensure the consistency of categories in A despite the set A^* given, we have to introduce the constraints given by proposition 5.2.(ii) into the program $P1$.

6 Empirical validation of the inference procedure

The experimental questions are the following:

- Are the assignements of alternatives from A^* more stable when using the results of the program than when considering the profiles given by the DM? The stability of assignments refers to the invariance of the assignments to changes of the category mimits. In other words, is the tool able to increase the stability of assignments of alternatives in a set A^*?
- The results depend on the information given as input, i.e., on the set A^* of assignment examples. How large should A^* be in order to derive the profiles in a reliable manner?
- In practical decision situations, real DMs do not always produce reliable information. The tool should be able to highlight the assignment examples that are contradictory or not representable through the ELECTRE TRI preference model. Therefore, a question to consider concerns the reliability of the optimization procedure to identify inconsistencies in the DM's judgments?

6.1 Experimental Design

In this experiment, we consider only the program $P1$ without additional conditions, and the construction of the profiles. This experiment is a laboratory work, i.e., it takes its material from a past real world case study to perform *a posteriori* computation in order to test the operational validity of the optimization model proposed. The data considered are taken from [MFN97] which, in turn, come from the real world application described in [Yu92a].

This application considers the problem of assigning a set A of 100 alternatives ($A = \{a_1, a_2, ..., a_{100}\}$ is described in appendix A) to three ordered categories C_1, C_2 and C_3 on the basis of seven criteria (preferences on all criteria are decreasing with the evaluations, i.e., the lower the better).

As no interaction with the DM is possible, we consider the assignment of the ELECTRE TRI pessimistic procedure (with the parameters given in [Yu92a], see appendix B) as assignment examples expressed by a "fictitious" DM. Concerning the importance coefficients, we take the mean value of the coefficients inferred for the group of A^j_{48} (the one with the largest size) which are in the results of the procedure of inference of importance coefficients presented in [MFN97].

We randomly generate 80 subsets of A, the cardinality of these subsets being respectively 6, 12, 18, 24, 30, 36, 42, 48 (10 sets of each size), denoted by A^j_i the j^{th} subset of size i. Each of these subsets is conceived so that the alternatives are assigned uniformly on the three categories.

The ability to improve the "stability" of the assignment of the alternatives is observed through the value $\beta_0(i) - \beta_d(i)$ where $i \in \{6, ..., 48\}$ is the size of the subsets A^* chosen, $\beta_o(i), \beta_d(i)$ are respectively the mean stability (for all subset A^j_i) of assignments resulting from our procedure and that computed from values given in appendix B (representing the "fictitious" DM).

To determine the minimum required size of A^*, we compute Err_{100}, the percentage of assignment errors resulting from the use of the obtained profiles on the whole set A.

To estimate the capacity to identify the inconsistencies in the assertions of the DM, we intentionally introduce in A^* an "assignment error". Let β the resulting stability of assignment (usually < 0), an inconsistency is identified if the alternative under consideration is found in the set E of alternatives which are the most difficult to assign ($E = \{a_k, \min\{x_k, y_k, z_k\} = \beta\}$). We will observe $\beta(i)$ (mean stability of the subsets having size i), $n(i)$ (mean cardinalities of E). For this question, we consider 48 subsets E^j_i ($i = \{6, ..., 48\}, j = 1..6$), each of them with one assignment error of type j.

Type j	Initial Cat.	Erroneous Cat.
1	C_1	C_2
2	C_2	C_1
3	C_2	C_3
4	C_3	C_2
5	C_1	C_3
6	C_3	C_1

Table 1: Types of errors introduced

6.2 Results

6.2.1 Ability to improve the stability of the assignment

The results of the test are summarized in the table 2:
Considering these results, we can observe:

- first, that the results show that the larger the set of assignment examples, the less stable the assignments, i.e., the more sensitive are these assignments to a change in profiles. This is obvious as each assignment example adds 3+5mp constraints to the program,
- second, within this set of data, there is a considerable improvement of the stability of the assignments whatever the size of the set of examples.

Size: i	$\beta_o(i)$	$\beta_d(i)$	$\beta_o(i) - \beta_d(i)$
6	0.2692	0.0723	0.1969
12	0.2288	0.0059	0.2229
18	0.2019	0.0636	0.1383
24	0.2019	0.0616	0.1403
30	0.1966	0.0563	0.1403
36	0.1966	0.0611	0.1355
42	0.2019	0.0298	0.1721
48	0.1913	0.0490	0.1423
		mean	0.1611

Table 2: Improvement of the stability of the solution

6.2.2 The amount of information necessary

We observe now the means of assignment errors in A when different sizes of A^* are considered.
The parameters to be inferred, $g_j(b_h), p_j(b_h), q_j(b_h)$, (there are $3mp$ parameters) depend on the number of criteria as well as the number of categories. Considering the results in Table 3, it seems that 2mp (28=2x7x2 in this example) is a reasonable balance for the estimation of the number of assignment examples to infer weights in a reliable way. However, it is important to notice that, in this example, we have to accept a certain tolerance of errors, approximately 1.5% .

Size: i	$Err_{100}(i)$
6	9.9
12	5.6
18	3.3
24	1.2
30	2.1
36	1.1
42	1.4
48	1.0

Table 3: The necessary size to infer the profiles

6.2.3 Ability to identify inconsistencies in assignments

The results of the test are summarized in the Table 4:
In all the tests, the "wrongly" assigned alternative is found in the alternatives being the most difficult to assign ($min\{x_k, y_k, z_k\} = \beta$). However, within this experiment, the model does not seem to be very efficient in identifying errors as the most difficult examples represent a large proportion of A^* $\frac{n(i)}{i}$.

Size: i	$\beta(i)$	$n(i)$	$n(i)/i$
6	0.10	3.36	0.56
12	0.13	7.68	0.64
18	0.04	6.30	0.35
24	0.03	7.68	0.32
30	0.04	13.80	0.46
36	0.01	12.96	0.36
42	0.02	11.34	0.27
48	0.00	25.92	0.54

Table 4: Identification of "errors"

7 Conclusions and further research

This paper presents an inference procedure aiming at inferring the category limits of the ELECTRE TRI method on the basis of assignment examples. This procedure is grounded on a mathematical program formulation and is validated through a laboratory experiment. This inference procedure is intended to be used in an interactive aggregation-disaggregation process. Moreover, this procedure is complementary to the weight inference procedure [MFN97] and both can be used iteratively (fixing weights and inferring category limits, and then fixing category limits and inferring weights).

But this approach suffers from certain weaknesses. The major difficulty is due to the integrity hypothesis and the construction of continue functions $\phi_{jh}(x)$ from discrete values. Under such restrictions, we can not always ensure an improvement of the stability of the solution given by the program. What we can say (see also [Ngo98]) is that if we want to remain in a direction of constructing the profiles (characterized by the profile characterization function $\phi_{jh}(a_k)$) by means of discrete values of the local concordance indices $c_j(a_k, b_h)$ and using a linear program, we can not get better solution. So, if we want to improve the situation of the inference of profiles, other direction should be considered.

In conclusion, we can state the following:

- The ideal case to apply this model is that of a true- or quasi-criterion, where $p = q$, and the integrity hypothesis is naturally satisfied.
- Otherwise, we can use this model knowing its limitations. The program can be used to obtain a first estimation of the profiles, but it is not suitable to refine the result during the process.
- Within the data considered in the experiences, the profiles obtained assign the examples correctly in a stable way. The stability of assignment β is improved considerably.
- We propose, with a certain reserve, that the reasonable estimation of the number of examples necessary to infer the profiles be $2mp$, where m and p are the number of criteria and the number of profiles, respectively.

Despite its limitations, it appears that the proposed inference procedure is suitable for a DM to define category limits of ELECTRE TRI method providing assignment examples. Moreover, we believe that such procedure is helpful in order to provide a formal framework for the DM to learn about the relation between the category limits and his/her preferences in a constructive learning process.

References

[MFN97] V. Mousseau, J. Figueira, and J.Ph. Naux. Using Assignment Examples to Infer Weights for ELECTRE TRI Method: Some Experimental Results. Cahiers du LAMSADE no 150, Université de Paris-Dauphine, 1997.

[MS98] V. Mousseau and R. Slowinski. Inferring an ELECTRE TRI Model from Assignment Examples. *Journal of Global Optimization*, 12(2):157–174, 1998.

[MSZ00] V. Mousseau, R. Slowinski, and P. Zielniewicz. A user-oriented implementation of the electre-tri method integrating preference elicitation support. *Computers and Operations Research*, (27), 757-777, 2000.

[Nau96] J.-Ph. Naux. Calage d'un modèle ELECTRE TRI à l'aide d'actions exemples. Mémoire de DEA, Université Paris-Dauphine, 1996.

[Ngo98] T. Ngo. Inférence des profiles limites dans le calage d'un modèle de type ELECTRE TRI. Mémoire de DEA, Université Paris-Dauphine, 1998.

[RB93] B. Roy and D. Bouyssou. *Aide multicritère à la décision : méthodes et cas.* Economica, Paris, 1993.

[Roy85] B. Roy. *Méthodologie multicritère d'aide à la décision.* Economica, Paris, 1985.

[RV84] B. Roy and Ph. Vincke. Relational systems of preferences with one or more pseudo-criteria: some new concepts and results. *Management Science*, 30(11):1323–1334, 1984.

[Yu92a] W. Yu. *Aide multicritère à la décision dans le cadre de la problématique du tri : concepts, méthodes et applications.* PhD thesis, Université Paris-Dauphine, 1992.

[Yu92b] W. Yu. ELECTRE TRI : Aspects méthodologiques et manuel d'utilisation. Document du LAMSADE n^o74, Université Paris-Dauphine, 1992.

Appendix

Appendix A: Assignment of alternatives using the initial weights

Num.	Name	g_1	g_2	g_3	g_4	g_5	g_6	g_7	Categ.
1	a_1	14.47	14.02	18.02	7.24	22.16	3.02	58.65	C_1
2	a_6	14.15	11.01	18.02	19.13	49.87	14.83	58.65	C_1
3	a_9	14.54	14.02	18.02	4.65	12.92	3.02	58.65	C_1
4	a_{12}	14.42	11.01	18.02	4.65	12.92	8.91	58.64	C_1
5	a_{16}	14.23	7.99	18.02	4.65	22.16	8.91	58.65	C_1
6	a_{17}	14.42	11.01	18.02	16.55	12.92	3.02	58.65	C_1
7	a_{18}	13.66	7.99	31.49	81.60	40.64	30.70	61.59	C_1
8	a_{19}	13.02	15.74	18.02	7.24	79.21	42.63	79.32	C_1
9	a_{20}	13.39	4.98	26.96	81.60	69.01	18.77	61.59	C_1
10	a_{21}	13.60	4.02	22.44	4.65	31.40	31.73	58.65	C_1
11	a_{22}	13.75	11.01	22.44	81.13	69.01	7.88	21.71	C_1
12	a_{23}	13.66	11.01	14.11	70.55	69.01	18.77	42.39	C_1
13	a_{24}	13.80	14.02	14.11	60.43	59.11	8.91	20.24	C_1
14	a_{25}	13.04	7.99	22.44	40.19	40.64	6.96	100.0	C_1
15	a_{26}	12.97	7.99	18.02	29.66	31.40	18.77	100.0	C_1
16	a_{28}	13.26	11.01	14.11	7.24	22.16	7.88	79.32	C_1
17	a_{31}	13.24	11.01	14.11	4.65	12.92	6.96	100.0	C_1
18	a_{32}	13.54	14.02	14.11	4.65	12.92	3.02	58.65	C_1
19	a_{33}	13.19	11.01	18.02	7.24	12.92	7.88	79.32	C_1
20	a_{36}	13.42	11.01	14.11	4.65	12.92	8.91	58.65	C_1
21	a_{39}	13.92	7.99	18.02	7.24	22.16	6.96	100.0	C_1
22	a_{41}	13.04	7.99	22.44	7.24	31.40	14.83	58.65	C_1
23	a_{43}	13.42	11.01	14.11	16.55	12.92	3.02	58.65	C_1
24	a_{45}	12.97	7.99	18.02	7.24	40.64	14.83	58.65	C_1
25	a_{49}	13.48	1.05	18.02	6.45	31.40	18.77	100.0	C_1
26	a_{50}	12.74	11.01	22.44	91.72	69.01	7.88	21.71	C_1
27	a_{54}	12.07	7.99	14.11	4.65	12.92	7.88	79.32	C_1
28	a_{56}	10.94	7.99	14.11	7.24	12.92	7.88	79.32	C_1
29	a_{58}	10.41	1.96	18.02	19.13	22.16	18.77	100.0	C_1
30	a_{60}	11.04	7.99	14.11	4.65	12.92	3.02	58.65	C_1
31	a_{61}	10.91	7.99	18.02	7.24	12.92	3.02	58.65	C_1
32	a_{64}	9.90	4.98	18.02	71.48	59.11	7.88	60.12	C_1
33	a_{65}	9.93	7.99	18.02	61.36	49.87	3.93	79.32	C_1
34	a_{68}	9.91	7.99	14.11	7.24	12.92	3.02	58.65	C_1
35	a_{70}	9.60	4.98	14.11	7.24	12.92	6.96	100.0	C_1
36	a_{71}	9.29	1.05	14.11	7.24	22.16	18.77	100.0	C_1
37	a_{72}	8.94	4.98	18.02	81.60	59.11	7.88	40.92	C_1
38	a_{73}	8.65	1.96	18.02	71.48	59.11	7.88	60.12	C_1
39	a_{74}	8.65	4.98	18.02	61.36	40.64	6.96	100.0	C_1
40	a_{75}	8.46	1.96	18.02	61.36	49.87	6.96	100.0	C_1
41	a_{78}	8.66	4.98	14.11	7.24	12.92	3.02	58.65	C_1
42	a_{79}	8.57	4.99	14.11	7.24	12.92	3.93	79.32	C_1
43	a_{81}	7.58	1.96	18.02	71.48	49.87	7.88	60.12	C_1
44	a_{82}	7.58	4.98	18.02	61.36	31.40	6.96	100.0	C_1
45	a_{83}	7.31	1.96	14.11	19.13	12.92	7.88	79.32	C_1
46	a_{86}	6.33	1.96	18.02	61.36	31.40	6.96	100.0	C_1
47	a_{87}	6.46	4.98	18.02	50.78	12.92	6.96	100.0	C_1
48	a_2	14.64	14.02	18.02	5.46	22.16	3.02	39.44	C_2
49	a_3	14.48	11.01	18.02	17.36	22.16	20.32	39.44	C_2
50	a_4	14.65	11.01	14.11	6.43	22.16	8.91	39.44	C_2
51	a_5	14.43	11.01	18.02	29.25	22.16	31.73	39.44	C_2
52	a_7	15.04	14.02	10.47	18.32	12.92	8.91	39.44	C_2
53	a_8	14.71	14.02	18.02	6.43	12.92	3.02	39.44	C_2
54	a_{10}	14.59	11.01	18.02	6.43	12.92	8.91	39.44	C_2
55	a_{11}	14.72	11.01	14.11	18.32	12.92	8.97	39.44	C_2
56	a_{13}	14.65	14.02	18.02	6.43	22.16	14.83	39.44	C_2
57	a_{14}	14.52	11.01	18.02	6.43	22.16	3.02	39.44	C_2
58	a_{15}	14.40	7.99	18.02	6.43	22.16	8.91	39.44	C_2
59	a_{27}	13.51	14.02	18.02	29.25	22.16	8.91	39.44	C_2
60	a_{29}	13.39	11.01	18.02	17.36	22.16	8.91	39.44	C_2
61	a_{30}	13.71	14.02	14.11	6.43	12.92	3.02	39.44	C_2

following next page

following of the preceeding page

Num.	Name	g_1	g_2	g_3	g_4	g_5	g_6	g_7	Categ.
62	a_{34}	13.59	11.01	14.11	6.43	12.92	8.91	39.44	C_2
63	a_{35}	13.46	11.01	18.02	5.46	12.92	8.91	39.44	C_2
64	a_{37}	13.52	11.01	14.11	6.43	22.16	3.02	39.44	C_2
65	a_{38}	13.39	11.01	18.02	5.46	22.16	3.02	39.44	C_2
66	a_{40}	13.40	7.99	14.11	6.43	22.16	8.91	39.44	C_2
67	a_{42}	13.79	11.01	10.47	18.32	12.92	8.91	39.44	C_2
68	a_{44}	13.46	11.01	18.02	6.43	12.92	3.02	39.44	C_2
69	a_{46}	13.40	11.01	18.02	6.43	22.16	14.83	39.44	C_2
70	a_{47}	13.15	4.98	18.02	6.43	22.16	8.91	39.44	C_2
71	a_{48}	13.24	4.98	14.11	6.43	22.16	20.32	39.44	C_2
72	a_{51}	12.07	7.99	18.02	29.25	31.40	8.91	39.44	C_2
73	a_{52}	12.46	11.01	14.11	5.46	12.92	8.91	39.44	C_2
74	a_{53}	12.46	11.01	14.11	6.43	12.92	3.02	39.44	C_2
75	a_{55}	12.14	7.99	18.02	5.46	22.16	3.02	39.44	C_2
76	a_{57}	11.08	7.99	18.02	17.36	12.92	8.91	39.44	C_2
77	a_{59}	11.01	7.99	18.02	17.36	22.16	3.02	39.44	C_2
78	a_{62}	11.07	7.99	18.02	5.46	12.92	3.02	39.44	C_2
79	a_{63}	11.40	11.01	14.11	6.43	3.76	3.02	39.44	C_2
80	a_{67}	10.08	7.99	14.11	5.46	12.92	3.02	39.44	C_2
81	a_{76}	8.76	4.98	14.11	17.36	22.16	3.02	39.44	C_2
82	a_{80}	8.71	1.96	14.11	5.46	12.92	8.91	39.44	C_2
83	a_{88}	6.65	4.98	14.11	5.46	3.76	14.83	39.44	C_2
84	a_{89}	5.28	1.05	18.02	60.43	40.64	3.02	20.24	C_2
85	a_{90}	4.06	1.05	18.02	39.78	12.92	3.02	39.44	C_2
86	a_{91}	4.64	1.96	6.74	5.46	3.76	3.02	39.44	C_2
87	a_{66}	10.25	7.99	14.11	15.58	12.92	8.91	20.24	C_3
88	a_{69}	10.65	11.01	10.47	3.69	3.76	8.91	20.24	C_3
89	a_{77}	9.40	7.99	10.47	3.69	3.76	8.91	20.24	C_3
90	a_{84}	8.26	7.99	10.47	3.69	3.76	3.02	20.24	C_3
91	a_{85}	8.45	7.99	6.74	3.69	3.76	8.91	20.24	C_3
92	a_{92}	5.11	4.98	3.02	3.69	3.76	3.02	20.24	C_3
93	a_{93}	3.68	1.96	6.74	15.58	3.76	3.02	20.24	C_3
94	a_{94}	3.86	1.96	3.02	3.69	3.76	3.02	20.24	C_3
95	a_{95}	2.56	1.96	6.74	15.58	3.76	14.83	20.24	C_3
96	a_{96}	2.81	1.96	3.02	3.69	5.41	3.02	20.24	C_3
97	a_{97}	1.04	1.05	14.11	49.90	3.76	3.02	20.24	C_3
98	a_{98}	1.48	1.05	3.02	15.58	3.76	3.02	20.24	C_3
99	a_{99}	1.68	1.96	3.02	15.58	5.41	3.02	20.24	C_3
100	a_{100}	0.60	1.05	0.71	3.69	5.41	3.02	20.24	C_3

Appendix B: Parameters in the original dataset

g_j	w_j	$g_j(b_1)$	$q_j(b_1)$	$p_j(b_1)$	$v_j(b_1)$	$g_j(b_2)$	$q_j(b_2)$	$p_j(b_2)$	$v_j(b_2)$
1	3.0	17.0	0.67	1.34	2.01	14.0	0.64	1.28	1.92
2	1.5	43.0	1.84	3.73	11.19	29.0	1.56	3.17	9.51
3	1.5	43.0	1.84	3.73	11.19	29.0	1.56	3.17	9.51
4	1.5	43.0	1.84	3.73	11.19	29.0	1.56	3.17	9.51
5	1.5	43.0	1.84	3.73	11.19	29.0	1.56	3.17	9.51
6	1.5	43.0	1.84	3.73	11.19	29.0	1.56	3.17	9.51
7	2.0	40.0	2.56	5.19	20.76	27.0	2.04	4.15	16.60

Table 6: Parameters in the original dataset